

Lecture Notes in Networks and Systems 461

Harish Sharma
Vivek Shrivastava
Kusum Kumari Bharti
Lipo Wang *Editors*

Communication and Intelligent Systems

Proceedings of ICCIS 2021

 Springer

Lecture Notes in Networks and Systems

Volume 461

Series Editor

Janusz Kacprzyk, Systems Research Institute, Polish Academy of Sciences,
Warsaw, Poland

Advisory Editors

Fernando Gomide, Department of Computer Engineering and Automation—DCA,
School of Electrical and Computer Engineering—FEEC, University of
Campinas—UNICAMP, São Paulo, Brazil

Okyay Kaynak, Department of Electrical and Electronic Engineering,
Bogazici University, Istanbul, Turkey

Derong Liu, Department of Electrical and Computer Engineering, University of
Illinois at Chicago, Chicago, USA

Institute of Automation, Chinese Academy of Sciences, Beijing, China

Witold Pedrycz, Department of Electrical and Computer Engineering, University of
Alberta, Alberta, Canada

Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland

Marios M. Polycarpou, Department of Electrical and Computer Engineering,
KIOS Research Center for Intelligent Systems and Networks, University of Cyprus,
Nicosia, Cyprus

Imre J. Rudas, Óbuda University, Budapest, Hungary

Jun Wang, Department of Computer Science, City University of Hong Kong,
Kowloon, Hong Kong

The series “Lecture Notes in Networks and Systems” publishes the latest developments in Networks and Systems—quickly, informally and with high quality. Original research reported in proceedings and post-proceedings represents the core of LNNS.

Volumes published in LNNS embrace all aspects and subfields of, as well as new challenges in, Networks and Systems.

The series contains proceedings and edited volumes in systems and networks, spanning the areas of Cyber-Physical Systems, Autonomous Systems, Sensor Networks, Control Systems, Energy Systems, Automotive Systems, Biological Systems, Vehicular Networking and Connected Vehicles, Aerospace Systems, Automation, Manufacturing, Smart Grids, Nonlinear Systems, Power Systems, Robotics, Social Systems, Economic Systems and other. Of particular value to both the contributors and the readership are the short publication timeframe and the world-wide distribution and exposure which enable both a wide and rapid dissemination of research output.

The series covers the theory, applications, and perspectives on the state of the art and future developments relevant to systems and networks, decision making, control, complex processes and related areas, as embedded in the fields of interdisciplinary and applied sciences, engineering, computer science, physics, economics, social, and life sciences, as well as the paradigms and methodologies behind them.

Indexed by SCOPUS, INSPEC, WTI Frankfurt eG, zbMATH, SCImago.

All books published in the series are submitted for consideration in Web of Science.

For proposals from Asia please contact Aninda Bose (aninda.bose@springer.com).

Harish Sharma · Vivek Shrivastava ·
Kusum Kumari Bharti · Lipo Wang
Editors

Communication and Intelligent Systems

Proceedings of ICCIS 2021

 Springer

Editors

Harish Sharma
Department of Computer Science
and Engineering
Rajasthan Technical University
Kota, India

Kusum Kumari Bharti
Design and Manufacturing
Indian Institute of Information Technology
Jabalpur, India

Vivek Shrivastava
Institutional Area Narela Delhi
National Institute of Technology Delhi
New Delhi, India

Lipo Wang
School of Electrical and Electronic
Engineering
Nanyang Technological University
Singapore, Singapore

ISSN 2367-3370

ISSN 2367-3389 (electronic)

Lecture Notes in Networks and Systems

ISBN 978-981-19-2129-2

ISBN 978-981-19-2130-8 (eBook)

<https://doi.org/10.1007/978-981-19-2130-8>

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022

This work is subject to copyright. All rights are solely and exclusively licensed by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors, and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd. The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

Preface

This book contains outstanding research papers as the proceedings of the 3rd International Conference on Communication and Intelligent Systems (ICCIS 2021), which was held on 18–19 December 2021 at National Institute of Technology Delhi, India, under the technical sponsorship of the Soft Computing Research Society, India. The conference is conceived as a platform for disseminating and exchanging ideas, concepts, and results of researchers from academia and industry to develop a comprehensive understanding of the challenges of the advancements of intelligence in computational viewpoints. This book will help in strengthening congenial networking between academia and industry. This book presents novel contributions in areas of communication and intelligent systems, and it serves as reference material for advanced research. The topics covered are intelligent system: algorithms and applications, intelligent data analytics and computing, informatics and applications, and communication and control systems.

ICCIS 2021 received a significant number of technical contributed articles from distinguished participants from home and abroad. ICCIS 2021 received 476 research submissions from 43 different countries, viz. Australia, Bahrain, Bangladesh, Brazil, Bulgaria, Burkina Faso, Chile, China, Ecuador, Egypt, Ethiopia, Finland, Germany, India, Iran, Iraq, Italy, Japan, Liberia, Malaysia, Mauritius, Morocco, Nepal, Oman, Poland, Portugal, Romania, Russia, Saudi Arabia, Serbia, Singapore, Slovakia, South Africa, South Korea, Sri Lanka, Thailand, Turkey, Ukraine, United Arab Emirates, UK, USA, Viet Nam, and Yemen. After a very stringent peer-reviewing process, only 92 high-quality papers were finally accepted for presentation and final proceedings.

This book presents novel contributions in areas of communication and intelligent systems, and it serves as reference material for advanced research.

Kota, India
New Delhi, India
Singapore
Jabalpur, India

Harish Sharma
Vivek Shrivastava
Lipo Wang
Kusum Kumari Bharti

Contents

A Design of Frequency Encoded Dibit-Based Inhibitor Logic Using Reflective Semiconductor Optical Amplifier with Simulative Verification	1
Surajit Bosu and Baibaswata Bhattacharjee	
Ocean Surface Pollution Detection: Applicability Analysis of V-Net with Data Augmentation for Oil Spill and Other Related Ocean Surface Feature Monitoring	11
Naishadh Mehta, Pooja Shah, Pranshav Gajjar, and Vijay Ukani	
Simulation and Investigations of I-shaped Patch Antenna with Induced SIW and Slit for S and C Bands Radar Applications	27
P. Esther Rani and Saam Prasanth Dheeraj	
Road Network Extraction from Satellite Images Using Deep Learning	39
Yadav Maharaj and Jules-Raymond Tapamo	
An Evolutionary Online Motion Planning of Car-Like Mobile Robots with Velocity Obstacles	53
S. Ramabalan, V. Sathiya, and M. Chinnadurai	
Towards Formalization of Constructivist Seed AI	61
Swarna Kamal Paul and Parama Bhaumik	
The Effective Learning Approach to ICT-TPACK and Prediction of the Academic Performance of Students Based on Machine Learning Techniques	79
T. Saravanan, N. Nagadeepa, and B. Mukunthan	
Verification of Iris with Consideration of Constraints	95
Sayan Das and Biswajit Kar	

Theoretical Validation of Data Warehouse Requirements Metrics Based on Agent Goal Decision Information Model Using Zuse’s Framework	107
Tanu Singh and Manoj Kumar	
Corpus-Based Hashing Count Frequency Vectorization of Sentiment Analysis of Movie Reviews	119
M. Shyamala Devi, R. Aruna, Y. Lakshmi Akshitha, G. Chandana, G. Bhavisha, B. Lohitha, and M. Anusha	
Anime Scene Generator from Real-World Scenario Using Generative Adversarial Networks	129
Le Xuan Huy, Bui Thi Bich Ngoc, and Phan Duy Hung	
Employing AI for Development of a Smart Entry Log System at Entry Gates	139
Anusha Gadgil, Arjun Thakur, Mihir Gohad, Rahee Walambe, and Ketan Kotecha	
Automated Spammer Detection for Limited Length Social Media	157
Shilpa Mehta	
Ensemble Model Discovery for Prognostication of Diabetes	169
Pranjal Bahore, Shreyansh Paliwal, Dipanshu Rautela, and Rahul Chaurasiya	
Classification of Epileptic Seizure Using Machine Learning and Deep Learning Based on Electroencephalography (EEG)	179
Mohammed Tawfik, Ezzaldden Mahyoub, Zeyad A. T. Ahmed, Nasser M. Al-Zidi, and Sunil Nimbhore	
An Analytical Approach for Extracting Entities and Their Emotional Tones in Narrative Scenarios	201
V. Ashwanth and Sneha Sreedevi	
A Simple Divide-and-Conquer Algorithm for Solving an Instance of Planar Convex Hull Problems	211
Sariah López-Fierro	
COVID-19 Pandemic: Review on Emerging Technology Involvement with Cloud Computing	223
K. Anushka Xavier, S. L. Chetradavee, and N. Jayapandian	
An Empirical Examination on Forecasting VN30 Short-Term Uptrend Stocks Using LSTM along with the Ichimoku Cloud Trading Strategy	235
Pham Ngoc Hai, Hoang Trung Hieu, and Phan Duy Hung	

Applications of IoT in Industrial Transformation and Green Manufacturing 245
 Arshi Naim, Mohammad Rashid Hussain, Salem Alelyani, and Mohammed Saleh Alsaqer

Yaw Motion Control of a Ship Based on Improved Quasi-Sliding Mode 261
 Rajashree Taparia and Priya Gautam

Natural Language Inference on Imbalanced Datasets 273
 Nidarshan Kumar, Anirudh V. Ragam, GBS Akhil, and H. R. Mamatha

Robotics Process Automation Implementation in Project Management 283
 Sharad Garg, Pooja Dehraj, and Ritvik Shrivastava

A New Hybrid Boost Converter with Cuckoo Search MPPT for High Gain Enhancement of PEMFC 295
 CH. Siva Kumar and G. Mallesham

Survey on Smart Personalized Healthcare System in Fog-Assisted Cloud Environments 309
 T. Veni

News Bias Detection Using Transformers 319
 Varun Magotra, Ebrahim Hirani, Vedant Mehta, and Surekha Dholay

A Novel Feature Reduction Methodology Using Siamese and Deep Forest Classification for Intrusion Detection 327
 V. Gokula Krishnan, K. Sreerama Murthy, Ch. Viswanathasarma, K. Venkata Rao, K. Sankar, and D. Gurupandi

A Review on Deepfake Media Detection 343
 Rajneesh Rani, Tarun Kumar, and Mukund Prasad Sah

Investigation of Error-Tolerant Approximate Multipliers for Image Processing Applications 357
 D. Tilak Raju and Y. Srinivasa Rao

Artificial Intelligence Technological Revolution in Education and Space for Next Generation 371
 S. L. Chetradavee, K. Anushka Xavier, and N. Jayapandian

Frame Duplication Detection Using CNN-Based Features with PCA and Agglomerative Clustering 383
 Neetu Singla, Sushama Nagpal, and Jyotsna Singh

Detection of MA Based on Iris Blood Vessel Segmentation and Classification Using Convolutional Neural Networks (ConvNets) 393
 S. Karthika and M. Durgadevi

Implementation of Laboratory Information Management to Medical Analyzer Data Integration	411
Devashri Raich, Yashpal Singh, and Asha Ambhaikar	
Framework for the Integration of Transmission Optimization Components into LoRaWAN Stack	421
Bruno Mendes, Shani du Plessis, Dário Passos, and Noélia Correia	
Design of Low-Power Parallel Prefix Adder Templates Using Asynchronous Techniques	433
J. Sudhkar and E. Jagadeeswara Rao	
Intellectualization of Lean Production Logistic Technology Based on Fuzzy Expert System and Multi-agent Metaheuristics	447
Eugene Fedorov, Svitlana Smerichevska, Olga Nechyporenko, Tetyana Utkina, and Yuliia Remyha	
A Testing Methodology for the Internet of Things Affordable IP Cameras	463
Grazyna Dzwigala, Baraq Ghaleb, Talal A. Aldhaheri, Isam Wadhaj, Craig Thomson, and Nasser M. Al-Zidi	
Detecting Equatorial Plasma Bubbles on All-Sky Imager Images Using Convolutional Neural Network	481
Worachai Srisamoodkham, Kazuo Shiokawa, Yuichi Otsuka, Kutubuddin Ansari, and Punyawati Jamjareegulgarn	
A Comparative Study of Traditional Bank A and Digital Bank B from an Organizational Innovation Perspective	489
Easwaramoorthy Rangaswamy, Naresh Nadipilli, and Nishad Nawaz	
A Novel Approach to Improve the Performance of a Classifier Using Visual and Haptic Data	509
Sekhar R. Aravind and K. G. Sreeni	
KGAN: A Generative Adversarial Network Augmented Convolution Neural Network Model for Recognizing Kannada Language Digits	523
H. S. Shrisha, V. Anupama, D. Suresha, and N. Jagadisha	
Sparse Autoencoder-Based Speech Emotion Recognition	533
Vishal Balaji Sivaraman, Sheena Christabel Pravin, K. Surendaranath, A. Vishal, M. Palanivelan, J. Saranya, and L. Priya	
Hyperspectral Image Classification Using Transfer Learning	545
Usha Patel, Smit Patel, and Preeti Kathiria	

Design, Implementation and Performance Analysis of Shift Register Using Reversible Sayem Gate 557
 Ruqaiya Khanam, Gitanjali Mehta, and Vinod Kumar Yadav

Automated Oxygen Blender for Regulation of Oxygen Saturation in Hypoxia Patient 573
 Samruddhi Anikhindi, Shreyas Patil, and Pauroosh Kaushal

An Interleaving Approach to Control Mobile Device and Elements via Screen Buffer and Audio Streaming 587
 Jayavel Kanniappan, Rajesh Kumar Jayavel, and Jithin Gangadharan

Optimization of Algorithms for Simple Polygonizations 603
 Maksim Kovalchuk, Vasyl Tereshchenko, and Yaroslav Tereshchenko

A Super Ensembled and Traditional Models for the Prediction of Rainfall: An Experimental Evaluation of DT Versus DDT Versus RF 619
 Sheikh Amir Fayaz, Majid Zaman, and Muheet Ahmed Butt

Novel User Association Scheme Deployed for the Downlink NOMA Systems 637
 Sunkaraboina Sreenu and Kalpana Naidu

Analyzing the Performance of a Digital Shadow for a Mixed-Model Stochastic System 651
 Philane Tshabalala and Rangith B. Kuriakose

Fuzzy Logic-Based Cluster Head Selection an Underwater Wireless Sensor Network: A Survey 661
 Hetal Panchal and Sachin Gajjar

Improving the Efficiency of Forecasting Sports Events Using a Cascade of Neural Networks 675
 Vasily Meltsov, Alexander Krutikov, and Dmitry Strabykin

Meta-Analysis of Research into the Issue of Brand Building on Social Media as a Subset of e-Business During the COVID-19 Pandemic 685
 L'udovít Nastišin and Richard Fedorko

Nonlinear Direct Adaptive Inverse Control Methodology Based on Volterra Model 703
 Rodrigo Possidônio Noronha

Alerting the Impact of Adversarial Attacks and How to Detect it Effectively via Machine Learning Approach: With Financial and ESG Data 713
 Ook Lee, Hyodong Ha, Hayoung Choi, Hanseon Joo, and Minjong Cheon

Positioning Comparison Using GIM, Klobuchar, and IRI-2016 Models During the Geomagnetic Storm in 2021 725
Worachai Srisamoodkham, Kutubuddin Ansari, and Punyawati Jamjareegulgarn

Wearable Patch Antennas on Fr4, Rogers and Jeans Fabric Substrates for Biomedical Applications 735
Regidi Suneetha and P. V. Sridevi

Puzzling Solid–Liquid Phase Transition of Water (mW) from Free Energy Analysis: A Molecular Dynamics Study 745
Chandan K. Das

Social Media Flood Image Classification Using Transfer Learning with EfficientNet Variants 759
S. M. Jaisakthi and P. R. Dhanya

A Survey-Based Study to Understand Various Aspects of Kanban 771
Anupam Kumar, Nilesh Kumar, Sayani Mondal, and Tarun Biswas

Integrated Bioinformatics Analysis to Identify the Potential Molecular Biomarkers for Neuropathic Pain Among Patient of Lumbar Disc Prolapse and COVID-19 789
Manisha Chaudhary and Veena Puri

Political Optimizer Algorithm for Optimal Location and Sizing of Photovoltaic Distribution Generation in Electrical Distribution Network 807
D. Sreenivasulu Reddy, Varaprasad Janamala, and Pappu Soundarya Lahari

Cyberbullying Detection in Social Media Using Supervised ML and NLP Techniques 817
Karthiga Shankar, A. M. Abirami, K. Indira, C. V. Nisha Angeline, and K. Shubhavya

Investigating the Positioning Capability of GPS and Galileo Constellations Over Indian Sub-continent 829
Devadas Kuna and Naveen Kumar Perumalla

Mapping User-Submitted Short Text Questions to Subjects of Study: A Multinomial Classification Approach 843
Sanjay Singh and Vikram Singh

Physical Layer Security Aspects of D2D Communications in Future Networks 853
Chinnam S. V. Maruthi Rao and Ramakrishna Akella

The Modern Problem of Accessibility and Complexity of Big Data 863
Rodmonga Potapova, Vsevolod Potapov, and Petr Gorbunov

Gray Scale Image Enhancement with CPSO Algorithm for Medical Applications 873
 Mani Kumar Jogi and Y. Srinivasa Rao

Domain-Specific Chatbot Development Using the Deep Learning-Based RASA Framework 883
 Vijay Kumari, Chinmay Gosavi, Yashvardhan Sharma, and Lavika Goel

Pulse Shaper Design for UWB-Based Medical Imaging Applications 897
 M. K. Devika Menon and Joseph Rodrigues

Quantitative Analysis of Transfer Learning in Plant Disease Classification 909
 Pawan Dubey, Vineeta Kumari, Ajay K. Sharma, and Gyanendra Sheoran

Absolute Moment Block Truncation Coding and Singular Value Decomposition-Based Image Compression Scheme Using Wavelet 919
 Rajiv Ranjan and Prabhat Kumar

Cross-Project Defect Prediction by Using Optimized Light Gradient Boosting Machine Algorithm 933
 Shailza Kanwar, Lalit Kumar Awasthi, and Vivek Shrivastava

XGBoost Hyperparameters Tuning by Fitness-Dependent Optimizer for Network Intrusion Detection 947
 Miodrag Zivkovic, Luka Jovanovic, Milica Ivanovic, Nebojsa Bacanin, Ivana Strumberger, and P. Mani Joseph

Temperature Estimation in Multi-Core Processors Using Statistical Approach for Task Scheduling 963
 Leena Ladge and Y. S. Rao

A Generic Ontology and Recovery Protocols for Human–Robot Collaboration Systems 973
 Kamil Skarzynski, Marcin Stepniak, Waldemar Bartyna, and Stanislaw Ambroszkiewicz

Analysis, Modeling, and Forecasting of Day-Ahead Market Prices in Indian Power Exchange 989
 Madhuri Saha and Nitai Pal

Traffic Density Classification for Multiclass Vehicles Using Customized Convolutional Neural Network for Smart City 1015
 Deepak Mane, Ranjeet Bidwe, Bhusan Zope, and Nihar Ranjan

U-shaped Transformer for Enhancing Low-Dose CT Images 1031
 Aswin Unnikrishnan, Amal Pavithran, Arpith G. Naik,
 Abhishek P. Jiju, and P. V. Sudeep

Vehicle-Type Classification Using Capsule Neural Network 1043
 Deepak Mane, Chaitanya Kharche, Shweta Bankar, Swati V. Shinde,
 and Suraksha Suryawanshi

**Trend Prediction of Power Transformers from DGA Data Using
 Artificial Intelligence Techniques** 1053
 A. S. Kunju Lekshmi, Deepa S. Kumar, and K. Sabeena Beevi

**Artificial Intelligence and Machine Learning in the Context
 of E-commerce: A Literature Review** 1067
 Richard Fedorko, Štefan Král, and Igor Fedorko

**Improved Sliding Mode Control for Glucose Regulation of Type
 1 Diabetics Patients Considering Delayed Nonlinear Model** 1083
 Hamed Khodadadi, Hamid Ghadiri, and Ali Dehghani

**Overview and Computational Analysis of PSO Variants
 for Solving Systems of Nonlinear Equations** 1093
 Sérgio Ribeiro and Luiz Guerreiro Lopes

Multimedia Immersion System for Band Jumping Training 1107
 David Rivas-Lalaleo, Marcelo Alvarez-Veintimilla,
 Víctor Bautista-Naranjo, Rosa Granizo-López,
 Pepe Ibañez-Jacome, Hector Lasluisa-Naranjo, Daniel Yanez-Bravo,
 and Bryan Sandoval-Maiza

**Modeling Simulation of SIR PC Infection Spreading Model
 with Fuzzy Parameters** 1119
 M. N. Srinivas, B. S. N. Murthy, M. A. S. Srinivas, and M. Naga Raju

**CatBoost Encoded Tree-Based Model for the Identification
 of Microbes at Genes Level in 16S rRNA Sequence** 1137
 M. Meharunnisa and M. Sornam

Roadkill Avoidance System Using YOLOv5 1157
 Mrunal Mendgudle and Mrunal Shidore

**Automated Cluster Head Selection in Fog-VANET Via Machine
 Learning** 1169
 Anshu Devi, Ramesh Kait, and Virender Ranga

**Neural Network-Based BLDC Motor Drive for Electric Vehicle
 Application** 1181
 Kishore Kumar Pedapenki

Rule Placement-Based Energy-Aware Routing in SDN: Review 1191
 Rachid Ben Said, Sakirin Tam, and Omer Ozgur Tanriover

Strengthening Auto-Feature Engineering of Deep Learning Architecture in Protein–Protein Interaction Prediction	1205
Bhawna Mewara and Soniya Lalwani	
Author Index	1217

Editors and Contributors

About the Editors

Harish Sharma is Associate Professor at Rajasthan Technical University, Kota, in Department of Computer Science and Engineering. He has worked at Vardhaman Mahaveer Open University, Kota, and Government Engineering College Jhalawar. He received his B.Tech. and M.Tech. degree in Computer Engineering from Government Engineering College, Kota, and Rajasthan Technical University, Kota, in 2003 and 2009, respectively. He obtained his Ph.D. from ABV—Indian Institute of Information Technology and Management, Gwalior, India. He is Secretary and one of the founder member of Soft Computing Research Society of India. He is Lifetime Member of Cryptology Research Society of India, ISI, Kolkata. He is Associate Editor of *International Journal of Swarm Intelligence* (IJSI) published by Inderscience. He has also edited special issues of the many reputed journals like *Memetic Computing*, *Journal of Experimental and Theoretical Artificial Intelligence*, *Evolutionary Intelligence*, etc. His primary area of interest is nature-inspired optimization techniques. He has contributed in more than 105 papers published in various international journals and conferences.

Dr. Vivek Shrivastava has approx. 20 years of diversified experience of scholarship of teaching and learning, accreditation, research, industrial, and academic leadership in India, China, and USA. Presently, he is holding the position of Dean Research and Consultancy at National Institute of Technology Delhi. Prior to his academic assignments, he has worked as System Reliability Engineer at SanDisk Semiconductors Shanghai, China, and USA. Dr. Shrivastava has significant industrial experience of collaborating with industry and government organizations at SanDisk Semiconductors, and he has made significant contribution to the design development of memory products. He has contributed to the development and delivery of Five-Year Integrated B.Tech.–M.Tech. Program (Electrical Engineering) and Master’s program (Power

Systems) at Gautam Buddha University, Greater Noida. He has extensive experience academic administration in various capacity of Dean (Research and Consultancy), Dean (Student Welfare), Faculty In-charge (Training and Placement), Faculty In-charge (Library), Nodal Officer (Academics, TEQIP-III), Nodal Officer RUSA, Experts in various committees in AICTE, UGC, etc. Dr. Shrivastava has carried out research and consultancy and attracted significant funding projects from Ministry of Human Resources and Development, Government of India, Board of Research in Nuclear Science (BRNS) subsidiary organization of Bhabha Atomic Research Organization. Dr. Shrivastava has published over 80 journal articles, presented papers at conferences, and has published several chapters in books. He has supervised 05 Ph.D. and 16 Master's students and currently supervising several Ph.D. students. His diversified research interests are in the areas of reliability engineering, renewable energy, and conventional power systems which include wind, photovoltaic (PV), hybrid power systems, distributed generation, grid integration of renewable energy, power systems analysis, and smart grid. Dr. Shrivastava is Editor/Associate Editor of the Journals, *International Journal of Swarm Intelligence (IJSI)*, and *International Journal of System Assurance Engineering and Management*. He is Fellow of the Institution of Engineers (India) and Senior Member of the Institute of Electrical and Electronics Engineers (IEEE).

Dr. Kusum Kumari Bharti is Assistant Professor at PDPM IITDM Jabalpur. Dr. Bharti has obtained her Ph.D. in Computer Science and Engineering from ABV-IIIITM, Gwalior. She has guided 06 M.Tech. and presently guiding 02 Ph.D. students and 05 M.Tech. students. She has published more than 12 journal and conference papers in the area of text clustering, data mining, online social network, and soft computing. She has been Active Member of many organizing committees of various conferences, workshops, and faculty development program. Her research areas include machine learning, data mining, machine translation, online social network, and soft computing.

Dr. Lipo Wang received the Bachelor's degree from National University of Defense Technology (China) and Ph.D. from Louisiana State University (USA). He is presently on the faculty of the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. His research interest is artificial intelligence with applications to image/video processing, biomedical engineering, and data mining. He has 330+ publications, a US patent in neural networks, and a patent in systems. He has co-authored 2 monographs and (co-)edited 15 books. He has 8000+ Google Scholar citations, with H-index 43. He was Keynote Speaker for 36 international conferences. He is/was Associate Editor/Editorial Board Member of 30 international journals, including 4 IEEE Transactions, and Guest Editor for 10 journal special issues. He was Member of the Board of Governors of the International Neural Network Society, IEEE Computational Intelligence Society (CIS), and the IEEE Biometrics Council. He served as CIS Vice President for Technical Activities and Chair of Emergent Technologies Technical Committee, as well as Chair of Education Committee of the IEEE Engineering in Medicine and Biology

Society (EMBS). He was President of the Asia-Pacific Neural Network Assembly (APNNA) and received the APNNA Excellent Service Award. He was founding Chair of both the EMBS Singapore Chapter and CIS Singapore Chapter. He serves/served as Chair/Committee Member of over 200 international conferences.

Contributors

Abirami A. M. Department of Information Technology, Thiagarajar College of Engineering, Madurai, India

Ahmed Zeyad A. T. Department of Computer Science, Dr. Babasaheb Ambedkar Marathwada University Aurangabad, Aurangabad, India

Akella Ramakrishna Department of Electronics & Communication Engineering, KL University, Vaddeswaram, India

Akhil GBS Department of CSE, PES University Bangalore, Bangalore, India

Al-Zidi Nasser M. Faculty of Administrative and Computer Sciences, Albaydha University, Albaydha, Yemen

Aldhaheri Talal A. Faculty of Administrative and Computer Sciences, Albaydha University, Albaydha, Yemen

Alelyani Salem Center for Artificial Intelligence, College of Computer Science, King Khalid University, Abha, Kingdom of Saudi Arabia

Alvarez-Veintimilla Marcelo Departamento de Eléctrica y Electrónica, Universidad de las Fuerzas Armadas - ESPE Sangolquí, Sangolquí, Ecuador

Ambhaikar Asha Department of CSE, Kalinga University, Raipur, India

Ambroszkiewicz Stanislaw Siedlce University of Natural Sciences and Humanities, Siedlce, Poland;

Institute of Computer Science, Polish Academy of Sciences, Warsaw, Poland

Angeline C. V. Nisha Department of Information Technology, Thiagarajar College of Engineering, Madurai, India

Anikhindi Samruddhi MIT School of Bioengineering Sciences and Research, MIT Art, Design and Technology University, Pune, India

Ansari Kutubuddin Integrated Geoinformation (IntGeo) Solution Private Limited, New Delhi, India

Anupama V. Canara Engineering College, Visveswaraya Technological University, Benjanapadavu, Bantwal, India

Anusha M. Computer Science and Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, Tamil Nadu, India

Anushka Xavier K. Department of Computer Science and Engineering, CHRIST (Deemed to Be University), Kengeri Campus, Bangalore, India

Aravind Sekhar R. Department of Electronics and Communication, College of Engineering Trivandrum, Trivandrum, India

Aruna R. Computer Science and Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, Tamil Nadu, India

Ashwanth V. Muthoot Institute of Technology and Science, Varikoli, Kerala, India

Awasthi Lalit Kumar National Institute of Technology Hamirpur, Hamirpur, Himachal Pradesh, India

Bacanin Nebojsa Department of Informatics and Computing, Singidunum University, Belgrade, Serbia

Bahore Pranjal Maulana Azad National Institute of Technology Bhopal, Bhopal, MP, India

Bankar Shweta JSPM's Rajarshi Shahu College of Engineering, Pune, Maharashtra, India

Bartyna Waldemar Siedlce University of Natural Sciences and Humanities, Siedlce, Poland

Bautista-Naranjo Víctor Departamento de Eléctrica y Electrónica, Universidad de las Fuerzas Armadas - ESPE Sangolquí, Sangolquí, Ecuador

Beevi K. Sabeena Department of EEE, TKM College of Engineering Kollam, Kollam, Kerala, India

Bhattacharjee Baibaswata Ramananda College, Bankura, West Bengal, India

Bhaumik Parama Jadavpur University, Kolkata, India

Bhavisha G. Computer Science and Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, Tamil Nadu, India

Bidwe Ranjeet Pune Institute of Computer Technology, Pune, Maharashtra, India

Biswas Tarun Department of CSE, National Institute of Technology, Sikkim, India

Bosu Surajit Bankura Sammilani College, Bankura, West Bengal, India

Butt Muheet Ahmed Department of Computer Sciences, University of Kashmir, Srinagar, India

Chandana G. Computer Science and Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, Tamil Nadu, India

Chaudhary Manisha Centre for Systems Biology and Bioinformatics (U.I.E.A.S.T), Panjab University, Chandigarh, India

Chaurasiya Rahul Maulana Azad National Institute of Technology Bhopal, Bhopal, MP, India

Cheon Minjong Department of Information Systems, Hanyang University, Seoul, South Korea

Chetradevee S. L. Department of Computer Science and Engineering, CHRIST (Deemed to Be University), Kengeri Campus, Bangalore, India

Chinnadurai M. Department of Computer Science and Engineering, E.G.S. Pillay Engineering College, Nagapattinam, Tamil Nadu, India

Choi Hayoung Department of Information Systems, Hanyang University, Seoul, South Korea

Correia Noélia CEOT, University of Algarve, Faro, Portugal;
Faculty of Science and Technology, University of Algarve, Faro, Portugal

Das Chandan K. Department of Chemical Engineering, National Institute of Technology Rourkela, Rourkela, India

Das Sayan Central Institute of Technology Kokrajhar, Kokrajhar, Assam, India

Dehghani Ali National Laboratory of Pattern Recognition, Institute of Automation of Chinese Academy of Sciences, Beijing, China;
University of Chinese Academy of Sciences, Beijing, China

Dehraj Pooja Computer Science and Engineering Department, Noida Institute of Engineering and Technology, Greater Noida, India

Devi Anshu Kurukshetra University, Kurukshetra, Haryana, India

Devika Menon M. K. CHRIST (Deemed to Be University), Bengaluru, India

Dhanya P. R. School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, India

Dheeraj Saam Prasanth Department of Electronics and Communication Engineering, Vel Tech Rangarajan Dr, Sagunthala R&D Institute of Science and Technology, Chennai, Tamil Nadu, India

Dholay Surekha Bharatiya Vidya Bhavan's Sardar Patel Institute of Technology (SPIT), Mumbai, India

du Plessis Shani CEOT, University of Algarve, Faro, Portugal

Dubey Pawan National Institute of Technology Delhi, New Delhi, Delhi, India

Durgadevi M. Department of Computer Science and Engineering, College of Engineering and Technology, SRM Institute of Science and Technology, Vadapalani Campus, Chennai, Tamil Nadu, India

Dzwigala Grazyna School of Computing at Edinburgh, Napier University, Edinburgh, UK

Esther Rani P. Department of Electronics and Communication Engineering, Vel Tech Rangarajan Dr, Sagunthala R&D Institute of Science and Technology, Chennai, Tamil Nadu, India

Fayaz Sheikh Amir Department of Computer Sciences, University of Kashmir, Srinagar, India

Fedorko Igor Faculty of Management and Business, University of Presov, Prešov, Slovakia

Fedorko Richard Faculty of Management and Business, University of Prešov, Prešov, Slovakia

Fedorov Eugene Cherkasy State Technological University, Cherkasy, Ukraine

Gadgil Anusha Department of ENTC, Symbiosis International Deemed University, Pune, Maharashtra, India

Gajjar Pranshav Institute of Technology, Nirma University, Ahmedabad, India

Gajjar Sachin Department of Electronics and Communication Engineering, Institute of Technology, Nirma University, Ahmedabad, India

Gangadharan Jithin Intelligence & IoT, Samsung R&D Institute, Bengaluru, India

Garg Sharad Computer Science and Engineering Department, Noida Institute of Engineering and Technology, Greater Noida, India

Gautam Priya University Departments, Rajasthan Technical University, Kota, Rajasthan, India

Ghadiri Hamid Department of Electrical Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran

Ghaleb Baraq School of Computing at Edinburgh, Napier University, Edinburgh, UK

Goel Lavika Malaviya National Institute of Technology, Jaipur, India

Gohad Mihir Department of Mechanical, Symbiosis International Deemed University, Pune, Maharashtra, India

Gokula Krishnan V. CSIT Department, CVR College of Engineering, Hyderabad, Telanagana, India

Gorbunov Petr Institute of Applied and Mathematical Linguistics, Moscow State Linguistic University, Moscow, Russia

Gosavi Chinmay Birla Institute of Technology and Science, Pilani, India

Granizo-López Rosa Departamento de Eléctrica y Electrónica, Universidad de las Fuerzas Armadas - ESPE Sangolquí, Sangolquí, Ecuador

Gurupandi D. ECE Department, Panimalar Institute of Technology, Chennai, Tamil Nadu, India

Ha Hyodong Department of Information Systems, Hanyang University, Seoul, South Korea

Hai Pham Ngoc Computer Science Department, FPT University, Hanoi, Vietnam

Hieu Hoang Trung Computer Science Department, FPT University, Hanoi, Vietnam

Hirani Ebrahim Bharatiya Vidya Bhavan's Sardar Patel Institute of Technology (SPIT), Mumbai, India

Hung Phan Duy Computer Science Department, FPT University, Hanoi, Vietnam

Huy Le Xuan Computer Science Department, FPT University, Hanoi, Vietnam

Ibañez-Jacome Pepe Departamento de Eléctrica y Electrónica, Universidad de las Fuerzas Armadas - ESPE Sangolquí, Sangolquí, Ecuador

Indira K. Department of Information Technology, Thiagarajar College of Engineering, Madurai, India

Ivanovic Milica Department of Informatics and Computing, Singidunum University, Belgrade, Serbia

Jagadisha N. Canara Engineering College, Visveswaraya Technological University, Benjanapadavu, Bantwal, India

Jaisakthi S. M. School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, India

Jamjareegulgarn Punyawit Prince of Chumphon Campus, King Mongkut's Institute of Technology Ladkrabang, Chumphon, Thailand

Janamala Varaprasad Department of Electrical and Electronics Engineering, School of Engineering and Technology, Christ (Deemed to Be University), Bengaluru, Karnataka, India

Jayapandian N. Department of Computer Science and Engineering, CHRIST (Deemed to Be University), Kengeri Campus, Bangalore, India

Jayavel Rajesh Kumar Intelligence & IoT, Samsung R&D Institute, Bengaluru, India

Jiju Abhishek P. National Institute of Technology Calicut, Kozhikode, Kerala, India

Jogi Mani Kumar Department of Instrument Technology, Andhra University, Visakhapatnam, A.P., India

Joo Hanseon Department of Information Systems, Hanyang University, Seoul, South Korea

Joseph P. Mani Department of Mathematics & Computer Science, Modern College of Business and Science, Muscat, Sultanate of Oman

Jovanovic Luka Department of Informatics and Computing, Singidunum University, Belgrade, Serbia

Kait Ramesh Kurukshetra University, Kurukshetra, Haryana, India

Kanniappan Jayavel Intelligence & IoT, Samsung R&D Institute, Bengaluru, India

Kanwar Shailza National Institute of Technology Delhi, Delhi, New Delhi, India

Kar Biswajit Central Institute of Technology Kokrajhar, Kokrajhar, Assam, India

Karthika S. Department of Computer Science and Engineering, College of Engineering and Technology, SRM Institute of Science and Technology, Vadapalani Campus, Chennai, Tamil Nadu, India

Kathiria Preeti Institute of Technology, Nirma University, Ahmedabad, India

Kaushal Pauroosh MIT School of Bioengineering Sciences and Research, MIT Art, Design and Technology University, Pune, India

Khanam Ruqaiya Computer Science and Engineering, Center for Artificial Intelligence in Medicine, Imaging and Forensic, Sharda University, Greater Noida, India

Kharche Chaitanya JSPM's Rajarshi Shahu College of Engineering, Pune, Maharashtra, India

Khodadadi Hamed Department of Electrical Engineering, Khomeinishahr Branch, Islamic Azad University, Isfahan, Iran

Kotecha Ketan Faculty of Engineering, Symbiosis International Deemed University, Pune, Maharashtra, India

Kovalchuk Maksim Taras Shevchenko National University of Kyiv, Kyiv, Ukraine

Krutikov Alexander Vyatka State University, Kirov, Russia

Kráľ Štefan Faculty of Management and Business, University of Presov, Prešov, Slovakia

Kumar Anupam Department of CSE, National Institute of Technology, Sikkim, India

Kumar Deepa S. Power Networks Demonstration Centre, University of Strathclyde, Glasgow, Scotland

Kumar Manoj Department of Computer Science and Engineering, Netaji Subhas University of Technology, East Campus (Formerly Ambedkar Institute of Advanced Communication Technologies and Research), Delhi, India

Kumar Nidarshan Department of CSE, PES University Bangalore, Bangalore, India

Kumar Nilesh Energy Institute, Centre of Rajiv Gandhi Institute of Petroleum Technology, Bengaluru, Karnataka, India

Kumar Prabhat NIT Patna, Patna, India

Kumar Tarun Dr. B R Ambedkar National Institute of Technology, Jalandhar, Punjab, India

Kumari Vijay Birla Institute of Technology and Science, Pilani, India

Kumari Vineeta National Institute of Technology Delhi, New Delhi, Delhi, India

Kuna Devadas Advanced GNSS Research Laboratory, Department of Electronics and Communication Engineering, University College of Engineering, Osmania University, Hyderabad, India

Kuriakose Rangith B. Central University of Technology, Bloemfontein, Free State, South Africa

Ladge Leena SIES Graduate School of Technology, Mumbai, India

Lahari Pappu Soundarya Department of Electrical and Electronics Engineering, School of Engineering and Technology, Christ (Deemed to Be University), Bengaluru, Karnataka, India

Lakshmi Akshitha Y. Computer Science and Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, Tamil Nadu, India

Lalwani Soniya Department of Computer Science and Engineering, Career Point University, Kota, India

Lasluisa-Naranjo Hector Departamento de Eléctrica y Electrónica, Universidad de las Fuerzas Armadas - ESPE Sangolquí, Sangolquí, Ecuador

Lee Ook Department of Information Systems, Hanyang University, Seoul, South Korea

Lekshmi A. S. Kunju Department of EEE, TKM College of Engineering Kollam, Kollam, Kerala, India

Lohitha B. Computer Science and Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, Tamil Nadu, India

Lopes Luiz Guerreiro Faculty of Exact Sciences and Engineering, University of Madeira, Funchal, Madeira Is., Portugal

López-Fierro Sariah Technological Research Department, Soluciones Wandarina S. A., Guayaquil, Ecuador

Magotra Varun Bharatiya Vidya Bhavan's Sardar Patel Institute of Technology (SPIT), Mumbai, India

Maharaj Yadav Discipline of Electrical, Electronic and Computer Engineering, University of KwaZulu-Natal, Durban, South Africa

Mahyoub Ezzaldden Department of Computer Science, Dr. Babasaheb Ambedkar Marathwada University Aurangabad, Aurangabad, India

Mallesham G. Department of Electrical Engineering, University College of Engineering, Osmania University, Hyderabad, Telangana, India

Mamatha H. R. Department of CSE, PES University Bangalore, Bangalore, India

Mane Deepak JSPM's Rajarshi Shahu College of Engineering, Pune, Maharashtra, India

Maruthi Rao Chinnam S. V. Department of Electronics & Communication Engineering, KL University, Vaddeswaram, India

Meharunnisa M. Department of BCA, Ethiraj College For Women, Chennai, Tamil Nadu, India

Mehta Gitanjali Electrical and Electronics Engineering, Galgotias University, Greater Noida, India

Mehta Naishadh Institute of Technology, Nirma University, Ahmedabad, India

Mehta Shilpa ECE, SoE, Presidency University, Bangalore, India

Mehta Vedant Bharatiya Vidya Bhavan's Sardar Patel Institute of Technology (SPIT), Mumbai, India

Meltsov Vasily Vyatka State University, Kirov, Russia

Mendes Bruno CEOT, University of Algarve, Faro, Portugal

Mendgudle Mrunal Vishwakarma Institute of Technology, Pune, India

Mewara Bhawna Department of Computer Science and Engineering, Career Point University, Kota, India

Mondal Sayani Indian Institute of Technology, Kharagpur, West Bengal, India

Mukunthan B. Department of Computer Science, Jairams Arts and Science College, Karur, India;
Bharathidasan University, Tiruchirappalli, India

Murthy B. S. N. Department of Mathematics, Aditya College of Engineering and Technology, Surampalem, A.P., India

Nadipilli Naresh Credit Agricole Corporate Investment Bank, Singapore, Singapore

Naga Raju M. Department of Mathematics, Aditya College of Engineering and Technology, Surampalem, A.P., India

Nagadeepa N. Bharathidasan University, Tiruchirappalli, India;
Sri Sarada Niktan College of Science for Women, Karur, India

Nagpal Sushama Netaji Subhas University of Technology, New Delhi, India

Naidu Kalpana Department of ECE, National Institute of Technology, Warangal, India

Naik Arpith G. National Institute of Technology Calicut, Kozhikode, Kerala, India

Naim Arshi Department of Information Systems, King Khalid University, Abha, Kingdom of Saudi Arabia

Nastišin L'udovít Faculty of Management and Business, University of Prešov, Prešov, Slovakia

Nawaz Nishad Department of Business Management, College of Business Administration, Kingdom University, Riffa, Kingdom of Bahrain

Nechyporenko Olga Cherkasy State Technological University, Cherkasy, Ukraine

Ngoc Bui Thi Bich Computer Science Department, FPT University, Hanoi, Vietnam

Nimbhore Sunil Department of Computer Science, Dr. Babasaheb Ambedkar Marathwada University Aurangabad, Aurangabad, India

Noronha Rodrigo Possidônio Department of Electrical Engineering, Federal Institute of Maranhão, Imperatriz, MA, Brazil

Otsuka Yuichi Institute for Space-Earth Environmental Research, Nagoya University, Nagoya, Japan

Pal Nitai Department of Electrical Engineering, Indian Institute of Technology (ISM), Dhanbad, Jharkhand, India

Palanivelan M. Department of Electronics and Communication Engineering, Rajalakshmi Engineering College, Chennai, India

Paliwal Shreyansh Maulana Azad National Institute of Technology Bhopal, Bhopal, MP, India

Panchal Hetal Department of Electronics and Communication Engineering, Institute of Technology, Nirma University, Ahmedabad, India

Passos Dário CEOT, University of Algarve, Faro, Portugal

Patel Smit Institute of Technology, Nirma University, Ahmedabad, India

Patel Usha Institute of Technology, Nirma University, Ahmedabad, India

Patil Shreyas MIT School of Bioengineering Sciences and Research, MIT Art, Design and Technology University, Pune, India

Paul Swarna Kamal Jadavpur University, Kolkata, India

Pavithran Amal National Institute of Technology Calicut, Kozhikode, Kerala, India

Pedapenki Kishore Kumar Electrical and Electronics Engineering, Jain (Deemed to be University), Bengaluru, India

Perumalla Naveen Kumar Advanced GNSS Research Laboratory, Department of Electronics and Communication Engineering, University College of Engineering, Osmania University, Hyderabad, India

Potapov Vsevolod Centre of New Technologies for Humanities, Lomonosov Moscow State University, Moscow, Russia

Potapova Rodmonga Institute of Applied and Mathematical Linguistics, Moscow State Linguistic University, Moscow, Russia

Pravin Sheena Christabel Department of Electronics and Communication Engineering, Rajalakshmi Engineering College, Chennai, India

Priya L. Department of Information Technology, Rajalakshmi Engineering College, Chennai, India

Puri Veena Centre for Systems Biology and Bioinformatics (U.I.E.A.S.T), Panjab University, Chandigarh, India

Ragam Anirudh V. Department of CSE, PES University Bangalore, Bangalore, India

Raich Devashri Department of CSE, Kalinga University, Raipur, India

Ramabalan S. Department of Mechanical Engineering, E.G.S. Pillay Engineering College, Nagapattinam, Tamil Nadu, India

Ranga Virender Delhi Technological University, Delhi, New Delhi, India

Rangaswamy Easwaramoorthy Amity Global Institute, Singapore, Singapore

Rani Rajneesh Dr. B R Ambedkar National Institute of Technology, Jalandhar, Punjab, India

Ranjan Nihar JSPM's Rajarshi Shahu College of Engineering, Pune, Maharashtra, India

Ranjan Rajiv BIT Sindri, Dhanbad, India

Rao E. Jagadeeswara Department of Electronics and Communication Engineering, Vignan's Institute of Engineering for Women, Visakhapatnam, AP, India

Rao Y. S. Sardar Patel Institute of Technology, Mumbai, India

Rashid Hussain Mohammad Center for Artificial Intelligence, College of Computer Science, King Khalid University, Abha, Kingdom of Saudi Arabia

Rautela Dipanshu Maulana Azad National Institute of Technology Bhopal, Bhopal, MP, India

Remyha Yuliia International European University, Kyiv, Ukraine

Ribeiro Sérgio Postgraduate Programme in Informatics Engineering, University of Madeira, Funchal, Madeira Is., Portugal

Rivas-Lalaleo David Departamento de Eléctrica y Electrónica, Universidad de las Fuerzas Armadas - ESPE Sangolquí, Sangolquí, Ecuador

Rodrigues Joseph CHRIST (Deemed to Be University), Bengaluru, India

Sah Mukund Prasad Dr. B R Ambedkar National Institute of Technology, Jalandhar, Punjab, India

Saha Madhuri Department of Electrical Engineering, Indian Institute of Technology (ISM), Dhanbad, Jharkhand, India

Said Rachid Ben Graduate School of Natural and Applied Science, Ankara University, Ankara, Turkey

Saleh Alsaqer Mohammed Center for Artificial Intelligence, College of Computer Science, King Khalid University, Abha, Kingdom of Saudi Arabia

Sandoval-Maiza Bryan Departamento de Eléctrica y Electrónica, Universidad de las Fuerzas Armadas - ESPE Sangolquí, Sangolquí, Ecuador

Sankar K. CSE Department, CVR College of Engineering, Hyderabad, Telangana, India

Saranya J. Department of Electronics and Communication Engineering, Rajalakshmi Engineering College, Chennai, India

Saravanan T. Department of Computer Science, Jairams Arts and Science College, Karur, India;
Bharathidasan University, Tiruchirappalli, India

Sathiya V. Department of Electronics and Communication Engineering, E.G.S. Pillay Engineering College, Nagapattinam, Tamil Nadu, India

Shah Pooja Institute of Technology, Nirma University, Ahmedabad, India

Shankar Karthiga Department of Information Technology, Thiagarajar College of Engineering, Madurai, India

Sharma Ajay K. National Institute of Technology Delhi, New Delhi, Delhi, India

Sharma Yashvardhan Birla Institute of Technology and Science, Pilani, India

Sheoran Gyanendra National Institute of Technology Delhi, New Delhi, Delhi, India

Shidore Mrunal Vishwakarma Institute of Technology, Pune, India

Shinde Swati V. Pimpri Chinchwad College of Engineering, Pune, Maharashtra, India

Shiokawa Kazuo Institute for Space-Earth Environmental Research, Nagoya University, Nagoya, Japan

Shrisha H. S. Canara Engineering College, Visveswaraya Technological University, Benjanapadavu, Bantwal, India

Shrivastava Ritvik Computer Science and Engineering Department, Noida Institute of Engineering and Technology, Greater Noida, India

Shrivastava Vivek National Institute of Technology Delhi, Delhi, New Delhi, India

Shubhavya K. Department of Information Technology, Thiagarajar College of Engineering, Madurai, India

Shyamala Devi M. Computer Science and Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, Tamil Nadu, India

Singh Jyotsna Netaji Subhas University of Technology, New Delhi, India

Singh Sanjay Department of Computer Science and Engineering, Chaudhary Devi Lal University, Sirsa, India

Singh Tanu University School of Information, Communication and Technology, Guru Gobind Singh Indraprastha University, Delhi, India

Singh Vikram Department of Computer Science and Engineering, Chaudhary Devi Lal University, Sirsa, India

Singh Yashpal Department of CSE, Kalinga University, Raipur, India

Singla Neetu Netaji Subhas University of Technology, New Delhi, India

Siva Kumar CH. Department of Electrical Engineering, University College of Engineering, Osmania University, Hyderabad, Telangana, India

Sivaraman Vishal Balaji Department of Electronics and Communication Engineering, Rajalakshmi Engineering College, Chennai, India

Skarzynski Kamil Siedlce University of Natural Sciences and Humanities, Siedlce, Poland

Smerichevska Svitlana National Aviation University, Kyiv, Ukraine

Sornam M. Department of Computer Science, University of Madras, Chennai, Tamil Nadu, India

Sreedevi Sneha Muthoot Institute of Technology and Science, Varikoli, Kerala, India

Sreeni K. G. Department of Electronics and Communication, College of Engineering Trivandrum, Trivandrum, India

Sreenivasulu Reddy D. Department of Electrical and Electronics Engineering, School of Engineering and Technology, Christ (Deemed to Be University), Bengaluru, Karnataka, India

Sreenu Sunkaraboina Department of ECE, National Institute of Technology, Warangal, India

Sreerama Murthy K. IT Department, Sreenidhi Institute of Science and Technology, Hyderabad, Telangana, India

Sridevi P. V. Andhra University College of Engineering (A), Andhra University, Visakhapatnam, Andhra Pradesh, India

Srinivasa Rao Y. Department of Instrument Technology, Andhra University, Visakhapatnam, A.P., India

Srinivas M. A. S. Department of Mathematics, Jawaharlal Nehru Technological University, Hyderabad, Telangana, India

Srinivas M. N. Department of Mathematics, School of Advanced Sciences, Vellore Institute of Technology, Vellore, Tamil Nadu, India

Srisamoodkham Worachai Faculty of Agricultural and Industrial Technology, Phetchabun Rajabhat University, Sadiang, Phetchabun, Thailand

Stepniak Marcin Siedlce University of Natural Sciences and Humanities, Siedlce, Poland

Strabykin Dmitry Vyatka State University, Kirov, Russia

Strumberger Ivana Department of Informatics and Computing, Singidunum University, Belgrade, Serbia

Sudeep P. V. National Institute of Technology Calicut, Kozhikode, Kerala, India

Sudhkar J. Department of Electronics and Communication Engineering, Vignan's Institute of Engineering for Women, Visakhapatnam, AP, India

Suneetha Regidi Andhra University College of Engineering (A), Andhra University, Visakhapatnam, Andhra Pradesh, India

Surendaranath K. Department of Electronics and Communication Engineering, Rajalakshmi Engineering College, Chennai, India

Suresha D. AJ Institute of Engineering and Technology, Visveswaraya Technological University, Mangaluru, India

Suryawanshi Suraksha JSPM's Rajarshi Shahu College of Engineering, Polytechnic, Pune, Maharashtra, India

Tam Sakirin Graduate School of Natural and Applied Science, Ankara University, Ankara, Turkey;
Faculty of Science and Information Technology, Phnom Penh International University, Phnom Penh, Cambodia

Tanriover Omer Ozgur Graduate School of Natural and Applied Science, Ankara University, Ankara, Turkey

Tapamo Jules-Raymond Discipline of Electrical, Electronic and Computer Engineering, University of KwaZulu-Natal, Durban, South Africa

Taparia Rajashree University Departments, Rajasthan Technical University, Kota, Rajasthan, India

Tawfik Mohammed Department of Computer Science, Dr. Babasaheb Ambedkar Marathwada University Aurangabad, Aurangabad, India

Tereshchenko Vasyi Taras Shevchenko National University of Kyiv, Kyiv, Ukraine

Tereshchenko Yaroslav Taras Shevchenko National University of Kyiv, Kyiv, Ukraine

Thakur Arjun Department of ENTC, Symbiosis International Deemed University, Pune, Maharashtra, India

Thomson Craig School of Computing at Edinburgh, Napier University, Edinburgh, UK

Tilak Raju D. Department of Electronics and Communication Engineering, Vignan's Institute of Engineering for Women, Visakhapatnam, A.P., India

Tshabalala Philane Central University of Technology, Bloemfontein, Free State, South Africa

Ukani Vijay Institute of Technology, Nirma University, Ahmedabad, India

Unnikrishnan Aswin National Institute of Technology Calicut, Kozhikode, Kerala, India

Utkina Tetyana Cherkasy State Technological University, Cherkasy, Ukraine

Veni T. National Institute of Technology, Calicut, Kozhikode, India

Venkata Rao K. CSE Department, Guru Nanak Institution Technical Campus, Hyderabad, Telangana, India

Vishal A. Department of Electronics and Communication Engineering, Rajalakshmi Engineering College, Chennai, India

Viswanathasarma Ch. CSE Department, Vignan's Institute of Information Technology, Visakhapatnam, India

Wadhaj Isam School of Computing at Edinburgh, Napier University, Edinburgh, UK

Walambe Rahee Department of ENTC, Symbiosis International Deemed University, Pune, Maharashtra, India

Yadav Vinod Kumar Electrical Engineering Department, Delhi Technological University, Delhi, India

Yanez-Bravo Daniel Departamento de Eléctrica y Electrónica, Universidad de las Fuerzas Armadas - ESPE Sangolquí, Sangolquí, Ecuador

Zaman Majid Directorate of IT & SS, University of Kashmir, Srinagar, India

Zivkovic Miodrag Department of Informatics and Computing, Singidunum University, Belgrade, Serbia

Zope Bhusan Pune Institute of Computer Technology, Pune, Maharashtra, India

A Design of Frequency Encoded Dibit-Based Inhibitor Logic Using Reflective Semiconductor Optical Amplifier with Simulative Verification



Surajit Bosu and Baibaswata Bhattacharjee

Abstract Optical signal communication and computation are very popular these days. Therefore, optical-based different logic gates, combinational circuits, devices, etc. are developed by scientists. In this article, the design of all-optical frequency-encoded inhibitor logic is devised using dibit-based logic, reflective semiconductor optical amplifier (RSOA), and add/drop multiplexer (ADM). This proposed design can perform at ultra-high speed with low noise because of the property of RSOA. The devised design decreases the probability of bit error by blocking the false dibit inputs with the help of debit-checking units. The benefit of the frequency encoding scheme is that the frequency-encoded signal propagates in long range because the frequency is constant in the case of reflection, refraction, absorption, etc. MATLAB (2018a) software has been used to simulate the devised design, and simulated outputs accurately verify the truth table of the inhibitor logic.

Keywords Optical communication · Reflective semiconductor optical amplifier · Inhibitor logic · Frequency encoding · Dibit-based logic

1 Introduction

Photon has important properties like a high degree of parallelism, high storage capacity, although it has high speed as well as low noise. Based on these properties, photons may bear a great advantage in the field of computation and data processing. For data transmission, the encoding of signal data is very necessary. Therefore, a number of encoding processes based on intensity [1], polarization [2, 3], phase [4, 5], hybrid [6], spatial [7, 8], frequency [9–13], etc. are developed for optical communication. Frequency encoding is very popular with respect to other encoding processes because the frequency is invariant in the case of reflection, refraction, absorption, etc. So, it

S. Bosu (✉)

Bankura Sammilani College, Bankura, West Bengal, India
e-mail: surajitbosu7@gmail.com

B. Bhattacharjee

Ramananda College, Bishnupur, Bankura, West Bengal, India

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
H. Sharma et al. (eds.), *Communication and Intelligent Systems*, Lecture Notes in Networks and Systems 461, https://doi.org/10.1007/978-981-19-2130-8_1

Table 1 Proposed inhibitor logic truth table

Dibit input						Dibit output		
A'	A''	Logic state	B'	B''	Logic state	Y'	Y''	Logic state
ν_1	ν_2	0	ν_1	ν_2	0	ν_1	ν_2	0
ν_1	ν_2	0	ν_2	ν_1	1	ν_2	ν_1	1
ν_2	ν_1	1	ν_1	ν_2	0	ν_1	ν_2	0
ν_2	ν_1	1	ν_2	ν_1	1	ν_1	ν_2	0

maintains its property throughout the transmission of data and thereby decreases the probability of bit error problem [14]. In frequency encoding, two distinct digital logic states ('0' and '1') are represented by two distinct frequencies like as ν_1 (for '0') and ν_2 (for '1'). Mukhopadhyay [15] first proposed the dibit-based representation scheme. According to this scheme, two consecutive bit positions are taken to define a bit. The digital logic states '0' and '1' are indicated by $\langle 0 \rangle \langle 1 \rangle$ and $\langle 1 \rangle \langle 0 \rangle$, respectively. In frequency encoding, digital logic states '0' and '1' are indicated by $\langle \nu_1 \rangle \langle \nu_2 \rangle$ and $\langle \nu_2 \rangle \langle \nu_1 \rangle$. If a NOT circuit is connected to one input terminal (say, A) of two-input AND gate, the system acts as an inhibitor or anti-coincidence circuit. The truth table of a two-input inhibitor circuit is given in Table 1. Ghosh et al. [16] have introduced a wavelength-encoded inhibitor logic operation using nonlinear material (NLM) with a phase conjugation system (PCS). High-intensity light beams are needed to operate and also it is tough enough to sustain the state of phase and intensity of the beams. In this communication, a design of the frequency-encoded inhibitor logic using dibit-based logic, ADM, and RSOA is devised. This design gives real-time operation and consumes low power because of RSOA [17, 18]. This design also decreases the bit error problems in long-range transmission due to frequency encoding. Dibit concept ensures a high degree of parallelism [16, 19]. Besides all of these, MATLAB (2018a) software has been used to simulate the devised design.

The remaining part is organized as the following: Sect. 2 describes the working function of RSOA and ADM. Section 3 introduces the operational scheme of the inhibitor logic design in detail. Section 4 provides the simulation of devised design. Results and discussion are given in Sect. 5. In Sect. 6, a conclusion of this work is given.

2 Operational Principle of RSOA and ADM

In this section, the basic components of frequency-encoded inhibitor logic are explained. So, the working principle of ADM and RSOA is discussed.

An anti-reflecting (AR) coating and a highly reflecting coating (HR) are placed in the two facets of the RSOA [19, 20]. RSOA has a high gain with low noise. It is

Fig. 1 Schematic diagram of RSOA

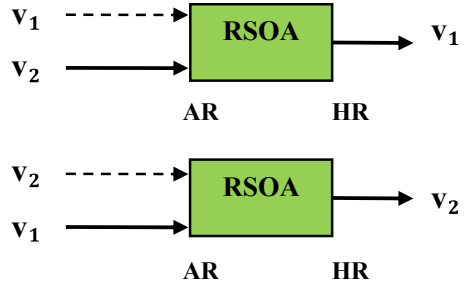
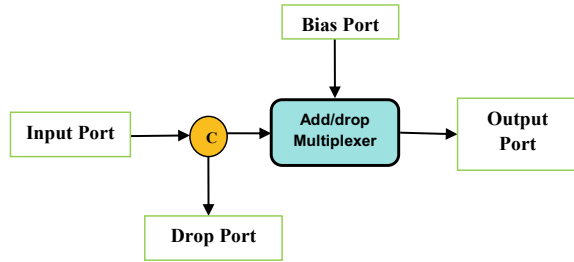


Fig. 2 Schematic diagram of ADM



popular in a passive optical network (PON) [17]. When a probe (weak) input signal and pump (strong) signal are injected into RSOA, then probe frequency with strong power is obtained at the output terminal. In the range of 5–20 dBm, saturation power may be used to operate the RSOA in the proposed design [9, 21]. Figure 1 represents RSOA’s schematic diagram. Here, the dotted line and solid line represent the probe and pump signals, respectively.

ADM is very popular as a frequency routing device. When same frequency signals are introduced at the bias and input port, then ADM reflects the input signal to the drop port. But when two signals with different frequency are introduced in the ADM as input and biasing signal, then ADM selects the input signal at the output and no signal at the drop port [19]. It is given in Fig. 2.

3 Proposed Scheme of Operation of Inhibitor Logic

In the introduction section, we explained the dibit-based logic system. In this section, the proposed scheme of inhibitor logic operation is introduced. Since this design is frequency encoded, so the dibit logic (0) (1) and (1) (0) represent in terms of frequency as $\langle v_1 \rangle \langle v_2 \rangle$ and $\langle v_2 \rangle \langle v_1 \rangle$. The operational scheme of the proposed inhibitor logic device is elucidated in the following illustration. A schematic diagram of this design is given in Fig. 3. Here, four cases are described to explain the operation of the devised design which are as follows.

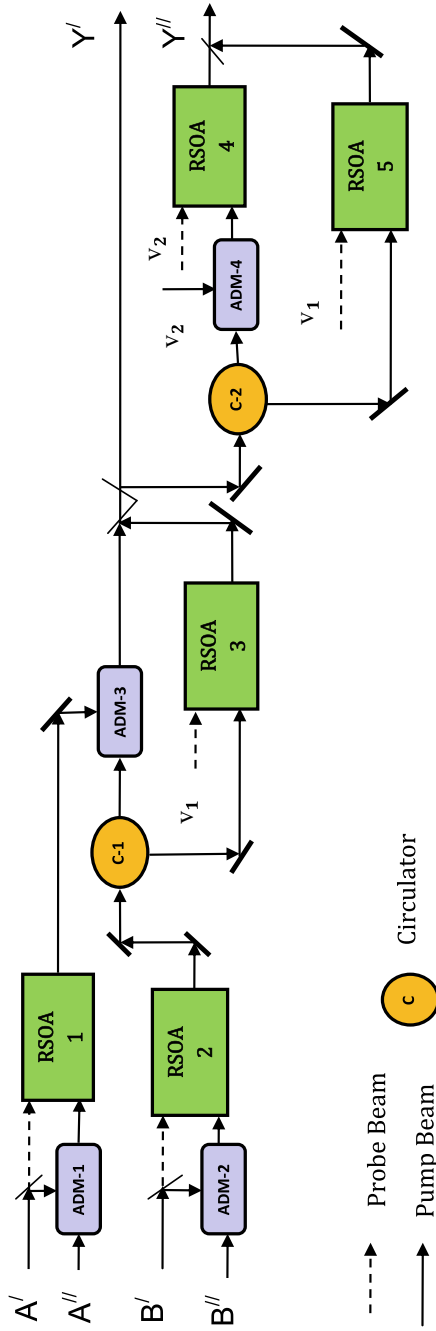


Fig. 3 Schematic diagram of the proposed inhibitor logic design

Case-1: In this case, RSOA-2 and RSOA-1 have the probe signals of frequency ν_1 , so the outputs are ν_1 . Since the biasing and input frequencies are ν_1 , so ADM-3 reflects the input frequency, ν_1 . This reflected frequency, ν_1 , works as a pump signal of RSOA-3 and also the frequency ν_1 works as a probe signal. As a result, RSOA-3 yields frequency ν_1 at the output. Therefore, one part of this frequency ν_1 is obtained at the terminal Y' and another part applied as the input signal of ADM-4. Since ν_2 is the bias signal of ADM-4, then ν_1 is selected by ADM-4. This frequency, ν_1 , works as a pump signal of RSOA-4, and frequency, ν_2 , is the probe signal, so the output is ν_2 . RSOA-5 is not working because missing of the pump signal. Therefore, frequency, ν_2 , is obtained at the output terminal, Y'' . The dibit output is $\langle \nu_1 \rangle \langle \nu_2 \rangle$, i.e., $\langle 0 \rangle \langle 1 \rangle$, digitally it shows '0' logic state.

Case-2: ν_1 and ν_2 frequency-encoded signals are applied to the input of RSOA-1 and RSOA-2, respectively, so at the output, ν_1 and ν_2 frequencies are obtained, respectively. As the biasing frequency of ADM-3 is ν_1 , so input frequency ν_2 is selected by the ADM-3. Therefore, frequency ν_2 reaches the output Y' . This signal acts as the input signal of ADM-4, and its biasing signal is frequency ν_2 , so ADM-4 reflects the input signal of frequency, ν_2 , to the pump signal port of RSOA-5. At the output Y'' , ν_1 is obtained because ν_1 is probe signal of RSOA-5, and RSOA-4 is not working because missing of pump signal. So, the dibit output is $\langle \nu_2 \rangle \langle \nu_1 \rangle$, i.e., $\langle 1 \rangle \langle 0 \rangle$ which indicates the digital logic state '1'.

Case-3: In this instance, RSOA-2 and RSOA-1 have the probe signals of ν_1 and ν_2 , so the outputs are ν_1 and ν_2 , respectively. This two frequencies work as input and biasing signal of ADM-3 which selects the frequency ν_1 . Therefore, one part of this frequency ν_1 is obtained at the terminal Y' and another part inserted to the input of ADM-4. Since ν_2 is the biasing signal of ADM-4, then input signal ν_1 is selected by ADM-4. This frequency, ν_1 , works as a pump signal of RSOA-4, and frequency, ν_2 , is the probe signal, so the output is ν_2 . RSOA-5 is not working because of the missing of pump signal. So, the frequency, ν_2 , is obtained at the output terminal, Y'' . The dibit output is $\langle \nu_1 \rangle \langle \nu_2 \rangle$, i.e., $\langle 0 \rangle \langle 1 \rangle$, digitally it shows '0' logic state.

Case-4: In this instance, RSOA-2 and RSOA-1 have the probe signals of ν_2 , so the outputs are ν_2 . The input and biasing signals are of frequency ν_2 , so ADM-3 reflects the frequency ν_2 . This reflected signal ν_2 works as a pump signal of RSOA-3, and the frequency ν_1 works as a probe signal that yields frequency ν_1 at the output. Therefore, one part of this frequency ν_1 is obtained at the terminal Y' and another part injected into the input of ADM-4. Since ν_2 is the biasing frequency of ADM-4, then ν_1 frequency is selected by ADM-4. This frequency, ν_1 , works as a pump signal of RSOA-4, and frequency, ν_2 , is the probe signal, so the output is ν_2 . RSOA-5 is not working because of the missing of pump signal. So, the frequency, ν_2 , is obtained at the output port, Y'' . The dibit output is $\langle \nu_1 \rangle \langle \nu_2 \rangle$, i.e., $\langle 0 \rangle \langle 1 \rangle$, that represents '0' logic state. Table 1 represents the inhibitor logic's Truth table.

4 Simulation of Proposed Inhibitor Logic

In the previous section, the operational scheme of the proposed design is logically explained. In this section, we discussed the simulation of the proposed inhibitor logic using MATLAB (R2018a) software. With the help of MATLAB language, programming is done for RSOA and ADM. Here, two different frequencies, $\nu_1=193.5$ THz (Wavelength=1550 nm) and $\nu_2=194.1$ THz (Wavelength=1545 nm), are taken to perform the simulation. In Figs. 4a, b, and 5, dibit input and output signals are delineated. According to Truth Table 1, $\langle 193.5 \rangle \langle 194.1 \rangle$, and $\langle 193.5 \rangle \langle 194.1 \rangle$, dibit signals are applied in the design for 50 ps to the dibit input (A' and A'') and input (B' and B'') terminal, respectively, and after the simulation, $\langle 193.5 \rangle \langle 194.1 \rangle$ obtains as dibit output Y' and Y'' respectively. Dibits, $\langle 193.5 \rangle \langle 194.1 \rangle$, $\langle 194.1 \rangle \langle 193.5 \rangle$ are applied to the inputs (A' and A''), and (B' and B'') terminal respectively then $\langle 194.1 \rangle \langle 193.5 \rangle$ obtained at the dibit output terminals. Similar way, simulated outcomes of this design are tabulated in Table 2.

5 Results and Discussion

In the previous section, a simulation of the devised design is done. Therefore, the outcome of the simulation is analyzed and discussed in this section. When the devised design is simulated according to the input signals of Truth Table 1, then dibit output signal waveforms are yielded. The output dibit signal waveforms are given in Fig. 5.

In first 50 ps, we applied ν_1, ν_2 in A and ν_1, ν_2 in B that indicates $A = 0, B = 0$. After simulation with these data, we obtained $Y' = \nu_1, Y'' = \nu_2$ that indicate $Y = 0$. In 50–100 ps, we applied ν_1, ν_2 in A, ν_2, ν_1 in B that indicate $A = 0, B = 1$. After simulation with these data, we obtained $Y' = \nu_2, Y'' = \nu_1$ that indicate $Y = 1$. In 100–150 ps, we applied ν_2, ν_1 in A, ν_1, ν_2 in B that indicate $A = 1, B = 0$. After simulation with these data, we obtained $Y' = \nu_1, Y'' = \nu_2$ that indicate $Y = 0$. In 150–200 ps, we applied ν_2, ν_1 in A, ν_2, ν_1 in B that indicate $A = 1, B = 1$. After simulation with these data, we obtained $Y' = \nu_1, Y'' = \nu_2$ that indicate $Y = 0$. After the verification of the simulation results (Table 2) and truth table (Table 1), it is interpreted that the proposed design works accurately. A comparative study with previous work is given in Table 3.

6 Conclusion

A design of all-optical frequency-encoded dibit-based inhibitor logic operation using RSOA and ADM is proposed here. To minimize the loss, one can be carefully select the optical components. This design can be fabricated in photonic band gap (Pbg) where RSOA can be combined with Pbg. MATLAB (R2018a) software is used to

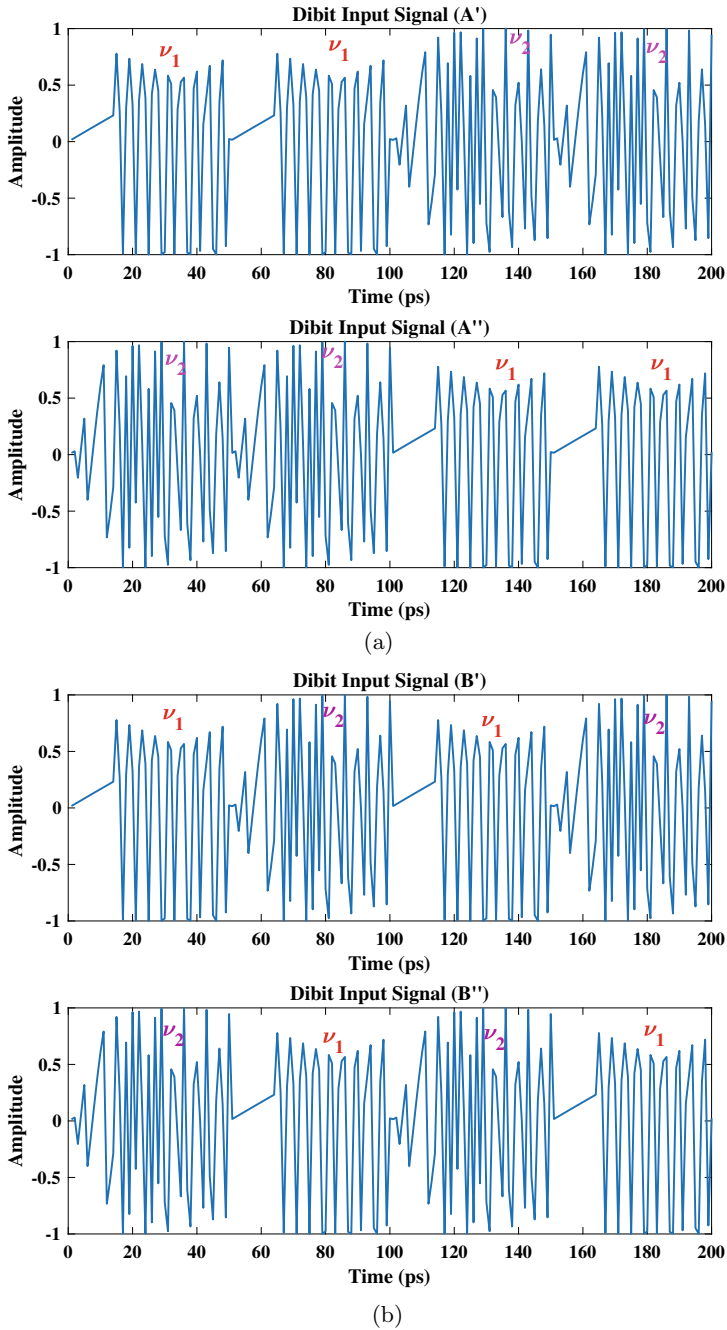


Fig. 4 a Dibit input (A' and A'') waveforms. b Dibit input (B' and B'') waveforms

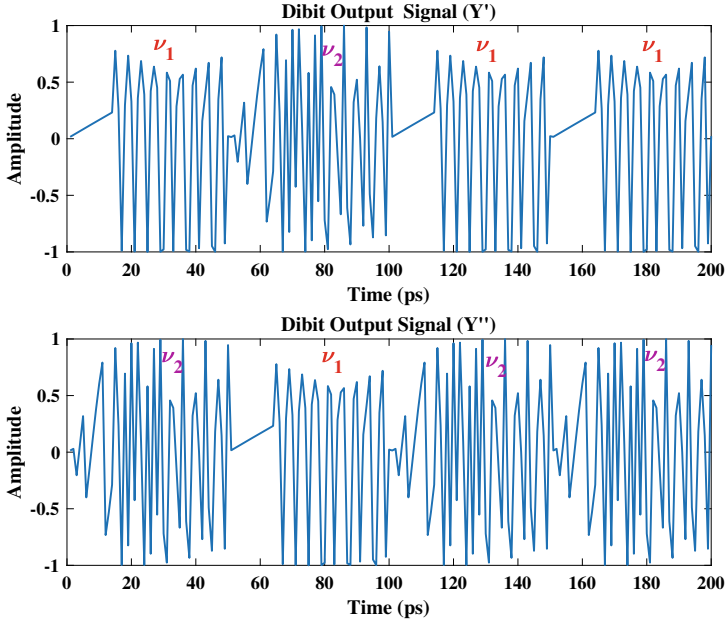


Fig. 5 Dibit output waveforms of proposed inhibitor logic

Table 2 Simulation results of the proposed design (THz frequencies are taken here)

Dibit input					Dibit output	
	Signal (A)		Signal (B)		Signal (Y)	
Time	A'	A''	B'	B''	Y'	Y''
0–50 ps	193.5	194.1	193.5	194.1	193.5	194.1
51–100 ps	193.5	194.1	194.1	193.5	194.1	193.5
101–150 ps	194.1	193.5	193.5	194.1	193.5	194.1
151–200 ps	194.1	193.5	194.1	193.5	193.5	194.1

simulate the devised design. In this design, frequency encoding decreases the probability of bit error problem. Another advantage is that false dibit logic cannot be injected into the device due to the dibit-checking units. This design performs at ultra-high speed with low noise. A high degree of parallelism can be expected from dibit-based concept. Using this device, we intend to design Feynman gate, Fredkin gate, adder, subtractor, etc. in the future.

Table 3 Comparison with previous work

Operating factor	Proposed design	Ref.[16]
Wavelength	1536–1570nm	Independent
Biasing current	120 mA	No need
Pump signal power	4–8 dBm	Not given
Probe signal power	–4 to –8 dBm	No need
Temperature	Dependent	Independent
Phase	Independent	Dependent
Bandwidth (BW)	Produces narrow channel spaced devices	Narrow
Tuneability	Medium	Low
Logic used	Dibit-based	Dibit-based
Encoding of data signal	Frequency	wavelength/frequency

Acknowledgments We are thankful to the Department of Physics, Bankura University, Bankura, Pin-722155, West Bengal, India, for support to conduct this research work.

References

1. Dey S, Chatterjee A, Mukhopadhyay S (2021) Photonic scheme for developing manchester-coded data using laser-based kerr switch. *J Opt*:1–5
2. Samanta D (2019) Implementation of polarization-encoded quantum fredkin gate using kerr effect. *J Opt Commun* 1
3. Raja A, Mukherjee K, Roy JN (2021) Analysis of new all optical polarization-encoded dual SOA-based ternary NOT and XOR gate with simulation. *Photon Netw Commun* 41(3):242–251
4. Rakheja S, Kani N (2017) Spin pumping driven auto-oscillator for phase-encoded logic-device design and material requirements. *AIP Advances* 7(5):055905
5. Mandal M, Maji R, Mukhopadhyay S (2021) Increase of side band powers in parallel phase modulation by lithium niobate-based electro-optic crystal. *Br J Phys* 51(3):738–745
6. Mukherjee K (2011) Implementation of a novel hybrid encoding technique and realization of all optical logic gates exploiting difference frequency generation alone. *Optik* 122(4):321–323
7. Qian C, Lin X, Lin X, Xu J, Sun Y, Li E, Zhang B, Chen H (2020) Performing optical logic operations by a diffractive neural network. *Light Sci Appl* 9(1):1–7
8. Zhong L (2017) Spatial encoded pattern array logic and design of 2d-encoding device for optical digital computing. In: *International conference on optoelectronic science and engineering'90*, vol 1230. International Society for Optics and Photonics, p 12307B
9. Bosu S, Bhattacharjee B (2021) A novel design of frequency encoded multiplexer and demultiplexer systems using reflected semiconductor optical amplifier with simulative verification. *J Opt* 50(3):361–370
10. Maji K, Mukherjee K, Raja A (2019) An alternative method for implementation of frequency-encoded logic gates using a terahertz optical asymmetric demultiplexer (toad). *J Comput Electron* 18(4):1423–1434
11. Bosu S, Bhattacharjee B (2021) All-optical frequency encoded dibit-based half adder using reflective semiconductor optical amplifier with simulative verification. In: *2021 devices for integrated circuit (DevIC)*. IEEE, pp 388–392

12. Saha S, Mukhopadhyay S et al (2020) All optical frequency encoded quaternary memory unit using symmetric configuration of mzi-*soa*. *Opt Laser Technol* 131:106386
13. Bosu S, Bhattacharjee B (2022) All-optical frequency encoded dicit-based half subtractor using reflective semiconductor optical amplifier with simulative verification. In: Dua M, Jain AK, Yadav A, Kumar N, Siarry P (eds) *Proceedings of the international conference on paradigms of communication, computing and data sciences. Algorithms for intelligent systems*, Springer, Singapore, pp 29–38
14. Garai SK (2014) A novel method of developing all optical frequency encoded fredkin gates. *Opt Commun* 313:441–447
15. Mukhopadhyay S (1992) Binary optical data subtraction by using a ternary dicit representation technique in optical arithmetic problems. *Appl Opt* 31(23):4622–4623
16. Ghosh B, Biswas S, Mukhopadhyay S (2015) A novel method of all-optical wavelength encoded logic and inhibitor operations with dicit representation technique. *Optik* 126(4):483–489
17. Maji K, Mukherjee K, Raja A, Roy JN (2020) Numerical simulations of an all-optical parity generator and checker utilizing a reflective semiconductor optical amplifier at 200 Gbps. *J Comput Electron*:1–15
18. Mukherjee K, Maji K, Raja A, Mandal MK (2021) All-optical soliton based universal logic norutilizing a single reflective semiconductor optical amplifier (RSOA). *Photon Netw Commun*:1–8
19. Ghosh B, Hazra S, Haldar N, Roy D, Patra SN, Swarnakar J, Sarkar PP, Mukhopadhyay S (2018) A novel approach to realize of all optical frequency encoded dicit based xor and xnor logic gates using optical switches with simulated verification. *Opt Spectroscopy* 124(3):337–342
20. Sarkar PP, Satpati B, Mukhopadhyay S (2013) New simulative studies on performance of semiconductor optical amplifier based optical switches like frequency converter and add-drop multiplexer for optical data processors. *J Opt* 42(4):360–366
21. Bosu S, Bhattacharjee B (2021) A design of frequency encoded dicit-based comparator using reflective semiconductor optical amplifier with simulative verification. In: *2021 devices for integrated circuit (DevIC)*. IEEE, pp 175–179

Ocean Surface Pollution Detection: Applicability Analysis of V-Net with Data Augmentation for Oil Spill and Other Related Ocean Surface Feature Monitoring



Naishadh Mehta, Pooja Shah, Pranshav Gajjar, and Vijay Ukani

Abstract Oil spills are considered to be one of the biggest obstacles to marine and coastal environments. Effective surveillance, ship detection, and accurate oil spill detection are crucial for the relevant agencies to respond adequately and minimize environmental emissions and avoid further disruption. Satellites deployed for capturing the data have led to the ingestion of huge amounts of remote sensing data to systems but to analyze that data via human effort is a tedious and extensive task. Hence, modern literature suggests the use of machine learning in paradigms such as image segmentation, image recognition, object detection as a substitute for traditional techniques. This research applies the contemporary deep Learning methods over the dataset available from the European Space Agency (ESA). The paper proposes the use of the volumetric convolution net (V-Net) architecture in addition to image augmentation methods like horizontal flipping, vertical flipping, and image rotation. The proposed computational pipeline resulted in a net Mean IOU of 88.29 and an accuracy of 90.65%.

Keywords Marine pollution · Remote sensing · Oil spill detection · SAR · V-Net

1 Introduction

Marine pollution occurs when chemicals, industrial, and domestic waste or invasive organisms enter oceanic bodies. Statistics estimate that 3.2 million tons of oil per year are released from all sources into the environment [27], threatening oceanic life.

N. Mehta (✉) · P. Shah · P. Gajjar · V. Ukani
Institute of Technology, Nirma University, Ahmedabad, India
e-mail: 19mced07@nirmauni.ac.in

P. Shah
e-mail: pooja.shah@nirmauni.ac.in

P. Gajjar
e-mail: 19bce060@nirmauni.ac.in

V. Ukani
e-mail: vijay.ukani@nirmauni.ac.in

During cleaning activities, oil contamination in the world's oceans is mostly caused by operational discharges from sea vessels, viz., ships, tankers, etc. These discharge amount to about 2 million tons of oil per year and are equivalent to the tragedy of one full tanker per week [12]. As seen recently in Mumbai [1], where 79 thousand liter spill is causing havoc, and Sri Lanka [22], which is bracing for an unpropitious disaster, oil spills have set off to be a major hindrance. A significant pollution factor is also land-based sources, such as urban waste and industrial discharges, which surround the ocean through rivers.

The challenge of oil spill detection and classification aims for retrieving significantly desired information that aids the grass-root officials on coastal relief zones to mitigate related problems with better efficiency. Ship detection provides ancillary information to aid spill classification.

Presently, marine pollution monitoring is done using data captured by synthetic aperture radar (SAR) [9]. SAR is an all-weather, day, and night sensor with significantly good spatial resolution and large coverage [8, 14]. Oil dampens ripple waves under wind conditions from 2.5–3 m/s to 10–12 m/s and forms a flat coating on the surface, which in turn appears as dark spots on the lighter water surface on radar images [2]. This is due to the Bragg scattering effect [3] where the oil on the sea surface appears to be darker than the rest of the sea. So if narrowed down, the problem of oil spill detection is the detection of dark spot, feature extraction, and classifying it as a spill or Look-alikes [3]. This problem of oceanography can be put up as a classification problem to detect and classify elements labeled as the clean sea, oil spill, look-alikes, ship, and land. Table 1 mentions the state-of-the-art research contributions.

In terms of type and scale, oil slicks appear to show extreme diversity. Deep learning methods should be used to take into account the physical features of the oil slicks and their dispersion, so that geometric characteristics such as form, scale can be tested and replaced effectively with crafted characteristics. Taking this fact into account, semantic segmentation models could be used as robust alternatives to remove the rich knowledge content from SAR images, along with the presence of multi-class instances [17].

The current state of research calls out for use of established deep learning methods to improve classification accuracy. The promising advancement in the domain of deep learning motivates researchers toward utilizing various deep learning models. On the other hand, there is a data crunch when you talk of a very specific typical application that needs typical ocean surface data. This motivated the research to bridge the gap between the available deep learning solutions and the much-needed application demands of oil spill detection through satellite data. In this work, the deep learning algorithm that is established for medical science applications is brought into play in context to features that are similar in characteristics (like pancreatic cancer) with a spill in satellite images that are examined [11].

The main objective of the paper is presented in Fig. 1. We aim to explore the applicability of the volumetric network which is a CNN used for 3D medical image classification [24]. A comparative study of other dominant models, viz., U-Net [17]

Table 1 Previous research work done

Author	Research highlight
Fiscella et al. [8]	Probabilistic approach in which the photographs processed are matched to identify models and then categorize them like an oil spill or look-alikes
Espedal [7]	Presented a strategy for enhancing the recognition of oil spills by providing details on wind history and an approximation of their time of life
Topouzelis and Psyllos [37]	For productive feature collection for oil spill classification, a decision tree forest was deployed
Del Frate et al. [6]	Introducing neural networks in the field of oil spill detection. Here to approximately related names, pre-determined features from SAR images were fed to the network
Authors of [5] [36] [30]	For the feature extraction process, a neural network was used to enrich the internal representations of the input
Song et al. [33]	To train a wavelet neural network, which categorizes black spots captured in SAR images into oil spills and unspoiled water areas, they used handcrafted features
Stathakis et al. [34]	To differentiate oil spills from look-alikes, a shallow neural network was developed, focusing on a solution in terms of computational cost and accuracy of detection

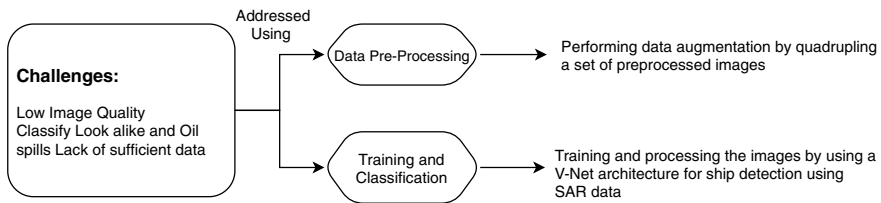


Fig. 1 Objective of the study

and other comparative architectures is also provided to confirm the usability of V-Net. The contribution of the study can be summarized as:

- Usability and analysis of V-Net for ocean surface feature like oil spills, look-alikes, ship, land and ocean surface detection from SAR data.
- Applying data augmentation variations, namely vertical flip, horizontal flip, and random rotation.
- Performance comparison with other dominant models

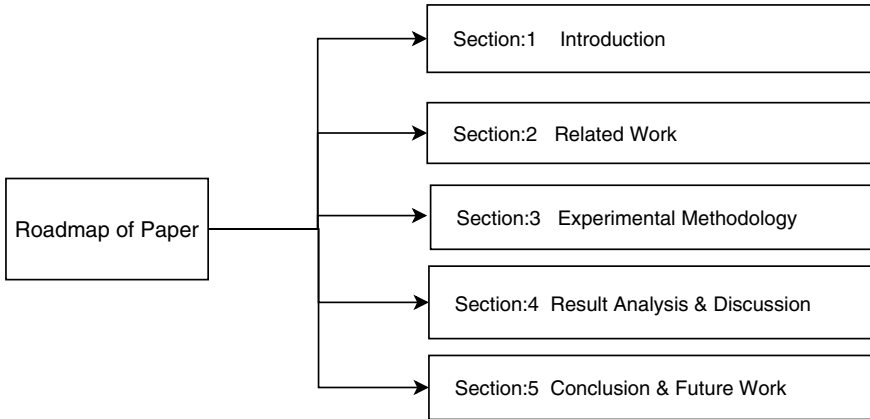


Fig. 2 Roadmap of the paper

- Presenting performance evaluation in terms of accuracy, Dice coefficient, and Mean IoU.

The roadmap of the paper is as shown in Fig. 2.

2 Related Work

Object detection is an active subject in computer vision and has witnessed momentous upgrading a few times. With the rising of self-sufficient vehicles, smart video surveillance, facial detection, ship detection, and many more other applications have exceptionally increased the use of object detection techniques. However, object detection still poses major challenges such as an unknown number of instances, unequal size of images, unknown dimensions of the model output, to name a few [21]. The object detection task consists of feature extraction and classification. Considering feature extraction method ship detection can be divided into two groups, namely handcrafted feature and machine learning feature [26]. Handcrafted ship detection feature techniques are based on form, edge, and texture characteristics, but the handcrafted extraction technique has low rationalization ability [26]. To overcome the shortfalls of handcrafted feature extraction, machine learning is trending as a befitting approach for ship detection. Solberg et al. [32] to distinguish SAR signatures between oil spills and look-alikes, this automated system relies on this three-phase method. In this case, the classifier must be granted prior knowledge of the likelihood of the occurrence of oil spills.

Liu et al. [20] have applied techniques like image thresholding, color/texture intolerance, create a histogram of oriented gradient, and support vector machines on optical remote sensing data. In deep learning, they have proposed techniques like

Table 2 Literature review summary

Approach	Dataset	Result
Singleshot multibox detector [21]	Self-collected data via Google earth	Mean average precision (mAP) of 87.3%
Thresholding-guided maximally stable extremal regions (TGMSEs) algorithm [39]	Radarsat-1	0.51–50.82% false alarm rate
Singleshot multibox detector [26]	Self-collected data via Google earth	87.9 % Average precision (AP)
Convolutional neural network [13]	Dataset from Google earth	95% Accuracy
Classifier based on Xception and U-Net model ResNet-18 as encoder [20]	Airbus ship detection challenge	84.78% accuracy
Decision fusion algorithm [38]	GF-1 WFV oil spill image	78.79 % <i>F1</i> -score
R-CNN [15]	COCO image dataset	Not mentioned
CapsNet [28]	22-Sentinel-1 and 3 Radarsat-2	91.03% accuracy
Xception NC + l2 + kNN [10]	MASATI dataset	94.69% <i>F1</i> -score
Generative adversarial network [18]	Gaofen-3 dataset	97.5% accuracy

convolutional Neural network (CNN) optimization, VGG-16, Xception, Inception, ResNet [20]. They have proposed another approach on Airbus dataset of 200,000 labeled optical satellite images of the ship with CNN model with the accuracy of 93.87% Li et.al [18] have applied improved residual condition generative adversarial network (GAN) for classification of SAR image ship object generation and achieved an accuracy of successfully classify cargo as 97.5%. A tabular depiction of the modern literature is mentioned below.

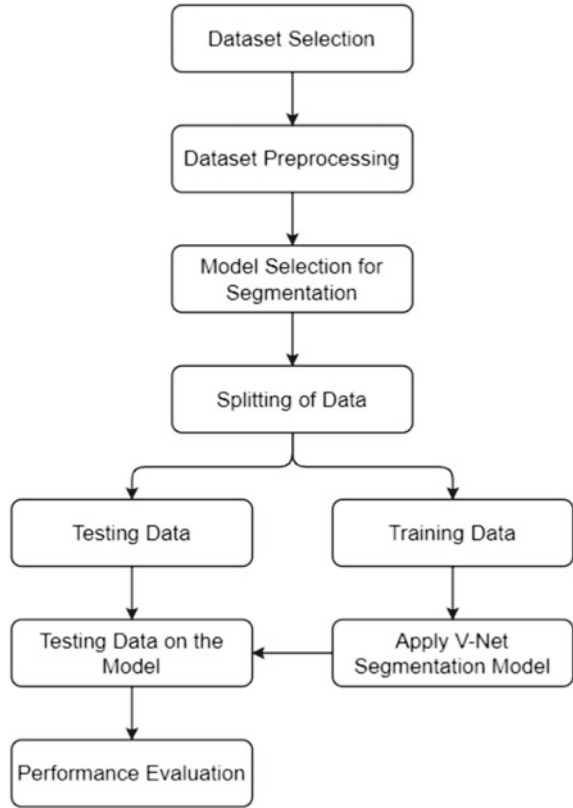
Table 2 gives us a thorough understanding of the deep learning architectures present throughout the modern literature, and this further enhances the novelty of V-Nets, augmentative approaches, and the proposed methodologies in this domain.

3 Fundamental Methodology

The fundamental methodology is presented in Fig. 3. This section focuses on portraying the significance of each phase of the fundamental methodology.

In this proposed research work first, we select the dataset which contains satellite images with five classes termed ships, look-alikes, oil spills, ocean surface, and land. After that, we preprocess the dataset as per our practical requirements followed by the selection of an appropriate segmentation model. Then, we had split the dataset in

Fig. 3 Fundamental methodology



terms of training and testing data as per the standard procedure. In the next step, we have trained the V-Net model on the training data, and to measure the performance of our applied model, we test the model on training data and measure its performance using different performance measures.

3.1 Data Selection

The dataset used for the research work is a collection of aerial SAR images of oil contaminated maritime regions acquired using the European Space Agency (ESA) database, the Copernicus Open Access Hub. The European Maritime Safety Agency (EMSA) released information on the spatial location of the pollution incident as well as timestamps when it occurred via the CleanSeaNet administrator. Oil pollution records cover the time period from September 28, 2015 to October 31, 2017, when Sentinel-1 European Satellite Mission captured SAR images. The collection comprises 1112 images divided into five categories: oil spills, look-alikes, ships,

Table 3 Ground label description

Color code	Denotes
Black	Ocean surface
Cyan	Oil spills
Red	Look-alikes
Brown	Ship
Green	Land

land, and ocean surface. For each ocean attribute, specific RGB colors have been assigned. As a result, each case is colored by the described class, resulting in ground truth pictures that signify for each unique image. The coverings are normally useful for the depiction of data after conducting semantically division; nevertheless, for the development and assessment method, 1D target markings can be used instead of RGB esteems [23]. The dataset is separated into two subgroups: training and testing. Training contains 90% of the photographic images (i.e., 1002 photographs), while testing contains 10% of the photographic images (i.e., 110 pictures) [17].

The dataset used for the research purpose consisted of five classes, and each class has been shaded by a specific color.

3.2 Data Preprocessing

A preprocessed dataset [17] is used having the following features:

1. Every recognized oil slick was defined by an EMSA statement.
2. A district holding substantial data or other extraneous data was clipped from the initial SAR images and downsized to the desired size of 1250×650 pixels.
3. To resize the image in a target size of 1250×650 pixels, a plane with a comparable radiometric orientation was employed.
4. To eliminate the noise sensor in the picture, a specular filtration was applied, accompanied by a window median filter of 7×7 .
5. For the transfer of decibel (dB) to true brightness value, a linear transformation approach was used.

3.3 Semantic Segmentation Model Selection

Semantic segmentation is the technique of classifying every pixel of an image into its respective class. In the case of this research work, volumetric neural network architecture (V-Net) is adopted.

V-Net is a well-known variation of U-Net recommended by Milletari [24] that was essentially important for medical image segmentation. However, in healthcare data such as CT or MRI images, it is a difficult task to convey specific and sensitive data about the contours and amounts of various organs. Automatic demarcation of organ and visualization of organ anatomy are required to undertake tasks such as visual augmentation, computer-assisted diagnosis, procedures, and separation of quantitative quantifiable indicators from pictures [24]. V-Net is also used to classify neuron pictures that have distortion and low voxel luminosity [19]. Cell experts utilized V-Nets to extract features and identify tiny entities in 3D pictures with a classification accuracy of 73% utilizing border and boundaries labels [4]. A conventional 3D data gathering consists of a set of 2D sliced pictures recorded by a CT, MRI, or micro-CT scanner [25]. Efforts have been made to semantically section the pancreas in computed tomography photographs for sensing tumors, infectious diseases, and other concussions, but due to variations of shape of the organ and small biomolecules, as well as low-resolution image formation, computed tomography did not provide precise findings, so V-Nets are used [25].

As discussed earlier, that V-Net is mainly applicable in semantic segmentation on medical image diagnosis but as that, we are going to apply V-Net for semantic segmentation from SAR images. SAR images also have a spectral resolution which is very low, and SAR images detect objects with the materials and geometry of the object itself [10]. So, we can say that medical imaging and satellite data for oceanic applications are having similar feature dimensionality, and small objects and/or boundaries are difficult to detect. For this purpose of detection, V-Net is implemented.

The V-Net design is bilateral, with bound connections connecting the encoding and decoding phases. There are latent connections at each point of the V-Net to ensure the training of the data generated. V-Net employs entirely linked layers and flatten procedures to generate equally sized outcome pictures [4]. Both pooling layers used in the downsampling portion are transformed into upsampling layers. On each layer, one contract path is built to connect the upsampling and downsampling parts [4]. The visual depiction of the vanilla V-Net infrastructure as proposed in the original textual literature is provided in the below mentioned Fig. 4. No changes are made to the architecture, and the same model is used to preserve experimental clarity.

The left portion of the network is a compression path, while the right section decompresses the signal until its original size is attained, and a residual function is learned at each level to ensure convergence. The characteristics collected from the early stages of the left portion of the CNN are transmitted to the right part through horizontal connections, assisting the right part in giving location information and resulting in a superior output.

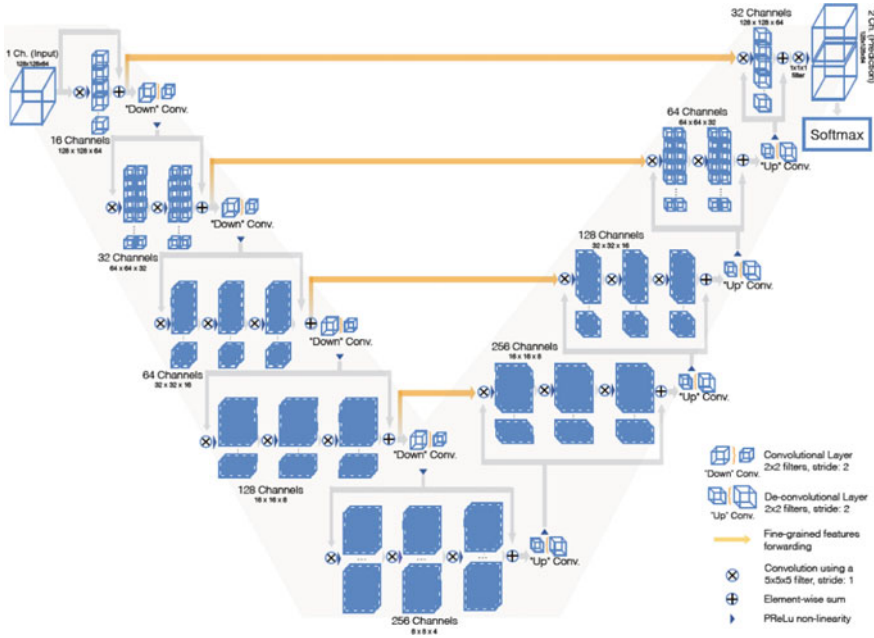


Fig. 4 V-Net infrastructure diagram [24]

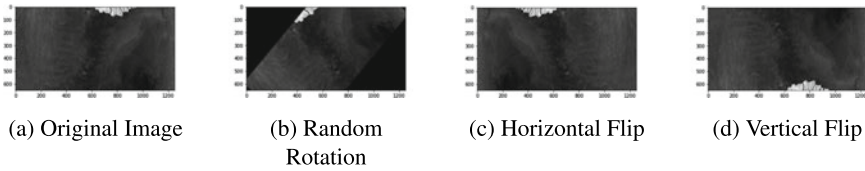


Fig. 5 Data augmentation on images

3.4 Data Augmentation

The paper proposes the use of image augmentation to increase the dataset size, further increasing the model performance. The suggested approach quadruples the training set by using horizontally rotating, vertically rotating, and randomized rotation [29]. Arrays of both the ground truth image and the satellite images undergo the same mathematical changes resulting in an augmented sample (Figs. 5 and 6).

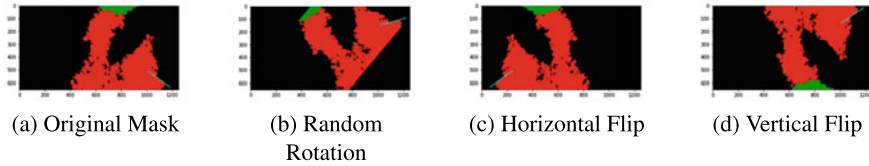


Fig. 6 Data augmentation on masks

3.5 Experimental Details

Training dataset consists of 1002 images, and testing data consists of 110 images. Data augmentation is performed on the 1002 images, which are quadrupled. The dataset, which includes ground truth photographs and masks, has an image size of 320×320 . For 600 epochs and two picture samples per batch, the models are trained using an 80–20 train validation split. V-Nets are trained on enhanced photographs for 600 epochs using Adam optimization and a learning rate of 10^{-3} . About 4008 pictures are used for training and 110 for testing in the V-Net with image augmentation. V-Net has the same image size, number of epochs, learning rate, optimizer, and batch size. The aforementioned hyper-parameters linked with model training are backed by experimental superiority. Following the training of our model on training data, we will test our model on testing data and compare the projected outcomes to the ground truth labels.

3.6 Performance Evaluation

Model performance and associated hyper-parameters are evaluated by using percentage accuracy values and dice coefficient.

Accuracy: Accuracy is defined as the number of correct predictions divided by the total number of forecasts. [31].

$$\text{Accuracy} \rightarrow (\text{TP} + \text{TN}) / (\text{TP} + \text{FP} + \text{FN} + \text{TN}) \quad (1)$$

where the number of true positives, false negatives, false positives, and true negatives are, respectively, represented by TP, FN, FP, and TN.

Dice Coefficient: Dice coefficient/F1-score in simple words can be described as $2 \times$ area of overlap in both images divided by the total pixels in both the images [35]. Dice coefficient's value can be between 0 and 1. Here, 1 represents higher similarity between predicted value and ground truth value. Dice coefficient can be measured in Eq. 1 as follows:

$$DSC \rightarrow 2|S \cap R| / |S| + |R| \quad (2)$$

Here, S is the segmentation result, and R is the ground truth label associated with the image.

Mean IoU: Intersection over Union is a performance metric that is used for measuring model performance when we are using semantic segmentation. IoU is defined as the area covered by overlapping between our model's expected results and ground truth segmentation divided by the union of the areas covered by both predicted results and ground truth. It is mainly between 0 and 1, with 0 indicating no overlap and 1 indicating complete overlap. Mean IoU is used when we have more than 2 classes, so in this method, we mainly take the IoU of each class and then take the mean of it.

$$IoU \rightarrow \text{Area of Overlap} / \text{Area of Union} \quad (3)$$

Categorical Cross entropy Loss: Cross entropy is the standard loss function used especially for classification problems where we have more than two classes in which a special attribute was allocated to each class [16].

It will calculate the average difference between the actual and predicted probability distributions for all the classes of the model. The ideal cross entropy value is 0.

The mathematical representation:

$$L(y, \hat{y}) \rightarrow -1/N \left(\sum_{i=0}^N (y_i \log(\hat{y}_i)) \right) \quad (4)$$

In this context, y_i is the probability that event i occurs and the sum of all y_i is 1, meaning that exactly one event may occur. N is the number of classes. The minus sign means that the loss becomes smaller as the distributions get similar to each other.

4 Results and Discussion

We have implemented this experimental work on NVIDIA GP 100 GPU with a different number of epochs. The model performance is measured by using accuracy, dice coefficients, Mean IoU, and loss. To rectify the model performances, proposed methodologies are compared to a baseline model following a U-Net architecture. A standard V-Net achieved an average segmentation accuracy of intended classes achieved is 90.5%, 2.7402 loss, and a 90.2% Dice coefficient for our testing dataset. The V-Net trained on the augmented dataset achieved a superior accuracy of 90.65%,

Table 4 Model results comparison

Model	Mean IoU (%)	Dice coefficient (%)	Accuracy (%)
DeepLabv2	49.27	–	–
PSPNet	55.60	–	–
LinkNet	64.79	–	–
U-Net	65	67.42	64.90
V-Net	85.68	90.20	90.50
V-Net + augmentation	88.29	90.34	90.65

Bold represents the best performing architecture

loss of 0.69, and a 90.34% Dice coefficient. For additional comparisons, deep learning architectures from [17] are added to Table 4 (Rows 1–3) which are trained on an identical dataset and comparable training conditions.

The use of V-Nets is deemed successful as an increase of 20.58 Mean IoU is observed when a standard V-Net is compared to a baseline U-Net. In V-Net models, the use of data augmentation yielded excellent results, with a mean IoU increase of 2.71. Because traditional augmentation methods like picture flipping and array rotation consume far less resources than a deep learning-based solution, the suggested pipeline made heavy use of them. Mean IOU is considered as the prime implicating factor of model performance, due to the segmenting nature of the task. V-Net and augmentation also outperformed the DeepLabv2, PSPNet, and LinkNet architectures [17] by a substantial mean IoU margin. The proposed methodology gave the best performance in this comparative study, with a stellar accuracy of 90.65%.

The predicted results of U-Net, V-Net, and V-Net with augmentation on random input samples and associated ground truth images are depicted in Fig. 7.

5 Conclusion and Future Work

Oil spills have set off to be a major hindrance to marine life and oceanic entities among the ocean surface pollutants. To automate the task of oil spill detection from SAR imaging, a novel V-Net-based deep learning pipeline-oriented approach is developed and described under this research work. It further involves mathematical augmentative implementations on image arrays. There was a threefold increase in the dataset, which led to a better training set for the V-Net architecture. The proposed methodology outperformed the tested baselines; in our case, it is U-Net with a percent Dice coefficient of 90.34 and percentage accuracy of 90.65, and other architectures like Deeplabv2, PSPNet, and LinkNets when compared for the mean IoU metric. In the near future, we are aiming to test generative adversarial network-based augmentative approaches for ocean-based feature detection and also in finding alternate use cases for the mentioned methodologies.

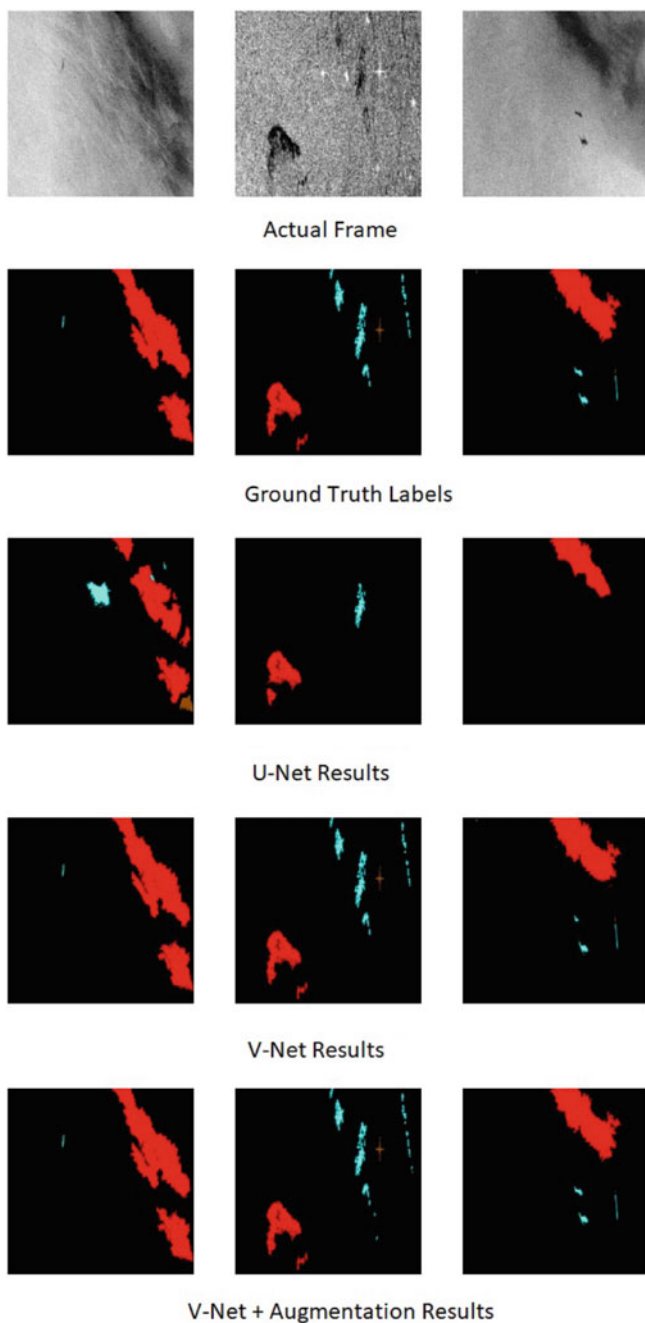


Fig. 7 Results comparison of the proposed architectures and the U-Net baselines with ground truth images

References

1. Admin O (2021) Mumbai: 85 floating tanks to remove 79,000 litres of barge's oil, May 2021
2. E.S. Agency (1998) European space agency brochure
3. Alpers W, Holt B, Zeng K (2017) Oil spill detection by imaging radars: Challenges and pitfalls. *Remote Sens Environ* 201:133–147
4. Chang C-S, Ding J-J, Chen P-H, Wu Y-F, Lin S-J (2019) 3-d cell segmentation by improved v-net architecture using edge and boundary labels. In: 2019 IEEE 2nd international conference on information communication and signal processing (ICICSP). IEEE, pp 435–439
5. de Souza DL, Neto AD, da Mata W (2006) Intelligent system for feature extraction of oil slick in sar images: speckle filter analysis. In: International conference on neural information processing. Springer, pp 729–736
6. Del Frate F, Petrocchi A, Lichtenegger J, Calabresi G (2000) Neural networks for oil spill detection using ers-sar data. *IEEE Trans Geosci Remote Sens* 38(5):2282–2287
7. Espedal H (1999) Satellite sar oil spill detection using wind history information. *International Journal of Remote Sensing* 20(1):49–65
8. Fiscella B, Giancaspro A, Nirchio F, Pavese P, Trivero P (2010) Oil spill monitoring in the mediterranean sea using ers sar data. In: Proceedings of the envisat symposium (ESA), Göteborg, Sweden, pp 16–20
9. Gade M, Alpers W (1999) Using ers-2 sar images for routine observation of marine pollution in european coastal waters. *Sci Total Environ* 237:441–448
10. Gallego A-J, Pertusa A, Gil P (2018) Automatic ship classification from optical aerial images with convolutional neural networks. *Remote Sens* 10(4):511
11. Giddwani B, Tekchandani H, Verma S (2020) Deep dilated v-net for 3d volume segmentation of pancreas in ct images. In: 2020 7th international conference on signal processing and integrated networks (SPIN). IEEE, pp 591–596
12. Holstein A, Kappas M, Propastin P, Renchin T (2018) Oil spill detection in the kazakhstan sector of the caspian sea with the help of envisat asar data. *Environ Earth Sci* 77(5):198
13. Hordiiuk D, Oliinyk I, Hnatushenko V, Maksymov K (2019) Semantic segmentation for ships detection from satellite imagery. In: 2019 IEEE 39th international conference on electronics and nanotechnology (ELNANO). IEEE, pp 454–457
14. Ivanov AY, Zatyagalova VV (2008) A GIS approach to mapping oil spills in a marine environment. *Int J Remote Sens* 29(21):6297–6313
15. Kartal M, Duman O (2019) Ship detection from optical satellite images with deep learning. In: 2019 9th international conference on recent advances in space technologies (RAST). IEEE, pp 479–484
16. Koeh KE (2020) Cross-entropy loss function, Nov 2020
17. Krestenitis M, Orfanidis G, Ioannidis K, Avgerinakis K, Vrochidis S, Kompatsiaris I (2019) Oil spill identification from satellite images using deep neural networks. *Remote Sens* 11(15):1762
18. Li L, Wang C, Zhang H, Zhang B (2020) Sar image ship object generation and classification with improved residual conditional generative adversarial network. *IEEE Geosci Remote Sens Lett*:1–5
19. Liu M, Luo H, Tan Y, Wang X, Chen W (2018) Improved v-net based image segmentation for 3d neuron reconstruction. In: 2018 IEEE international conference on bioinformatics and biomedicine (BIBM). IEEE, pp 443–448
20. Liu Y, Cui H-Y, Kuang Z, Li G-Q (2017) Ship detection and classification on optical remote sensing images using deep learning. In: ITM web of conferences, vol 12. EDP Sciences, p 05012
21. Lutherborrow T, Agoub A, Kada M (2018) Ship detection in satellite imagery via convolutional neural networks, pp 2117–2125
22. Mallawarachi B (2021) Sri lanka races to avoid environmental disaster after chemical container ship sinks, Jun 2021
23. Mehta N, Shah P, Gajjar P (2021) Oil spill detection over ocean surface using deep learning: a comparative study. *Marine Systems & Ocean Technology*, pp 1–8

24. Milletari F, Navab N, Ahmadi S-A (2016) V-net: fully convolutional neural networks for volumetric medical image segmentation. In: 2016 fourth international conference on 3D vision (3DV). IEEE, pp 565–571
25. Nguyen PX, Lu Z, Huang W, Huang S, Katsuki A, Lin Z (2019) Medical image segmentation with stochastic aggregated loss in a unified u-net. In: 2019 IEEE EMBS international conference on biomedical and health informatics (BHI). IEEE, pp 1–4
26. Nie G-H, Zhang P, Niu X, Dou Y, Xia F (2017) Ship detection using transfer learned single shot multi box detector. In: ITM web of conferences, vol 12. EDP Sciences, p 01006
27. Saadoun IM (2015) Impact of oil spills on marine life, Sept 2015
28. Schwegmann CP, Kleynhans W, Salmon BP, Mdakane LW, Meyer RG (2018) Synthetic aperture radar ship detection using capsule networks. In: IGARSS 2018-2018 IEEE international geoscience and remote sensing symposium. IEEE, pp 725–728
29. Shorten C, Khoshgoftaar TM (2019) A survey on image data augmentation for deep learning. *J Big Data* 6(1):60
30. Singha S, Bellerby TJ, Trieschmann O (2013) Satellite oil spill detection using artificial neural networks. *IEEE J Sel Topics Appl Earth Observ Remote Sens* 6(6):2355–2363
31. Sokolova M, Lapalme G (2009) A systematic analysis of performance measures for classification tasks. *Inf Process Manage* 45(4):427–437
32. Solberg AS, Stovik G, Solberg R, Volden E (1999) Automatic detection of oil spills in ers sar images. *IEEE Trans Geosci Remote Sens* 37(4):1916–1924
33. Song D, Ding Y, Li X, Zhang B, Xu M (2017) Ocean oil spill classification with radarsat-2 sar based on an optimized wavelet neural network. *Remote Sens* 9(8):799
34. Stathakis D, Topouzelis K, Karathanassi V (2006) Large-scale feature selection using evolved neural networks. In: *Image and signal processing for remote sensing XII*, vol 6365. International Society for Optics and Photonics, p 636513
35. Tiu E (2020) Metrics to evaluate your semantic segmentation model, Oct 2020
36. Topouzelis K, Karathanassi V, Pavlakis P, Rokos D (2007) Detection and discrimination between oil spills and look-alike phenomena through neural networks. *ISPRS J Photogram Remote Sens* 62(4):264–270
37. Topouzelis K, Psyllos A (2012) Oil spill feature selection and classification using decision tree forest on sar image data. *ISPRS J Photogram Remote Sens* 68:135–143
38. Yang J, Wan J, Ma Y, Hu Y (2019) Research on object-oriented decision fusion for oil spill detection on sea surface. In: *IGARSS 2019-2019 IEEE Int Geosci Remote Sens Symp*. IEEE, pp 9772–9775
39. Zhang Q, Huang Y, Huo W, Gu Q, Pei J, Yang J (2018) Oil spill candidate detection from sar imagery using thresholding-guided maximally stable extremal regions algorithm. In: *IGARSS 2018-2018 IEEE international geoscience and remote sensing symposium*. IEEE, pp 5800–5803

Simulation and Investigations of I-shaped Patch Antenna with Induced SIW and Slit for S and C Bands Radar Applications



P. Esther Rani and Saam Prasanth Dheeraj

Abstract Antenna has been a prominent device in the field of radars particularly in the transmission and receiving of the required electromagnetic waves or signals at the desired frequency, and hence establishes a predominant technique to track and identify the clutter in the near or far radar range. This chapter puts forth an innovative rectangular patch antenna with included substrate-integrated wave guide structures along with a slit especially for S and C bands application of radars. Low throughput, narrowband and the dimensions of antenna have been major limitations at the desired frequency between 2 and 8 GHz when compared with the other bands. The simulation of the addressed antenna is performed in high-frequency structure simulator software, where FR4 epoxy of the dielectric constant of 4.3 is used as the substrate, and an input impedance of 50 Ohms and an electrical length of 90° are considered using finite element analysis method to achieve enhanced output parameters while inducing and implanting structures of substrate-integrated wave guide and slit, respectively. A descent return loss of -17.08 , -18.25 , -37.05 dB is realized at frequencies of 3.67, 5.15, 6.19 GHz with an enhanced bandwidth of 630, 530 MHz, and 1.52 GHz correspondingly, and making the posited antenna desirable. Other parameters of design optimization are investigated and delivered in this paper along with other parameters, understandings, and codicil.

Keywords Bandwidth · High-frequency structure simulator · Return loss · Substrate-integrated wave guide · Throughput · Voltage standing wave ratio

P. Esther Rani · S. P. Dheeraj (✉)

Department of Electronics and Communication Engineering, Vel Tech Rangarajan Dr, Sagunthala R&D Institute of Science and Technology, Chennai, Tamil Nadu, India
e-mail: dheeraj.saam007@gmail.com

P. Esther Rani

e-mail: drpestherrani@veltech.edu.in

1 Introduction

The advent of radio detection and ranging has announced its importance in no time as it has completely shifted the phase of traditional warfare and security to a new exciting level [1]. Antenna has its eminence in the proposed field of communication as the entire process is transmitted and received through it.

Usually, a Cassegrain parabolic method is used for the radar communication to find the clutter but considering the level of difficulty in design and the uncompact size constraints. Microstrip antennas are widely used [2] obeying the factors of input impedance, power handling, and mechanical robustness. Another way of approach for the targeted application is the implementation of the phased arrays [3], taking the electromagnetic inference into the accountability, it is not well preferred [4]. Major advantage of the S and C bands can be understood from the reference [5] as these bands lie in the ultra-wideband frequency and hence need not down convert the received signal and hence saves time during the clutter identification. This dissertation contemplates a slotted I-shaped slotted patch antenna with induced SIW structure and a slit using FR4 ($\epsilon_r = 4.4$) as substrate with the dimensions of $60 \text{ mm} \times 42 \text{ mm} \times 1.6 \text{ mm}$, and a microstrip transmission line is used as it shows better impedance characteristics when compared to other similar port realization techniques [6]. The simulated measurements of the transmission lines are as follows $15 \text{ mm} \times 2.5 \text{ mm}$ for length and width of the transmission line, respectively, with the electrical length ($\beta l = 90^\circ$), and the input impedance is considered to be 50Ω for proper distribution of the input power and minimal loss [7]. Identical slots of dimensions $9.6 \text{ mm} \times 14 \text{ mm}$ in rectangular shape are etched on the rectangular patch of measurements of $42 \text{ mm} \times 29 \text{ mm}$. An additional slit of width and length $2 \text{ mm} \times 8 \text{ mm}$ is removed from the ground.

Four identical cylindrical vias are induced into the substrate of 0.5 in inner radius placed 1.6 mm into the substrate from the surface of the patch to sustain multiple resonating frequencies. The induction of SIW structure also helps in achieving symmetrical bandwidth tending the loss characteristics to below, attains high throughput, and helps in dimension reduction answering the constraints recorded in references [8–10] along with ease of compacting in PCB process [11]. Enhancement of return loss and bandwidth is mainly addressed in this paper, and the mentioned understandings and the achieved results are bounded only to these aspects of gaining better results and documented for further applications.

2 Design Parameters

A multiband resonating antenna is simulated at the frequencies of 3.67, 5.15, 6.19 GHz, and the design of various elements is represented below, and all the values regarding the dimensions of the antenna are encapsulated in Table 1. Design equations of the transmission line are shown as the following

Table 1 Design parameters

Variable	Dimensions
Length of the substrate (LS)	60
Width of the substrate (WS)	42
Height of the substrate (H)	1.6
Length of the patch (LP)	42
Width of the patch (WP)	29
Length of the identical slots (LI)	9.6
Width of the identical slot (WI)	14
Length of the feedline (LF)	15
Width of the feedline (WF)	2.5
Length of the ground (LG)	60
Width of the ground (WG)	42
Length of the ground slit (LGS)	8
Width of the ground slit (WGS)	2

$$WF = \left(\frac{8e^A}{e^{2A} - 2} \right) H \quad (1)$$

where

$$A = \frac{Z_0}{60} \left(\sqrt{\frac{\epsilon_r + 1}{2}} \right) + \frac{\epsilon_r - 1}{\epsilon_r + 1} \left[0.23 + \frac{0.11}{\epsilon_r} \right] \quad (2)$$

$$B = \frac{377\pi}{2Z_0\sqrt{\epsilon_r}} \quad (3)$$

The Z_0 represents the input impedance of the transmission line which is considered to be 50Ω . The dielectric constant is shown with the symbol ϵ_r . The effective dielectric constant is needed to be measured to draw the length of the transmission line.

$$\epsilon_{\text{eff}} = \frac{\epsilon_r + 1}{2} + \frac{\epsilon_r - 1}{2} \sqrt{1 + \frac{10H}{WF}} \quad (4)$$

The design formulas for the patch measurements are given by

$$LP = \frac{C}{2fo\sqrt{\epsilon_r}} - 2\Delta L \quad (5)$$

$$\Delta L = 0.412H \frac{(\epsilon_r + 0.3) \left(\frac{WP}{H} + 0.264 \right)}{(\epsilon_r - 0.258) \left(\frac{WP}{H} + 0.8 \right)} \quad (6)$$

Table 2 Various design parameters

Symbol	Quantity	Units (SI)
f_0	Frequency	GHZ
Z_0	Impedance	Ohm
ϵ_r	Dielectric constant	Farad per meter
ϵ_{eff}	Effective dielectric constant	Farad per meter
C	Velocity	Meter per second

$$WP = \frac{C}{2f_0} \sqrt{\frac{\epsilon_r + 1}{2}} \quad (7)$$

The length and the width of the rectangular patch are represented in Eqs. (6) and (7). Here, ΔL presents the extension of the length on the substrate like the C represents the velocity of the light and f_0 corresponds to the centralized frequency or the resonant frequency. A slit is introduced into the ground, where the ground has the same dimensions of width and length same as the substrate; the length and the width of the slot are chosen carefully by understanding the characteristics of the delivered antenna. But, it is always optimal to use a slot placed horizontally to the ground plane [12] for better impedance match and long transmission range.

Rectangular slots are etched on the surface of the patch which is identical to each other in dimensions; these slots help in order to achieve a better bandwidth especially, where the problem is reported to be a narrow bandwidth in the S and C bands. All the units expressing various quantities and defining symbols are presented in Table 2.

2.1 Units

See Table 2.

3 Methodology

The finite element analysis method is used by the software in order to simulate the desired antenna, for S and C bands frequency. FR4 epoxy is used as a substrate whose ϵ_r is considered to be 4.3, and the tangential loss is 0.02. FR4 has outstripped results over other substrates like Duroid family [13] such as return loss, impedance match, and VSWR. FR4 also exhibits various physical properties such as toughness and robust nature as well as compatibility with the PCB process. Rather than using other means of port realization techniques unlike like a microstrip helps in better compatibility and offers perfect impedance match.

The patch is etched with identical rectangular slots, and the ground is cut with a minute slit and hence making it partial ground, both the ground and patch (etched slots make it to be I shaped) are assigned with a perfect E excitation, making it to radiate the electromagnetic waves. The identical etched slots help in achieving enhanced bandwidth obeying the Babinet principle [14]. The slots are placed in a horizontal orientation and hence helps in long-range transmission and perfect impedance patch and intrinsic impedance [15].

SIW structures are induced into the substrate in order to achieve multiple resonating frequencies at the desired range. These SIW structures are easy to compact with the antenna in the design process of PCB; most importantly, asymmetrical bandwidth is obtained and hence answering the limitations and drawbacks of narrow bandwidth and low throughput and larger antenna sizes.

4 Simulation Results

The top view of the proposed antenna is shown in Fig. 1, where the dimension is considered according to the design parameters in Sect. 2. Figure 2 shows the bottom view of the proposed antenna, where slit is etched off from the ground of the proposed antenna.

The SIW structures help in achieving multiband characteristics, the height and the inner radius of the SIW are 1.6 and 0.5 mm, and copper is used as the material of the via. The side view of the vias induced into the substrate is shown in Fig. 3, and the two vias are distanced apart with a constant called as K , where the value of K is 20 mm.

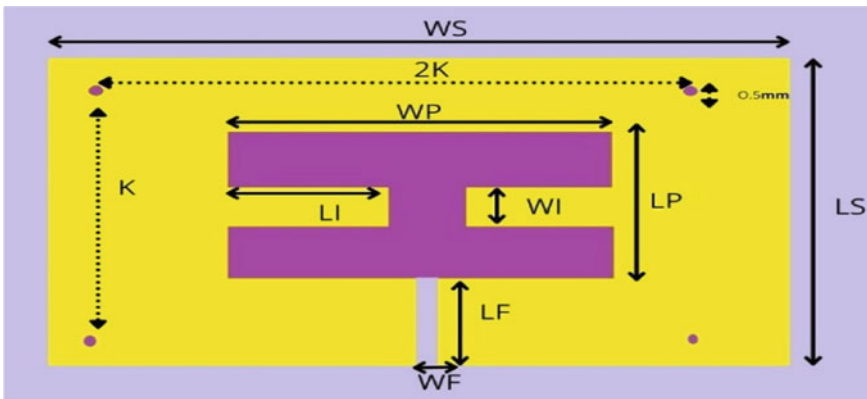


Fig. 1 Top view of the proposed antenna

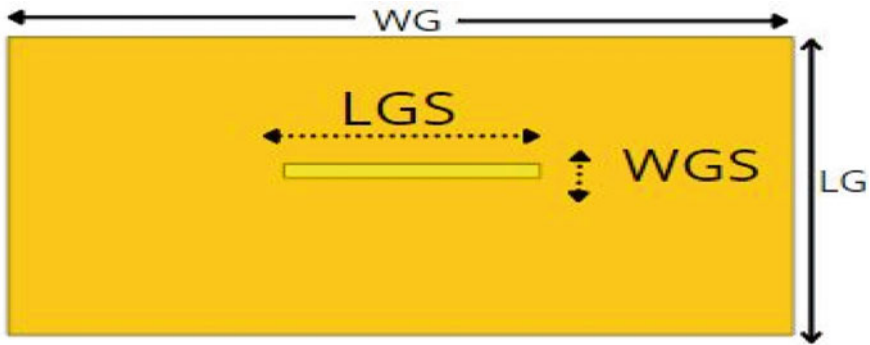


Fig. 2 Bottom view of the proposed antenna

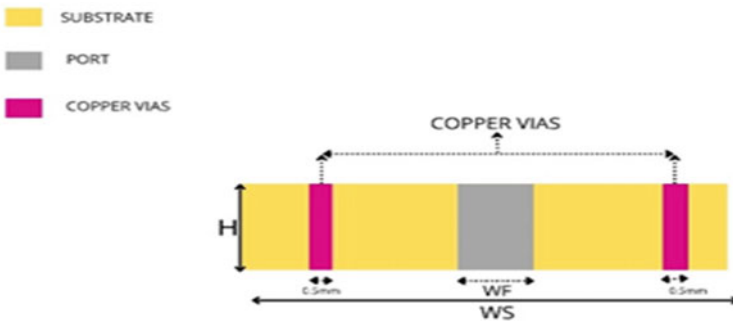


Fig. 3 Side view of the antenna

4.1 Frequency Analysis

The return loss characteristics of the proposed and existing antennas are compared and are presented in Fig. 4. The proposed antenna shows a descent enhancement in return loss when compared with the existing return loss, and the return loss and corresponding frequencies of the existing antenna are shown in Table 3.

The return loss and the corresponding properties of the proposed antenna are given in Table 4, The proposed antenna shows better return loss characteristics at 3.67, 5.15, and 6.19 GHz corresponding to the values of -37.7 , -17.10 , and -18.14 dB. The return loss is considered below -10 dB, and hence, 90% of the power is transmitted, and only 10% of the power is reflected back.

The return loss and the corresponding properties of the proposed antenna are given in Table 4, The proposed antenna shows better return loss characteristics at 3.67, 5.15, and 6.19 GHz corresponding to the values of -37.7 , -17.10 , and -18.14 dB. The return loss is considered below -10 dB, and hence, 90% of the power is transmitted, and only 10% of the power is reflected back. Large bandwidth helps in enhanced throughput and superior rate of transmission proving in more efficient transmission

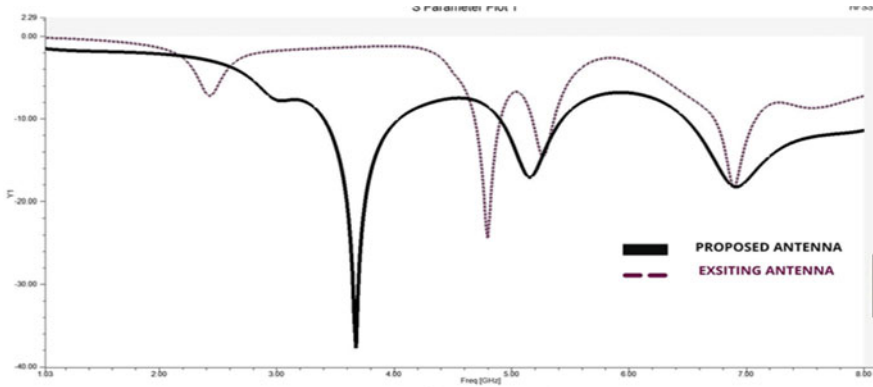


Fig. 4 Return loss of the proposed antenna

Table 3 Output parameters of the existing antenna

Existing antenna	
Return loss (dB)	Frequency (GHz)
- 7.15	2.41
- 24.41	4.79
- 14.57	5.26
- 18.22	6.92

Table 4 Output parameters of the simulated antenna

Simulated antenna	
Return loss (dB)	Frequency (GHz)
- 37.7	3.67
- 17.10	5.15
- 18.14	6.19

and receiving of the signal provided with low exposure to the false alarm and unclear clutter signatures. The bandwidths obtained for the proposed antenna are 0.63, 0.53, and 1.52 GHz accordingly.

The width of the feedline WF is compared with various widths ranging from 2 to 3 mm, where the best results were shown when the width is at 2.5 mm with a decent return loss of 37.7 dB, - 17.10 dB, and - 18.14 dB at 3.67 GHz, 5.15 GHz, and 6.19 GHz, respectively. A graph showing the comparison of various widths is shown in Fig. 5.

The optimized width for the feedline is considered as 2.5 mm as it has shown best performance among all the other values of WF tabulated, though at 3 mm, the antenna shows a good bandwidth of 1.29 GHz the return loss is not compromising (- 13.19 dB) in such a fashion considering the return loss to be optimal at WF 2 mm and 2.2 mm, and the bandwidths are noted to be - 31.11 and 31.12, respectively,

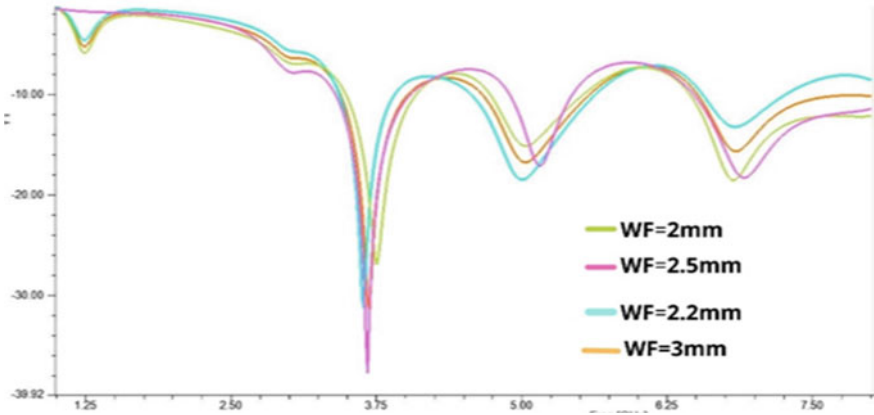


Fig. 5 Effects of the width of the transmission line on the frequency and bandwidth

Table 5 Illustration of effects of feedline on the output parameters

WF (mm)	Bandwidth (GHz)	S11 (dB)
2	0.41(3.89–3.48)	– 31.11
	1.01(5.60–4.49)	– 18.45
	0.68(7.25–6.58)	– 13.19
2.2	0.41(3.89–3.48)	– 31.22
	0.53(5.43–4.90)	– 18.45
	0.71(7.23–6.52)	– 13.19
3	0.57(4.00–3.44)	– 31.25
	0.91(5.58–4.67)	– 18.25
	1.29(7.81–6.52)	– 13.19

from the understandings and the information provided in the tabular rows and columns of Table 5.

The number of the vias is also chosen carefully by careful positioning of the vias apart from each other by a value of K , it results the symmetrical placing of the vias on either side of patch has shown optimal results than an odd number of vias positioning. This comparative analysis can be viewed in Fig. 6, though there are almost similar and satisfactory results at the same width of WF, yet the antenna with an even number of the slot as discussed in Sect. 2 better results in the form of return loss and bandwidth, where the antenna with an odd number of vias (5) shown only respective return loss values of – 25.0, – 17.04, – 23.08 dB.

The effects of change in the LGS are scrutinized for an obtained optimum measure of the LGS to be considered when the three following LGSs along with the rest of the design parameters is constant that are mentioned in Table 6 and depicted in Fig. 7. Various results of return loss and bandwidth are mentioned accordingly. The LGS is varied on the scale of 0.5 mm, where 0.5, 1.5, 2, and 2.5 mm are considered.

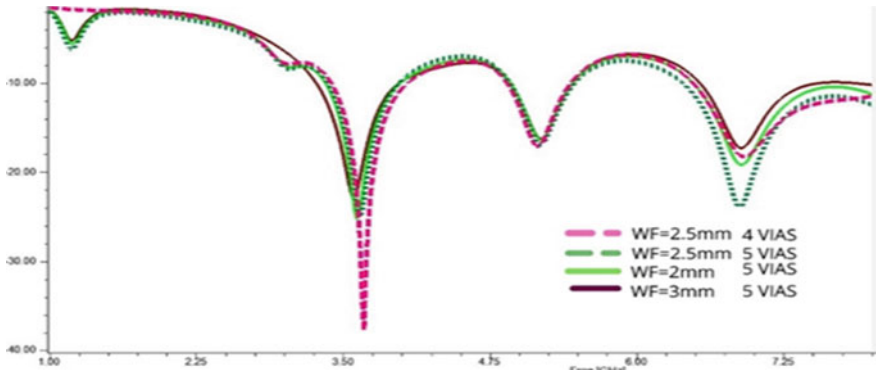


Fig. 6 Demonstration of effects of altering the number of SIWS on bandwidth

Table 6 Tabulation of relation between LGS and other parameters

LGS (mm)	Bandwidth (GHz)	S11(dB)
0.5	0.59	- 37.70
	0.53	- 17.04
	1.53	- 18.14
1.5	0.57	- 18.09
	0.58	- 17.24
	0.95	- 29.13
2	0.57	- 16.64
	0.50	- 16.67
	0.98	- 19.59
2.5	0.54	- 16.51
	0.50	- 15.89
	1.00	- 19.57

The radiation pattern of the existing and the proposed antenna is presented in Figs. 8 and 9, where the radiation patterns are shown in 2-dimension. The proposed antenna shows better results than the existing antenna; both the E-plane and H-plane of the radiation pattern are shown with long dash and solid line, respectively.

The pattern exhibited by the proposed and existing antenna is bidirectional in nature, whereas in the case of the H-plane, the existing antenna shows a partial side lobe which has overcome in the proposed radiation pattern. The values of the achieved patterns are presented in Table 7 and Fig. 10 show the radiation pattern of the proposed antenna. The color red shows that the particular antenna shows its best performance here whereas least at blue. The second-best performance is shown with the color orange, then slowly fading to yellow and finally blue, coming across the green on the scale moving from red to blue.

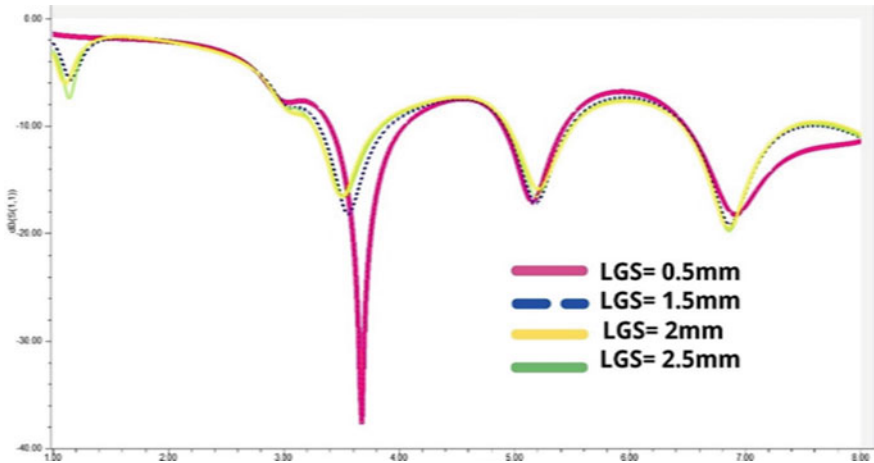


Fig. 7 Representation of dependence of bandwidth on the slit

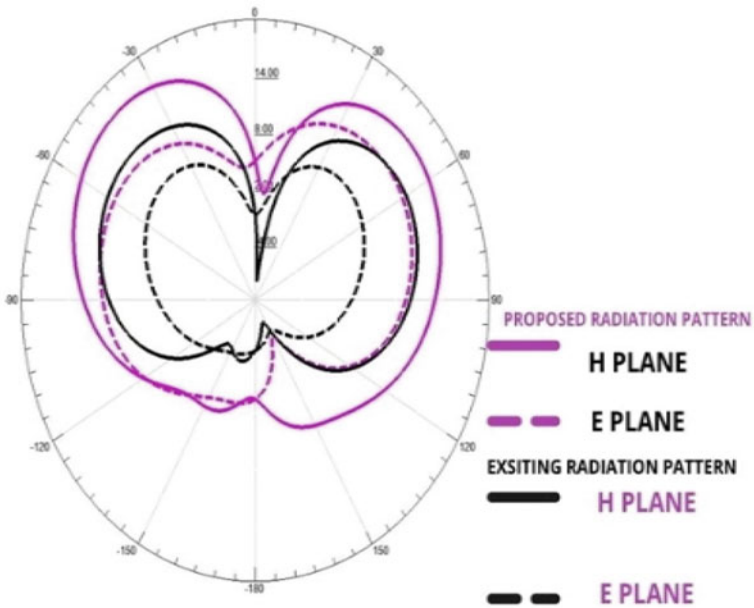


Fig. 8 Simulation discussion on the radiation pattern

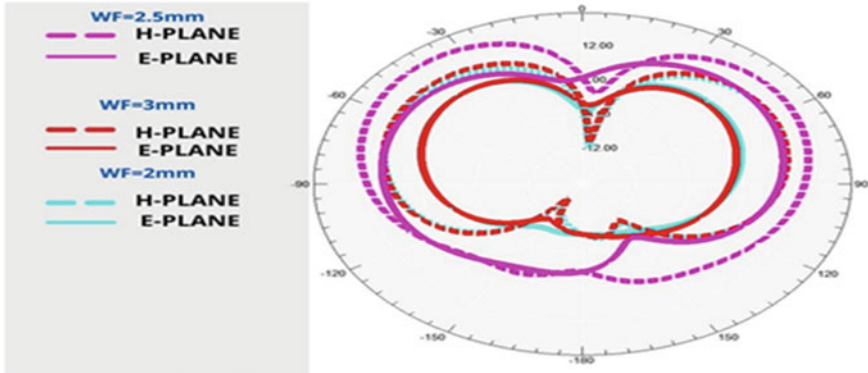


Fig. 9 Variation of patterns of radiation pattern with respect to WF

Table 7 Magnitude of the proposed antenna

	Maximum magnitude	Minimum magnitude
Proposed antenna	12.4	8

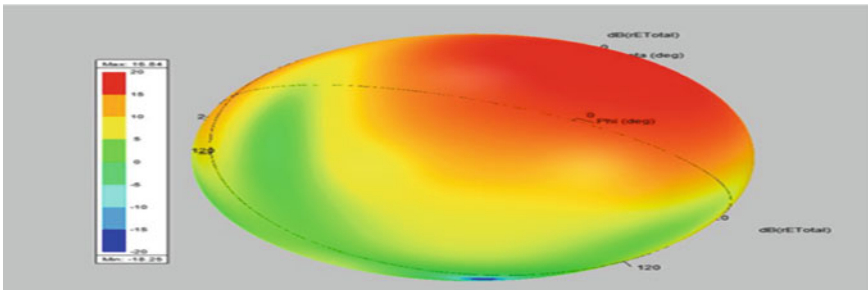


Fig. 10 Radiation pattern of the proposed antenna in 3D

5 Conclusion

The paper concentrated on enhancement of return loss and bandwidth of the antenna for multiband resonance application for radars especially in S and C bands. The most optimistic measures are tabulated in Table 1, and it is recurred from the understandings that the proposed antenna shows better performances across the parameters like return loss, bandwidth, and radiation patterns. One more parameter which shows a possible advantage is the achievement of symmetrical bandwidth with induction of SIW. The proposed paper can be further enhanced and optimized with respect to material and output parameters of both UWB and radar applications.

Future Advancements

The proposed antenna shows a wide range of significance and enhancements in the parameters of return loss, bandwidth, and radiation pattern but shows unlikely output in VSWR, and this can be addressed and provided with enhanced and optimized output in further advancements and publications.

References

1. Ramadhan FW, Ismail N, Lindra I (2019) Multi-band microstrip fractal antenna for S-band and C-band radar application. In: The 3rd international conference on electrical, telecommunication and computer engineering (ELTICOM)
2. Harikrishna P (2017) Wideband planar monopole antenna for S and C band monopulse radar application. In: IEEE WiSPNET conference
3. Zhang R, Xu X (2018) An irregular elliptical monopole antenna for ultra-wide-band (UWB) applications with dual notched bands. IEEE
4. Raveendra M, Saravanakumar U, Kumar GA, Suresh P, Pedapalli SPD (2021) A multiband substrate integrated waveguide antenna for Ku-band and K-band applications. In: 2021 6th international conference for convergence in technology (I2CT)
5. Raveendra M, Saravanakumar U, Kumar GA, Suresh P, Pedapalli SPD (2021) A broadband millimeter-wave SIW antenna for 5G mobile communication. In: 2021 5th international conference on trends in electronics and informatics (ICOEI)
6. Mishra M, Roy JS (2018) Investigations on the effect of mutual coupling in smart antenna using adaptive signal processing algorithm. In: International conference on applied electromagnetics, signal processing and communication (AESPC)
7. Munir A (2018) Shape enhancement and size reduction of UWB printed monopole antenna. IEEE
8. Darwehkar A, Dongaonkar P (2019) Design of a compact ultra-wideband printed elliptical ring monopole antenna for imaging radar application. In: 5th international conference for convergence in technology (I2CT)
9. Liu L, Liu C (2018) Slit-slot line and its application to low cross-polarization slot antenna and mutual-coupling suppressed tri-polarized MIMO antenna. IEEE
10. Kumar R, Ram Krishna RVS (2015) A dual polarized UWB slot antenna with kite shaped slot for high isolation. In: International conference on microwave, optical and communication engineering, Dec 18–20, IIT Bhubaneswar, India
11. Choi TH, Kim SS, Yoon YJ (2019) Wide-band slot Antenna on metal bezel for covering 28/39GHz in 5G communication system. IEEE
12. Gao S (2019) Dual-band digital beamforming synthetic aperture radar for earth observation. IEEE
13. Sharma MM (2019) Miniaturized quad band monopole microstrip antenna for deep space communication applications. IEEE
14. Su S-W, Hsieh Y-T (2017) Integration of very-low-profile slot antenna into notebook metal cover with narrow bezel. IEEE
15. Singh MS (2018) Pentagonal monopole patch antenna for UWB, X-band and Ku-band with parasitic resonator. In: 5th IEEE Uttar Pradesh section international conference on electrical, electronics and computer engineering (UPCON)

Road Network Extraction from Satellite Images Using Deep Learning



Yadav Maharaj and Jules-Raymond Tapamo 

Abstract Road network extraction plays a vital role in traffic analysis and safety monitoring, as well as in analysing, designing, and maintaining road structures. The task of road network extraction is tedious due to occlusions, shadows, non-road objects, and diversity of road network structures such as gravel, asphalt, sand. This study investigates road network extraction using deep learning techniques on high-resolution satellite images. More specifically, the combination of DenseNet and UNet deep neural networks is investigated for proficiency in road network extraction using high-resolution satellite images. Combining these complex neural networks has the potential to allow a deeper extraction of characteristics of a road network attribute, thereby increasing the accuracy in detecting and extracting all types of road networks. In addition, the use of a large dataset with very high-resolution images is to train the model, further increasing the accuracy of the model. The final combined neural network UNet-DenseNet-UNet, coupled with the high-resolution images, helps generate results that were better when compared to existing comparable models in literature which use deep learning techniques, with the intrinsic difference in the dataset used being the resolution of the images used to train the model.

Keywords Road network extraction · Deep learning · Machine learning · UNet · DenseNet

1 Introduction

The importance of up-to-date road maps can be seen in various important services such as GPS-based navigation systems for cities to enable accurate, reliable, and optimal routing directions for emergency vehicles as well as other users on such systems [6]. In the same light, new roads are constructed frequently, resulting in already extracted road map databases being outdated. Thus, ensuring that road maps

Y. Maharaj · J.-R. Tapamo (✉)

Discipline of Electrical, Electronic and Computer Engineering, University of KwaZulu-Natal, Durban 4041, South Africa
e-mail: tapamoj@ukzn.ac.za

are constantly updated timeously with the latest information is an important problem especially in today's world where people rely heavily on GPS navigation systems like Google Maps, Waze, etc. Updating road maps is also extremely beneficial in applications that include disaster management and urban planning. The distribution of roads has also been of crucial interest in recent scene understanding research works. Manual road extraction is time strenuous, and the data in GIS systems typically are out of date and are inaccurate due to human error in the process [6, 12, 14, 18]. Road network extraction for the purpose of updating road networks in geospatial databases has become an important data source, as well as an important research topic in the vast field of high-resolution image processing and remote sensing [2, 8, 17].

In recent years, various road network extraction techniques have been proposed. These consist of supervised and unsupervised learning and take advantage of the geometric, photometric, and textural features for the purpose of road network extraction, with unsupervised learning methods making use of mainly clustering methods to extract road networks from the high-resolution remote image by means of using different road classes. Machine learning approaches for road networks extraction remain a difficult problem due to the relatively low attribute information available and the prevalent missing values issue. This lack of attribute information can be alleviated by exploiting the network structure into the learning process. Complexity of roads structure, vegetation, activities on roads, and arrangements of buildings can be observed from high-resolution images [8], as well as sparse context of the road networks in remote sensing imagery. These issues make the task of road networks extraction from high-resolution imagery increasingly difficult.

In this paper, the use of deep learning techniques to extract road networks from high-resolution satellite images is explored. The corresponding expert labelled data known as masks provides the location of the pixel that corresponds to a road. Many previous attempts have failed in providing reliable and robust results, which can be attributed to the inaccurate labelling of expert labelled masks as well as the use of relatively small datasets which have in average 200 images, with low- to medium-resolution images 256×256 – 512×512 pixels. Added to the poor performance is the type of networks used to train models. Many machine learning models and simple deep learning models would use too few layers, resulting in very little information being extracted from the pixels contributing to road network, and this will mean that the network is unable to extract enough features from the pixel resulting in an overfitted model, which performs poorly on other datasets.

In order to address these issues, the proposed solution is designed as follows:

- A combination of two deep neural networks is used, i.e. DenseNet and UNet. This allows for a more meaningful and substantial features to be extracted from the images resulting in a highly robust model.
- A vast dataset covers vast road networks that include various surrounding landscapes around the road network, and these include sand, trees, grass, mountains, water, and buildings.

The proposed approach is similar to Xu et al. [14]. Their model combines DenseNet and UNet with global and local attention units.

The rest of the paper proceeds with the following organisation. Section 2 presents the research background; it covers and investigates similar models that have been proposed in the literature to solve the problem of road network extraction. In Sect. 3, the techniques and methods used to efficiently and accurately extract road networks from high-resolution images are discussed. Section 4 focuses on the techniques that are used in the experiment and then discusses the results achieved with the proposed method. Conclusion and future works are given in Sect. 5.

2 Background and Related Works

Traditionally, road network extraction methods were extremely cumbersome and riddled with errors and inaccuracies due to human error in the extraction process [12]. Road network extraction tasks proposed in literature include unsupervised and supervised learning techniques [18]. These methods rely on classification techniques to extract road networks from the high-resolution and remote sensing imagery [18]. Amongst unsupervised methods, clustering algorithms have been the most commonly used approaches for road network extraction. A semi-automatic technique utilising mean shift to detect roads was proposed by Miao et al. [5]. This method uses road seed points to extract the initial point and then uses a threshold to separate the identity of the pixel as either road or non-road. Van de Weijer et al. proposed an approach that achieves remote scene classification by combining RGB streams and texture coded maps to act as a dual stream deep network architecture [1]. Sirmacek suggested extracting road networks by combining probability and graph theory, which proved to be a complicated but effective method compared to other unsupervised learning methods [11]. Zhang et al. proposed a method to extract roads using object-based methods [13]. This method is made up of three steps: textural information extraction, road extraction, and finally post-processing step.

Unsupervised methods offered results that were subpar and are generally out-classed by supervised learning methods based on accuracy [13]. Some supervised learning techniques used include deep learning, support vector machines, random decision forests, and combinations of these techniques. The aforementioned methods extract road networks from images based on training models using labelled samples [10]. These samples are expert labelled data, also known as ground truths. Simler proposed a 3-class SVM using spectral and spatial features of the roads [9]. Ban proposed a model based on the combination of MAP-MRF classifiers that perform a good semantic segmentation on very high-resolution aerial images [16]. A SVM classifier that made use of visual features within remote sensing imagery to extract road networks, such as edge length, intensity and gradient, was proposed by Yager et al. [15]. Ling et al. proposed a method that uses MRF to extract the roads and a combination of SVM and fuzzy c-mean (FCM) to perform semantic segmentation [19]. It has been established in the literature that fully connected convolutional network produces outstanding results due to its model behaviour [4]. Kestur et al. proposed U-shaped FCN (UFCN) method to extract road network by feeding the

input image through a stack of convolutions followed by deconvolutions with skip connections [3]. Mnih used neural networks with specific pre-training and post-training algorithms, using restricted Boltzmann machine (RMB) for pre-training for high maximum likelihood intractability and custom post-processing procedure that relies on dependencies present in nearby map pixels to significantly improve the predictions [6]. Xu et al. used deep learning techniques to extract road networks. The method combined DenseNet and UNet models and using local attention units (LAU) and global attention unit (GAU) that help improve the accuracy of the model [14].

As mentioned earlier, this paper proposes a combination of using DenseNet and UNet to extract road networks, but as an improvement of the work done by Xu et al. [14]. They proposed a GL-Dense-UNet model, which makes use of local and global features in the pre-processing and post-processing of the DenseNet, followed by the UNet model as the classifier. Their experiment uses a dataset of 200+ images, of original resolution 512×512 pixels; however, these images were then clipped to 256×256 pixels. Overall accuracy achieved was over 97%. The goal of this paper is to further improve this performance, using a Dense-UNet model coupled with a larger dataset of 1117 images with higher resolution of 1500×1500 pixels, covering vast surroundings.

The proposed method offers advantages in intensity variation due to the following properties of the method, motion detection within images depends on the change of the intensity value regardless of the intensity value itself, coupled with built in binary labelling from each image, through the use of seeds. These seeds are selected pixels which provide information that is used in the binary labelling process.

3 Materials and Methods

This paper investigates the use of deep neural networks (DNNs) to extract road networks from high-resolution satellite imagery. Learning-based approaches are not new, and many have proven to have worked satisfactorily. These solutions, however, performed poorly on different datasets. A large dataset covering various types of surroundings (sand, mountains, grass, trees, buildings, waterways, rivers, etc.), occlusions (cars, shadows, etc.), and roads (different widths and lengths, gravel, asphalt, sand, etc.) was used to alleviate the issue of overfitting and improve the performance of the extraction model on other datasets. Raw satellite images were fed into DNNs and cropped into 500×500 slides with the corresponding ground truth mask. The resulting cropped 500×500 images were then input into the neural network for feature extraction. In this paper, the combination of DenseNet and UNet deep neural networks were used. Images were fed firstly into a UNet, followed by the DenseNet and finally fed into another UNet. The output of the UNet is a matrix of predictions determining if a pixel corresponding to the raw image input to the network is a road or not.

A general overview of what is being achieved is the following: Let I be a high-resolution/remote sensing image and let G be the extracted road network from the image I , where G is a $n \times m$ pixels image having $G(i, j)$ equals 1 whenever pixel at location (i, j) in the image I corresponds to a road pixel and $G(i, j)$ equals 0 otherwise. The goal of this paper is to learn the prediction of whether a pixel at a location (i, j) is on a road or not given the raw image S , which can be seen as the conditional probability, $p(G(i, j)|I)$, of a pixel on a location (i, j) of an input image, S being on a road or not [6]. In the context of very high-resolution aerial/satellite imagery, each pixel represents a square ground plot that could represent area between many centimetres and a few metres. Within the same light, the typical region of interest and major application of road extraction are for an entire city. Hence, a problem arises in making predictions for millions of pixels on satellite images [6].

3.1 Neural Network Implementation

As stated earlier, the process of road network extraction from images lies within the bounds of semantic segmentation. A common and typically used convolutional neural network for such a task is the UNet model.

The UNet model consists of an architecture that was originally designed for the complex task of biomedical image segmentation by Ronneberger et al. [7]. The complete architecture of the model can be split into two parts, the first part, being the encoder that focuses on utilising a two major steps that use convolution layers, followed by pooling operations which is essentially the feature extraction process of the model, which is then fed into the second part of the UNet model architecture known as the decoder [7]. The decoder enables the localization by performing transposed convolutions [7]. The layers of the network are fully connected. Figure 1 shows the general overview of the UNet architecture multi-channel feature maps are denoted by the blue boxes in Fig. 1. The numbers on top of these boxes indicates the number of channels, with the numbers on the bottom left of the box indicating the x - y size. The arrows denote the different operations as input [7].

Another important deep neural network used in image segmentation is the dense neural network, or DenseNet. A DenseNet is a type of convolutional neural network that utilises dense connections between layers, through dense blocks, where all layers are connected directly with each other. Each layer within the network receives, as additional input parameter, the output of every preceding layer, inadvertently meaning that the layer sends forth its own output feature map to all layers that follow thus highlighting the feedforward nature of the model [14] which has been shown that by shortening the connections between layers that are close to the input and output, the accuracy of convolutional layers can be increased [14], and this is the idea behind the DenseNet structure.

Figure 2 shows the general overview of how the DenseNet that is connected.

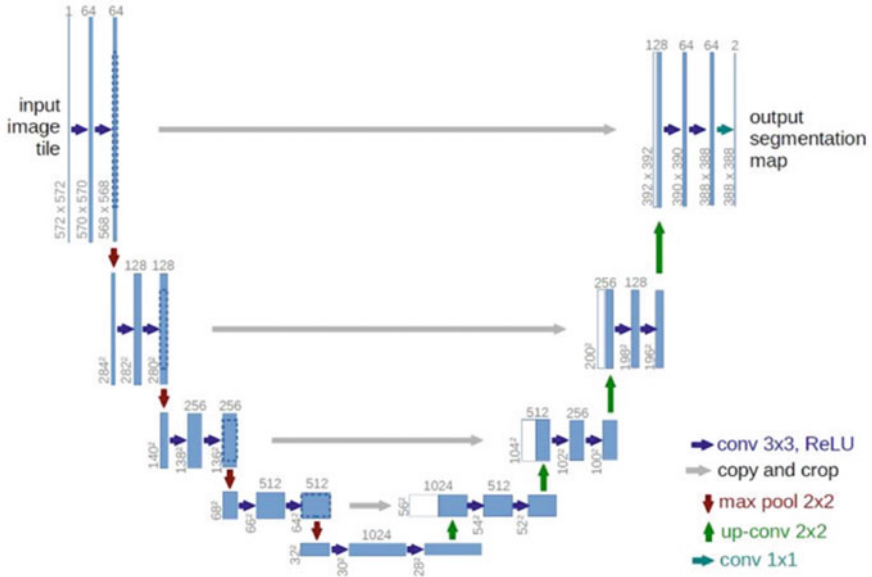


Fig. 1 UNet model architecture [7]

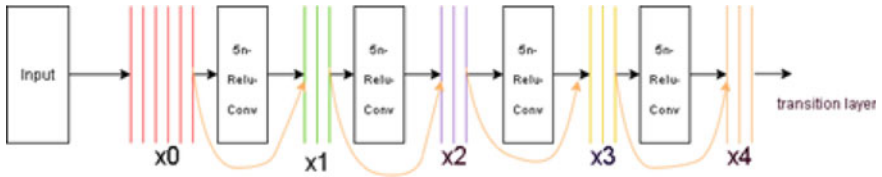


Fig. 2 Five-layer dense block network, each layer takes the previous layer as input

The structure of the DenseNet model can be seen in Fig. 2. Each colour represents a different layer, where one can see the input entering the first layer x_0 , enters through ReLU activation convolution, and the result is combined with the output of x_0 and fed into as input to the second layer x_1 . This process of feedforwarding is carried through the till the last layer x_4 , where the output is then fed as a transitional layer or to output layer, and it represents the end of the five-layer dense block.

3.2 Experimental Procedure

The dataset of raw images and corresponding ground truth/ expert labelled data was used as input to train the model. Figure 3 shows the parameters of the layers used in the complete training model, which follows the order

Input image \rightarrow Unet \rightarrow DenseNet \rightarrow UNet \rightarrow Output image

In the proposed model, neural network used is composed as follows: DenseNet provides state-of-the-art feature extraction. The input image of 1500×1500 pixels was cropped into 500×500 slides, and these slides were fed into the five-layer DenseNet, with the first layer starting off with 64 nodes, and subsequently increased to 128, 256, 512, and 1024 up until the fifth layer of the DenseNet. Each of the first four layers consisted of, Conv2D, Normalisation, Activation, Pooling 2D, dropout, Merge, and Conv2D, and the fifth layer did not have a second Conv2D attribute as its output was then fed as input to the second UNet layer. All layers were activated with the ReLU activation function. The input image of 1500×1500 pixels was cropped into 500×500 slides; these slides were then fed into the first UNet layer, of which the output was fed into the DenseNet; each layer of the UNet up until the fourth consists of Conv2D, Upsampling, Merge, Conv2D, Normalisation, ReLU activation, Conv2D, Dropout, Merge. The final layer of UNet uses the sigmoid Activation. The output of the second UNet is then a matrix of the prediction of whether each pixel is a road.

The dataset was split into 8:2 training: validation ratio model. The model was run for 100 epochs, using the Adam optimiser at a learning rate of 0.0001 and a batch of 4 images. Binary cross entropy loss function was used due to the nature of the classification with the main metric in evaluation being the accuracy. Edge enhancement filters were used in the Conv2D function for each layer, with kernel initialisation of *he_normal* thus increasing the extraction performance.

To gauge the performance of the model for road network extraction, precision (P), recall (R), $f1$ -score, and overall accuracy (OA) were used. Equations 1–4 show the respective formulae of how these important metrics are calculated, where *True positive* (TP), *False Positive* (FP), *False Negative* (FN), and *True Negative* (TN).

$$P = \frac{TP}{TP + FP} \quad (1)$$

$$R = \frac{TP}{TP + FN} \quad (2)$$

Equations 1 and 2 show the formulae for precision and recall, and these two metrics are then used to calculate the $f1$ -score shown in Eq. 3.

$$F\text{-score} = 2 \times \frac{R \times P}{R + P} \quad (3)$$

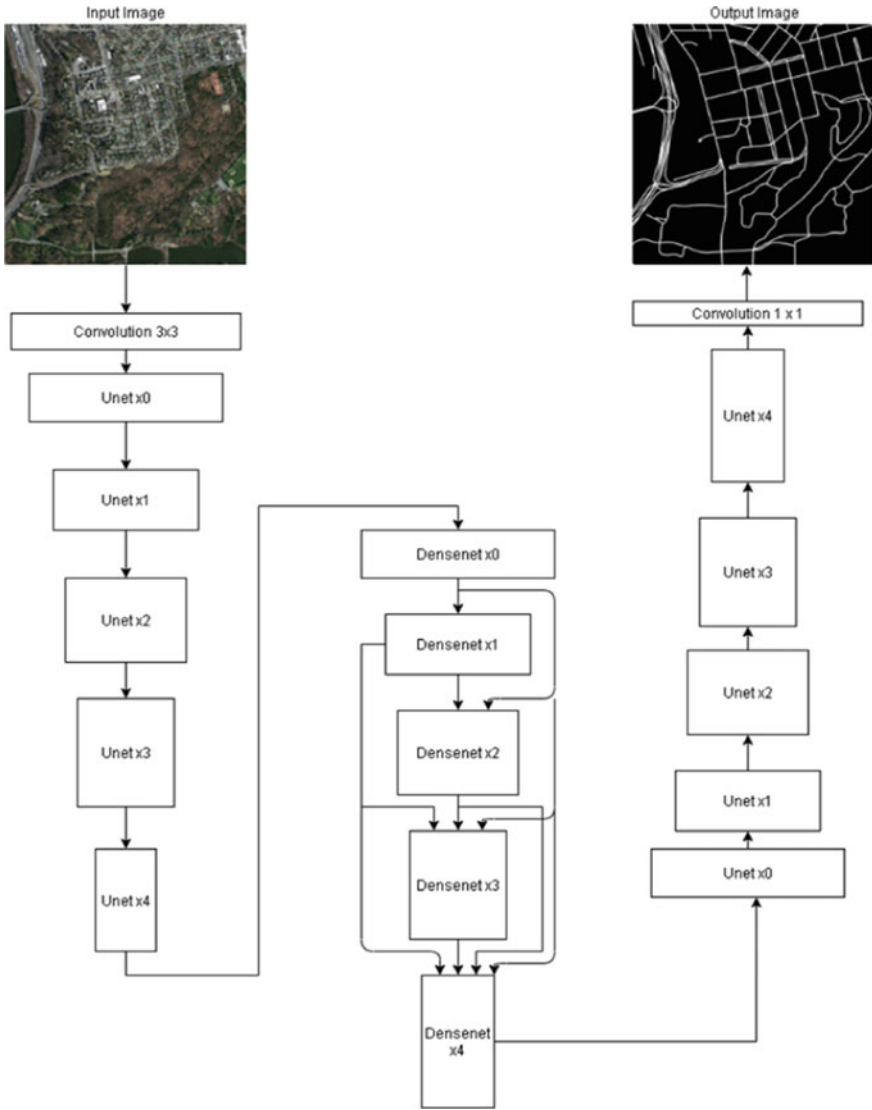


Fig. 3 Architecture used for training Dense-UNet model

$$OA = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

Equation 4 shows the formula for the overall accuracy of the model based on the validation set and the test set.

4 Experimental Results and Analysis

4.1 Dataset

Road network extraction from any image falls under the field of semantic image segmentation. Typically, image segmentation requires at least 100 images to produce decent results. In this paper, the Massachusetts Roads Dataset of 1117 images was used to train a model for road network extraction and 200 images were used to test the model. Ground truth images were labelled manually by the referenced maps done by Minh the author of the dataset, which is available publicly on the CS Toronto website [6].

The dataset covers urban, suburb, and rural areas c“overs an area of over 2600 square kilometres. Each image is 1500×1500 pixels in size, covering an area of 2.25 square kilometres”[6]. With many types of obstructions on the road networks such as trees, vehicles and many types of surroundings such as sand, grass, vegetation, buildings, and water. The aforementioned features of the dataset combined with the various widths of roads used result in a highly vast dataset enabling a trained model that works well for many other different datasets.

4.2 Results and Discussion

Figures 4a, b show a sample image of the input raw road map image and corresponding ground truth dataset used. The 1500×1500 pixel images were fed into the DNNs as and were cropped into 500×500 slides with the corresponding ground truth mask. The resulting cropped 500×500 images were then fed into the neural network for feature extraction.

The model was trained first using the 1117 image dataset. Post-training results on the validation split resulted in impressive results of validation loss: 0.2062 and validation accuracy of 98.21%. The model was then tested on a different test dataset of 220 images. The results for all metrics based on the performance of the model on the test dataset are shown in Table 1. The accuracy on the test dataset boasts an impressive 98.76%, with an $f1$ -score of 98.22%. Figure 5 shows the graph of accuracy for training and validation vs the number of epochs, and it is evident that



Fig. 4 **a** Input raw image of road map, **b** Corresponding ground truth image

Table 1 Observed results from trained UNet-Dense-UNet model

Model	Precision (%)	Recall (%)	<i>f</i> 1-score (%)	Accuracy (%)
DenseNet	98.89	97.56	98.22	98.76

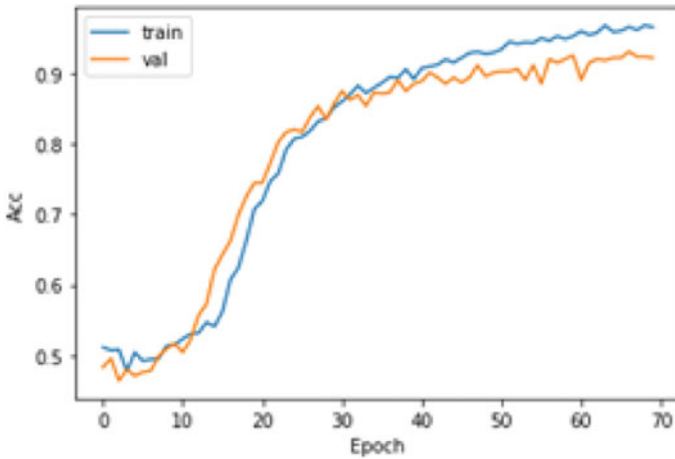


Fig. 5 Accuracy versus epoch graph, for training and validation

towards the last 30 epochs, i.e. epoch 70–100 that the training accuracy stabilised around 98%.

To show the effectiveness of this model on the high-resolution images, with the various types of surrounds, such as buildings vegetation water and sand, a sample

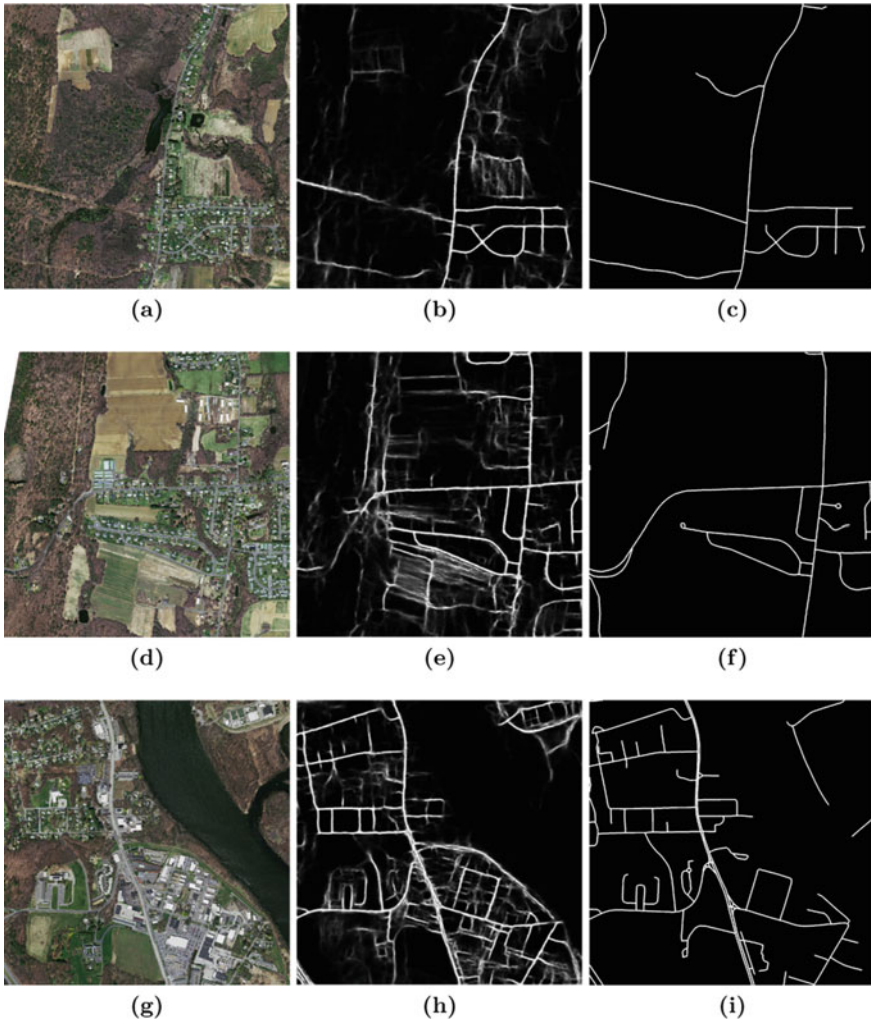


Fig. 6 Results of extraction for various cases

for each case was chosen, and the results after passing the images through the model are shown below in Figure 6. Figure 6a shows the raw image for vegetation and sand as surrounding areas, with corresponding Fig. 6b showing the extracted road network using the proposed model, and Fig. 6c shows the ground truth data for Fig. 6a. Figures 6d–f are, respectively, the raw image for sand, vegetation, and buildings as surround areas, the extracted network for the raw image, and ground truth data for Fig. 6d. Figures 6g–i show, respectively, the raw image for buildings, vegetation, and water as surrounding areas, the resulting extracted network, and the ground truth data for Fig. 6g.

Table 2 Comparison of results between proposed model and similar experiment

Model	Precision (%)	Recall (%)	<i>f</i> 1-score (%)	Accuracy (%)
Proposed	98.89	97.56	98.22	98.76
Xu et al.[14]	97.83	97.35	97.59	97.82

It is clear in Fig. 6 that the model is accurate in road network extraction. However, the listed problems of vegetation, sand, and waterways still affect the model. This can be improved by training the model for more epochs.

When comparing the proposed model to that of Xu et al. [14], their model trained for 10000 iterations, whereas the proposed model ran for 100 epochs. Table 2 compares the performance metrics between the two models. It is evident that the overall metrics of the proposed model by slight of margins performed better than the experiment done by Xu et al. [14]. This is largely attributed to the larger dataset, with high-resolution images and the UNet-DenseNet-UNet model, which offers increased performance on high-resolution datasets.

5 Conclusion

This paper has proposed a model that uses deep learning techniques to extract road network from very high-resolution satellite images. The proposed model is composed of two deep convolutional networks DenseNet and UNet connected in the order $Unit \rightarrow DenseNet \rightarrow UNet$. It was trained using 1117 images, which is a significantly larger dataset than that used in existing models and experiments. The use of the DenseNet and UNet combination was chosen to fulfil the need of deeper feature extraction for higher accuracy in correctly identifying and extracting road networks from the images. An overall accuracy of 98.76% was achieved which is close to a 1% improvement compared to similar experiment. However, this can be improved further. Without changing the combination network, increasing the number of epochs would drastically increase the recall and *f*1-score of the model, in turn increasing the accuracy. This model works relatively well in extracting roads from all kinds of surroundings and different angles.

For future research, we aim to improve the accuracy of this experiment, as well as using the output of the model proposed in this experiment as input for a classification model to identify road networks. The scope for such a model is vast and can be used in many applications, such as traffic analysis, safety, and navigation-based systems.

References

1. Anwer RM, Khan FS, van de Weijer J, Molinier M, Laaksonen J (2018) Binary patterns encoded convolutional neural networks for texture recognition and remote sensing scene classification. *ISPRS J Photogram Remote Sens* 138:74–85
2. Cheng J, Ding W, Ku X, Sun J (2012) Road extraction from high-resolution sar images via automatic local detecting and human-guided global tracking. *Int J Antennas Propag* 2012
3. Kestur R, Farooq S, Abdal R, Mehraj E, Narasipura OS, Mudigere M (2018) UFCN: a fully convolutional neural network for road extraction in rgb imagery acquired by remote sensing from an unmanned aerial vehicle. *J Appl Remote Sens* 12(1):016020
4. Maggiori E, Tarabalka Y, Charpiat G, Alliez P (2016) Convolutional neural networks for large-scale remote-sensing image classification. *IEEE Trans Geosci Remote Sens* 55(2):645–657
5. Miao Z, Wang B, Shi W, Zhang H (2014) A semi-automatic method for road centerline extraction from vhr images. *IEEE Geosci Remote Sens Lett* 11(11):1856–1860
6. Mnih V, Hinton GE (2012) Learning to label aerial images from noisy data. In: *Proceedings of the 29th international conference on machine learning (ICML-12)*, pp 567–574
7. Ronneberger O, Fischer P, Brox T (2015) U-net: Convolutional networks for biomedical image segmentation. *Corr abs/1505.04597* (2015). [arXiv:1505.04597](https://arxiv.org/abs/1505.04597)
8. Sghaier MO, Lepage R (2015) Road extraction from very high resolution remote sensing optical images based on texture analysis and beamlet transform. *IEEE J Sel Topics Appl Earth Observ Remote Sens* 9(5):1946–1958
9. Simler C (2011) An improved road and building detector on VHR images. In: *2011 IEEE international geoscience and remote sensing symposium*. IEEE, pp 507–510
10. Tian S, Zhang X, Tian J, Sun Q (2016) Random forest classification of wetland landcovers from multi-sensor data in the arid region of xinjiang, china. *Remote Sens* 8(11):954
11. Unsalan C, Sirmacek B (2012) Road network detection using probabilistic and graph theoretical methods. *IEEE Trans Geosci Remote Sens* 50(11):4441–4453
12. Wang J, Song J, Chen M, Yang Z (2015) Road network extraction: a neural-dynamic framework based on deep learning and a finite state machine. *Int J Remote Sens* 36(12):3144–3169
13. Wang W, Yang N, Zhang Y, Wang F, Cao T, Eklund P (2016) A review of road extraction from remote sensing images. *J Traff Transp Eng (English Edn)* 3(3):271–282
14. Xu Y, Wu L, Xie Z, Chen Z (2018) Building extraction in very high resolution remote sensing imagery using deep learning and guided filters. *Remote Sens* 10(1):144
15. Yager N, Sowmya A (2003) Support vector machines for road extraction from remotely sensed images. In: *International conference on computer analysis of images and patterns*. Springer, pp 285–292
16. Yousif O, Ban Y (2014) Improving sar-based urban change detection by combining map-mrf classifier and nonlocal means similarity weights. *IEEE J Sel Topics Appl Earth Observ Remote Sens* 7(10):4288–4300
17. Zhang J, Chen L, Wang C, Zhuo L, Tian Q, Liang X (2017) Road recognition from remote sensing imagery using incremental learning. *IEEE Trans Intell Transp Syst* 18(11):2993–3005
18. Zhang Y, Wang X, Zeng P, Chen X (2011) Centrality characteristics of road network patterns of traffic analysis zones. *Transp Res Rec* 2256(1):16–24
19. Zhu DM, Wen X, Ling CL (2011) Road extraction based on the algorithms of mrf and hybrid model of svm and fcm. In: *2011 international symposium on image and data fusion*. IEEE, pp 1–4

An Evolutionary Online Motion Planning of Car-Like Mobile Robots with Velocity Obstacles



S. Ramabalan , V. Sathiya , and M. Chinnadurai 

Abstract This paper presents an evolutionary algorithm-based online motion planning method for a car-like mobile robot which operates in a dynamic environment with velocity obstacles. More number of complications and difficulties in online motion planning (dynamic environment and obstacles, complicated motion dynamics, and decision-making time are in nanoseconds, etc.) have to be overcome by the method for planning an optimum path. Extensive works have been done in developing a novel method for robot motion planning in dynamic environments, and the research gap identified is a need of an efficient planner for online motion planning. So, to improve the efficiency, this paper proposes a method based on improved heterogeneous differential evolution (IHDE) algorithm. An experimental verification proved the worthy of the proposed algorithm.

Keywords Car-like mobile robot · Online motion planning · Velocity obstacles · IHDE algorithm

1 Introduction

Online motion planning in a dynamic environment is very difficult and complicated. This is because of velocity obstacles and ever-changing environment. Motion planner has to do path planning and velocity planning in concurrent mode. Finding an optimum path or route is termed as path planning. Finding the safest velocity for the robot according to velocity of moving obstacles is termed as velocity planning. The

S. Ramabalan (✉)

Department of Mechanical Engineering, E.G.S. Pillay Engineering College, Nagapattinam, Tamil Nadu 611002, India
e-mail: cadsrb@gmail.com

V. Sathiya

Department of Electronics and Communication Engineering, E.G.S. Pillay Engineering College, Nagapattinam, Tamil Nadu 611002, India

M. Chinnadurai

Department of Computer Science and Engineering, E.G.S. Pillay Engineering College, Nagapattinam, Tamil Nadu 611002, India

planner has to work in both path finding and velocity finding planes. Path finding is a kinematic problem. It considers the kinematics and geometrical details of both the robot and the obstacles. It involves in finding a shortest path. But, velocity finding is a dynamic problem. It considers the robot dynamics and constraints of motors (actuators). It involves in finding a safest and energy saving path.

2 Literature Survey

The main difficulty in online motion planning is within a nanosecond, the planner has found and take decision on robot path and robot velocity [1]. Extensive works have been done in developing a novel method for robot motion planning in dynamic environments [2]. Two classifications are in motion planning: They are global planner and local planner. The global planner knows all the details about the environment around the robot. So, it gives the best motion plan. But, the local planner does not know all or partial information about the environment around the robot. So, it gives a local motion plan and not a global plan. Both planners consider environmental modeling, motion searching procedure, and optimization criteria. Classic approach techniques used in the literature are, namely framework space approaches, free space approaches, cell decomposition approaches, topological methods, probabilistic roadmap methods, Gaussian process inference [3], sparse roadmap [4], etc. Few framework space approaches are visibility graph, Voronoi graph, tangent graph, etc. Heuristic approaches are Dijkstra algorithm, A* algorithm, D* algorithm, etc. Artificial intelligence algorithms are artificial neural network, fuzzy logic, ant colony optimization, particle swarm optimization, genetic algorithm, differential evolution [5], NSGA-II [5], ABC algorithm, firefly algorithm, memetic algorithm, artificial immune system, Tabu search [6], ANFIS [6], etc. Artificial potential field, behavior decomposition method, case-based learning method, and rolling window algorithm are frequently used classical local planning methods [7]. Sharma et al. [8] and Jain et al. [9, 10] deployed differential evolution for path planning with improved results.

2.1 Gap Identification

All methods have their own merits and demerits. But, in the race of finding best optimal motion plan, artificial intelligence techniques such as swarm and evolutionary algorithms are front runners. Few efficient algorithms were developed for online motion planning. But, their quality and efficiency are to be improved further, in finding the best motion plan. More experimental verifications have to be done for car-like mobile robot in online mode.

3 Proposed Methodologies

To improve the efficiency and quality of online motion planner, this research work proposes an evolutionary algorithm improved heterogeneous differential evolution (IHDE) algorithm. It is a new variant of DE developed by the authors. It overcomes more limitations of existing standard DE and heterogeneous differential evolution (HDE) [11].

3.1 Proposed Algorithm

Differential evolution (DE) is a best one among the available evolutionary algorithms. It solves all types of real-world problems. Applications of DE can be found in all fields including engineering and science. The basic DE has few limitations. So, to improve it, many researchers proposed and used many variants of DE. One among them is HDE. But, it also has few demerits. They are (a) four schemes—DE/rand/1/bin, DE/rand/2/bin, DE/best/1/bin, and DE/best/2/bin were used for creation of a new solution in noisy vector. But, DE/TSDE, DE/current-to-best/1, and DE/rand-to-best/1 are best for creating noisy vector as per the literature. (b) Scaling factor (SF) is constant. (c) They tested their HDE for solving a simple problem having only one objective function. (d) HDE was not tested for solving a real-world problem. (e) The problem considered does not have any constraint.

This research work proposes an improved heterogeneous differential evolution (IHDE) for solving a real-world problem. The algorithm used and experimented for a car-like mobile robot motion planning in online mode. The proposed algorithm has following merits over HDE: (a) DE/rand/2/bin, DE/TSDE, DE/current-to-best/1, and DE/rand-to-best/1 are used. (b) A constraint optimization problem with seven constraints is solved by IHDE. (c) A real-world problem is solved by IHDE. (d) In IHDE, scaling factors both K and SF are randomly created during all iterations. (6) Also, the crossover method is improved one in IHMDE.

IHMDE Operators: The total no. of population is 100, and crossover constant (CC) value varied in between 0.1 and 0.95. Termination criteria is no. of generations = 100.

The optimization procedure used in this research work is: Step 1: All required information is given to IHDE algorithm. Step 2: IHDE computes the dynamic parameters such as torque of actuators. Also, it computes kinematic parameters such as position, speed, jerks, and vibration or acceleration of the robot. Step 3: IHDE gets information about the environment. With help of fuzzy logic controller, it computes the robot path and the velocity. Step 4: IHDE communicates the information to the controller.

4 Problem Description

A car-like mobile robot with two active back wheels and two supportive front wheels is fabricated and tested in this research work [12]. The geometrical representation is shown in Fig. 1. The travel aim of the robot is to reach its end point (E) safely without kitting any obstacle from a start point (S) as in Fig. 2. Four velocity obstacles were considered around the robot. Velocity obstacles are battery operated toys in Fig. 2. The velocity obstacle 1 is yellow color car. The speed of moving obstacle 1 (MO1) is 0.1 m/s. The velocity obstacle 2 is red color lorry. The speed of moving obstacle 2 (MO2) is 0.07 m/s. The velocity obstacle 3 is orange color lorry. The speed of moving obstacle 3 (MO3) is 0.05 m/s. The velocity obstacle 4 is orange color JCB truck. The speed of moving obstacle 4 (MO4) is 0.03 m/s.

Fifth-order polynomial curve is used to model the robot path. A fuzzy logic-based controller is used to assist the robot controller for doing obstacle avoidance.

Motion planning is an optimization problem. This research problem optimization model is:

Two objective functions are to be minimized, and they are robot path length ($F_1(P) = \sum \text{segments' length}$) and travel time ($F_2(P) = \sum \text{segments' travel time}$). A combined objective function (COF) = $F_1(P) + F_2(P) * 0.1 F_1(P) + F_2(P)$ is used.

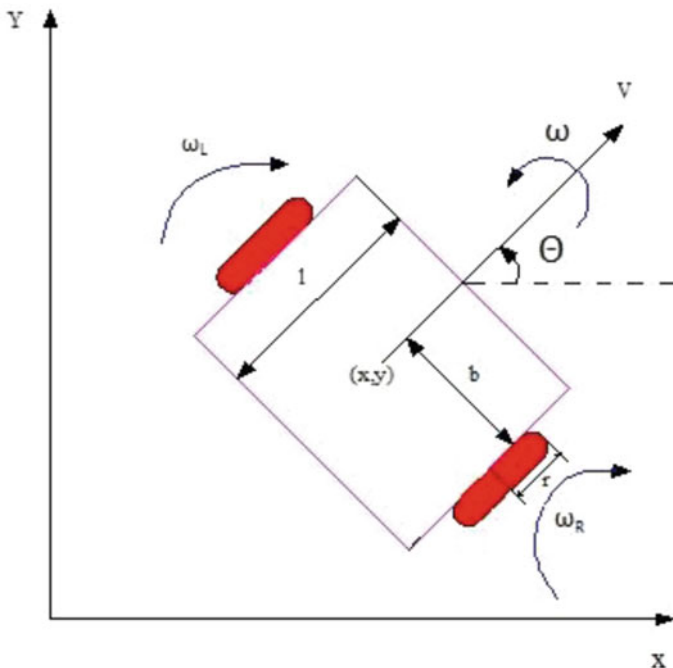


Fig. 1 Geometrical representation of car-like robot

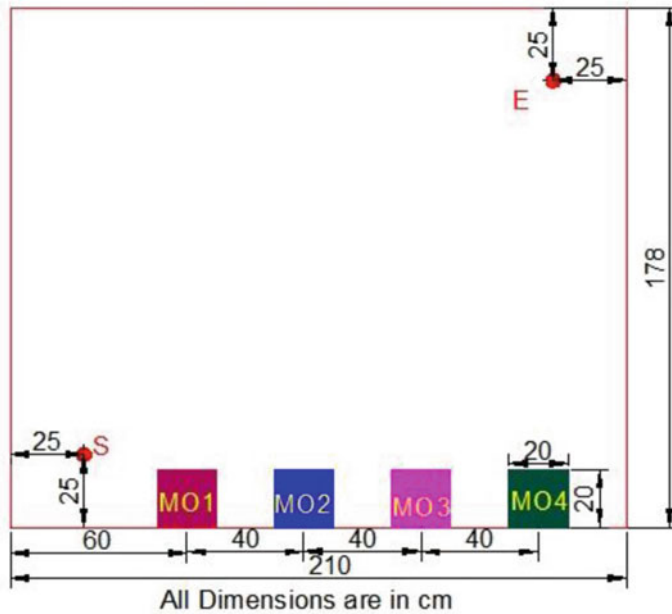


Fig. 2 Dynamic environment around the robot

The constraints are: (1) geometrical constraints—initial and final positions of the path. (2) Obstacle avoidance constraint—distance between robot and obstacles should be above 3 cm. (3) Kinematic constraints—speed of robot at start and end points is to be zero. Speed, acceleration, and jerk of the robot should be less than their safer limit. (4) Dynamic constraint—motor torque < its safe value.

The kinematics of the robot is given by $\dot{x} = v \cos \theta$, $\dot{y} = v \sin \theta$, $\dot{v} = \omega$, ω_R and ω_L are found by the fuzzy logic controller.

5 Experimental Verification

The experimental verification by IHDE algorithm is shown in Fig. 3. The following procedure is used in this research work for doing experiments.

- Step 1 All known values about the robot geometrical parameters, kinematic characteristics and dynamic performance values, actuator constraints, starting and goal positions of the path, current geometrical shape and positions of the obstacles, and obstacles velocity are given input to the motion planner.
- Step 2 Robot is deployed in the environment at the starting point. It is switched on.

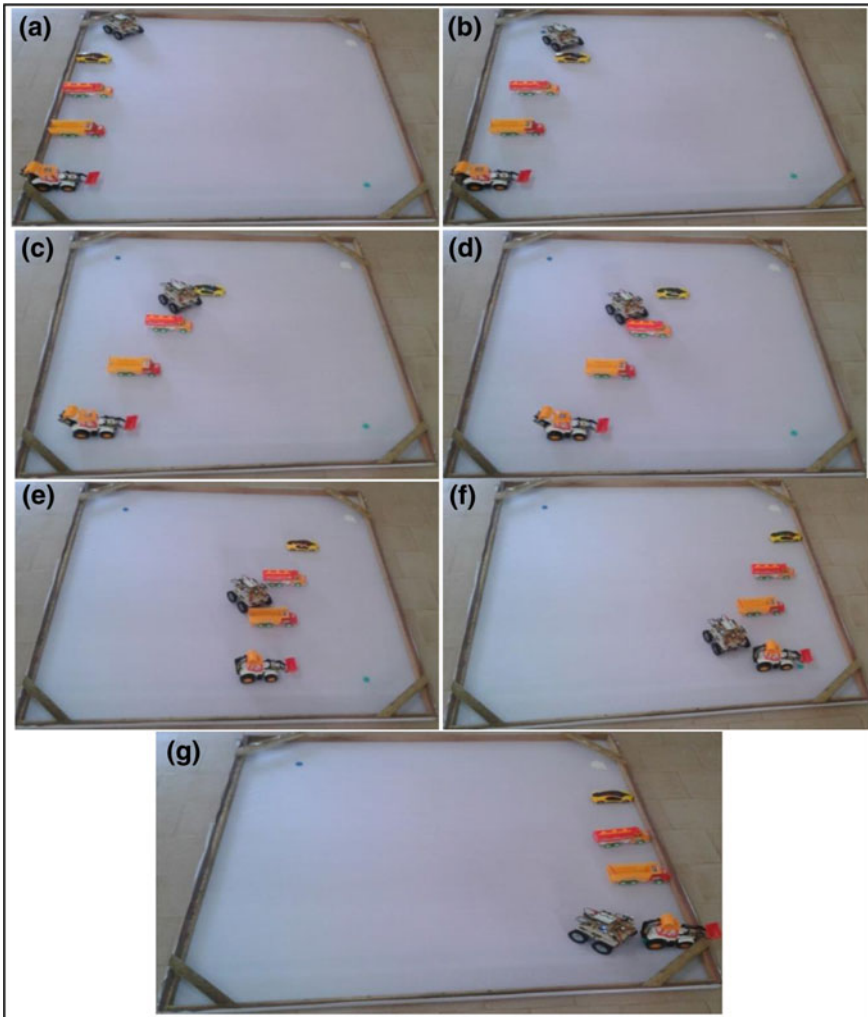


Fig. 3 a–g Mobile robot’s optimal motion

- Step 3 Robot sensors receive information about the environment. They observe the movement of the obstacles’ position and velocity. Sensors give the information to the motion planner.
- Step 4 Motion planner analyzes the information from sensors. It finds a best path for robot movement. Also, it selects a safe velocity to the robot. The information is communicated to the robot controller. Controller drives the robot. Feedback signals are continuously given to the controller. The controller interacts with the motion planner and does necessary corrections in the

Table 1 Various results got from IHDE and GA

	F_1 (m)	F_2 (s)	COF
<i>Simulation</i>			
GA	2.89	26	2.73
IHDE	2.68	21	2.39
<i>Experiment</i>			
IHDE	2.70	22	2.45
Error (%)	0.75	4.76	–

robot path and the robot velocity. Thus, it safely drives the robot toward the goal point.

Step 5 The robot safely moves to its goal position. Its travel time and electrical energy spent by the motors are noted.

6 Discussion About Results

The results from various techniques, namely IHDE and GA, are compared in Table 1. From Table 1, the following points are noted: (a) Path length (F_1) of the optimal path denoted by IHDE is less than that of GA. (b) Travel time (F_2) of the optimal path denoted by IHDE is less than that of GA. (c) Combined objective function (COF) of the optimal path denoted by IHDE is less than that of GA. (d) IHDE proved that it is better than GA. (e) By following the path dictated by IHDE, the robot safely moved from start point to end point. (f) The path dictated by IHDE is feasible and practically possible. (g) The path dictated by IHDE satisfied all geometrical, dynamics, and kinematics constraints. (h) The error between the simulation and the experiment is 0.75% for path length and 4.76% for travel time.

7 Conclusions

An experimental verification for an evolutionary algorithm-based online motion planning method for a car-like mobile robot which operates in a dynamic environment with velocity obstacles was performed in this work. From the experimental results, the points concluded are: (a) Path length (F_1) of the optimal path denoted by IHDE is less than that of GA. (b) Travel time (F_2) of the optimal path denoted by IHDE is less than that of GA. (c) Combined objective function (COF) of the optimal path denoted by IHDE is less than that of GA. (d) IHDE proved that it is better than GA. (e) By following the path dictated by IHDE, the robot safely moved from start point to end point. (f) The path dictated by IHDE is feasible and practically possible. (g) The path dictated by IHDE satisfied all geometrical, dynamics, and kinematics constraints. (h) The error between the simulation and the experiment is less. (i) The proposed IHDE can be used for all the robots' motion planning problem.

7.1 Limitations and Future Works

More number of real-world situations can be considered. The proposed IHDE can be used for all the robots' motion planning problem. So, experimental verification of other type of robots can be performed. IHDE can be improved for multi-objective problem-solving nature. More number of velocity obstacle and static obstacles can be considered.

References

1. Szádeczky-Kardoss EG, Gyenes Z (2018) Velocity obstacles for car-like mobile robots: determination of colliding velocity and curvature pairs. *Adv Sci Technol Eng Syst J (ASTESJ)* 3(1):225–233
2. Jiono M, Mahandi YD, Rahmawati Y, Putro SC, Ardiyansyah F, Prasetyo D (2019) Online motion planning for mobile robot. In: *Proceedings of international conference on electrical, electronics and information engineering (ICEEIE)*, Denpasar, Bali, Indonesia, Indonesia, pp 85–89
3. Kolor K, Chintalapudi S, Boots B, Mukadam M (2019) Online motion planning over multiple homotopy classes with Gaussian process inference. In: *Proceedings of IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Nevada, Las Vegas
4. Zhang X, Zhang B, Qi C, Li Z, Li H (2019) An online motion planning approach of mobile robots in distinctive homotopic classes by a sparse roadmap. In: Yu H, Liu J, Liu L, Ju Z, Liu Y, Zhou D (eds) *Intelligent robotics and applications. Lecture notes in artificial intelligence*, vol 11743. Springer-Verlag, Berlin Heidelberg, New York, pp 722–734
5. Ramabalan S, Sathiyar V, Chinnadurai M (2021) Wheeled mobile robot trajectory planning using evolutionary techniques. In: Kumar N et al (eds) *Advances in interdisciplinary engineering. Lecture notes in mechanical engineering*, vol 9956. Springer Nature Pte Ltd, Singapore, pp 291–301
6. Khaksar W, Hong TS, Sahari KSM, Khaksar M, Torresen J (2019) Sampling-based online motion planning for mobile robots: utilization of Tabu search and adaptive neuro-fuzzy inference system. *Neural Comput Appl* 31:1275–1289
7. Zhang H, Lin W, Chen A (2018) Path planning for the mobile robot: a review. *Symmetry* 10(450):1–17
8. Sharma P, Sharma H, Kumar S, Sharma K (2019) Black-hole gbest differential evolution algorithm for solving robot path planning problem. In: *Harmony search and nature inspired optimization algorithms*. Springer, Singapore, pp 1009–1022
9. Jain S, Kumar S, Sharma VK, Poonia RC (2020) Peregrine preying pattern based differential evolution for robot path planning. *J Interdisc Math* 23(2):555–562
10. Jain S, Sharma VK, Kumar S (2020) Robot path planning using differential evolution. In: *Advances in computing and intelligent systems*. Springer, Singapore, pp 531–537
11. Thangavelu S, Shanmuga Velayutham C (2015) An investigation on mixing heterogeneous differential evolution variants in a distributed Framework. *Int J Bio-inspired Comput* 7(5):307–320
12. Sathiyar V, Chinnadurai M (2019) Evolutionary algorithms-based multi-objective optimal mobile robot trajectory planning. *Robotica* 37(8):1363–1382

Towards Formalization of Constructivist Seed AI



Swarna Kamal Paul and Parama Bhaumik

Abstract A general intelligent system is expected to solve wide range of problems and adapt across multiple varied environments. It has already been hypothesized that seed AI needs to be bootstrapped in a system which could evolve to handle multiple problem domains. Such a seed AI may consist of few core intelligent components and a seed program. However, there is no formal structural definition of seed AI in place. Thus, an abstract model of the seed AI is presented and its generality has been proved. The formal structure of such an algorithm has also been derived. It has been discussed how and in what setting the proposed model of seed AI can achieve different properties of an intelligent agent like adaptability, constructivism, learnability and recursive self-improvement. The proposed theoretical framework for seed AI can serve as the basis for construction of any artificial general intelligent system. A prototype of seed AI has been developed using universal search to demonstrate and present a guidance on physical implementation of an agent-based system. The agent has been experimented in a heterogeneous toy problem to illustrate its usability.

Keywords Seed AI · Constructivist AI · Universal search · Category theory · Reinforcement learning

1 Introduction

There have been several major advancements in last few decades in domain specific artificial intelligence. Artificial intelligent systems have achieved state of the art performance in visual processing, audio processing, language processing, game playing etc. However, portability of an intelligent system from one problem environment to another is inefficient and requires manual reconstruction. Real world problems are barely restricted to a small domain. For example, driving a car requires complex interconnections among visual processing, auditory processing and motor

S. K. Paul (✉) · P. Bhaumik
Jadavpur University, Salt Lake, Kolkata, India
e-mail: swarna.kpaul@gmail.com

skills. These complex systems are mostly hand engineered by following a constructionist approach [1]. Constructionist method of designing an AI has its own limitations, like restricted learnability and portability, which calls for a new design approach, known as constructivist approach [2]. Constructivist systems are empowered with automatic construction of solutions in a given arbitrary problem environment with limited resources and little external bias. In order to optimize usage of available resources the system is expected to self-modify and improve its learnability and portability. To create such an artificial intelligent agent, it needs to be bootstrapped with a seed program [3] and an intelligent core which would eventually improve itself and adapt to variety of environments. Even for the constructivist approach a seed AI is necessary to let the agent know at least the method of construction. In this paper a principled approach of creating a seed program has been discussed and a formal grounding has been provided. Questions like, what is the minimum structure required to construct such a seed AI and how it achieves multiple properties of an intelligent system are answered. There are multiple architectures present in the literature [4–9] to achieve general intelligence, however any of them barely discusses the abstract minimal structural requirement to achieve different properties of general intelligence namely, constructivism, learnability, recursive self-improvement and adaptability across environments.

Rest of the paper is organized as following. Section 2 covers the related work followed by Sect. 3 on abstract modelling of the seed AI. The seed AI is formalized in Sect. 4 and its properties are discussed. Section 5 introduces an implementation of seed AI and an intelligent agent as a whole. Section 6 discusses experimental results on a toy problem environment followed by a conclusion.

2 Related Work

Thorisson proposed a theory of constructing seed-programmed intelligent system which rests on certain hypotheses of learning [9]. The theory contextualizes three high level aspects of problem solving, namely task-environments, cumulative learning and seed program. Nivel et al. presented an architecture blueprint for constructing a bounded seed AGI which can achieve operational autonomy in underspecified environment [4]. They developed a prototype system AERA which can learn complex real-world tasks. Steunebrink et al. introduced an experience-based AI system called as EXPAI, [10] which can search for safe and beneficial self-modifications. Several limitations of proof-based approach of self-modification have been highlighted and experience-based approach was chosen for self-modification. Hutter's AIXI framework [5] is a complete theoretical definition of an AI agent in the sense that it can optimally act in any environment given infinite computational resources. A modified framework is named as AIXItl whose computation time is bounded by $t.2^t$. Optimal Ordered Problem Solver (OOPS) [11] is a practical implementation of universal search [12] for solving an ordered set of problems and tries to reuse solutions found in earlier problems to solve later problems. Paul et al. also followed a novel

approach of using data flow graph-based programming model with universal search to solve multiple problems [13–15]. Gödel machine is a self-referential, self-improving optimal problem solver, originally proposed by Schmidhuber [6]. A probable implementation roadmap was given using continuous passing style (CPS) of programming and meta-circular evaluators [16]. Cognitive synergy [8] theory provides a theoretical framework to achieve AGI. A collection of cognitive processes acts in a synergistic way to control a single cognitive agent. The interconnectedness among the processes aids each other in overcoming memory-type specific combinatorial explosions during diversified knowledge creation. Non-Axiomatic Reasoning system [7] is a system designed to be adaptive and to work with insufficient knowledge and resources. It uses term-logic based knowledge representation, experience grounded semantics and non-axiomatic reasoning for inference. It can represent a notion of self and its related functions are developed through system experience.

3 Abstract Modelling of Seed AI

Category theoretical approach has been taken for abstract modelling of seed AI. Milewski [17] provides an excellent introduction to category theory. A useful seed AI is one which enables an agent to act optimally in a task environment and learns to adapt across varied task environments. It is assumed that all the task environments with which an agent interacts in its lifetime would be a computable one. Thus, every task environment can be represented as strings over some set of characters. Deductively, considering the agent as a part of the environment is also a computable one and all of its parts can be represented as strings. The environment can be considered as a single object monoidal category over a set of primitives in a Turing complete language and a composition operator. The object in the environment are programs. Programs can be created by composing primitives and other programs. Thus, the category is equipped with a bi-functor $\sigma_e: E \otimes E \rightarrow E$, an identity or empty string I_e , where E represents environment object. In a general sense there can be infinitely many possible tasks in any arbitrary environment, represented by different unique programs e or elements of object E . However, only those tasks are practically solvable by an agent which allows the agent to interact with it and halts on every input. On interaction with the agent, a task environment e can produce some output such that $e(x) \rightarrow y$. The output y can be interpreted by the agent to identify reward signals r . Considering the objective of the agent is to act optimally so as to maximize future expected total reward, let us assume a set of programs p exist in another Turing complete language which can act optimally in all such task environments e .

At least one optimal policy exists for all solvable task environments.

Proof Let us consider a cybernetic agent which runs a program p in a Turing complete language identified as policy for task environment e . The agent is coupled with the environment in a sense mentioned in [5]. The task environment's output

is consumed by the agent and vice-versa, in an iterative manner. The environment's output y is interpreted as a product of observation and reward received $o \times r$. Considering the agent's goal is to maximize future expected reward on some arbitrary horizon m an optimal policy is defined as,

$$p^* = \arg \max_p V_{1m}^{pe}, \text{ where } V_{km}^{pe} = \sum_{e'} \mu(e') \sum_{i=k}^m r_i^{pe'} \quad (1)$$

V_{km}^{pe} is the utility of the policy p on task environment e in cycles k to m . e is probabilistic mixture of task environments e' with probability distribution $\mu(e')$. If e is deterministic then $e' = e$ and $\mu(e') = 1$. As per initial assumption, both e and p are programs constructed in two Turing complete languages U_e and U_p , respectively. As they are Turing complete so they are Turing equivalent. Thus, any program e have a semantically equivalent program p . Thus, every solvable task environment e have an equivalent program p_e in U_p . V_{km}^{pe} can be calculated for any p and finite m if e is known, by running e in coupled fashion with p for m steps. Considering in each step, p is allowed to write a fixed length (l) string, a finite set of p can be tested by generating all possible combination of strings of a finite length ml and p^* can be selected generating maximum V_{1m}^{pe} . From the above argument it is clear that once task environment e is known there is a fixed algorithmic method to derive the optimal policy. Thus every p^* is a fixed functional extension of p_e . For every e in U_e there is a semantically equivalent p_e in U_p and consequently a p^* , although there might be many unsolvable e and p_e for which p^* does not make any sense.

Solution of any arbitrary solvable task environment can be computed using an intermediate Turing complete language, a set of two lax monoidal functors and a monoidal product operator.

Proof Let P denote a category of all such p_e which semantically maps with every e such that p^* can be constructed by functionally extending p_e . For a single e there can be multiple p_e which are semantically equivalent. To make a surjective functional map from E to P the p_e having minimum length is chosen for each e . All p can be represented by strings constructed by a finite set of alphabets A_p for U_p . Similar to E , P is also a monoidal category equipped with a bi-functor $\sigma_p: P \otimes P \rightarrow P$ and an identity object or empty string I_p .

Let us define a lax monoidal functor between categories E and P which signifies the map of every task from E to P . A lax monoidal functor comes with a functor which maps objects between categories, a morphism to map identity and a natural transformation to map functorial product which satisfies the usual associativity and unitality conditions.

$$F: E \rightarrow P \quad (2)$$

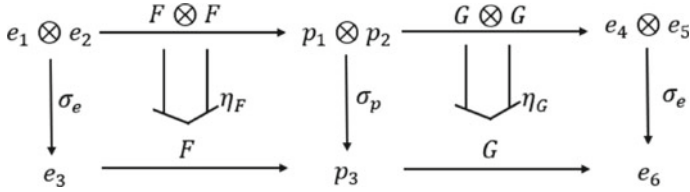


Fig. 1 Commutative diagram for problem environment and solution mapping

$$\in : I_P \rightarrow F(I_E) \quad (3)$$

$$\eta_F: F(e_1) \otimes_P F(e_2) \rightarrow F(e_1 \otimes_E e_2), \forall e_1, e_2 \in E \quad (4)$$

Functor F maps semantically equivalent programs from E to P . One way of doing it is taking any program in U_e and convert it into U_p by adding a required interpreter of U_e written in U_p . However monoidal product in σ_p keeps only the shortest among semantically equivalent programs. Thus F is a set of translation functions which translates any program in E to shortest equivalent program in P and preserves compositionality. As stated in Eq. 4 the usual commutative relation holds for F and monoidal products of E and P . Functor G derives the program corresponding to the optimal policy for task environment E using the simulated image P and maps that to semantically equivalent program in E . This can be done by adding specific code piece in P to derive the optimal policy using the model of the environment that can be interpreted by U_e . The code piece added by G should also contain logic to preserve product operation σ_e in E .

Figure 1 states that for any arbitrary solvable task in E there exist an optimal policy in E itself which can be derived by applying functor $G \circ F$. The policy can also be constructed from existing sub policies (e_4, e_5) which may be full or partial, using the same monoidal product operator σ_e in E . If a task e_1 transforms into a more complex task e_3 the commutative diagram states that solution policy e_6 can be derived by multiple pathways. One by applying $G \circ F$, which signifies deriving the solution from scratch. The other two solution pathways are by $G \circ \sigma_p \circ F \otimes F$ and $\sigma_e \circ G \otimes G \circ F \otimes F$. Both pathways reuse part of solutions found while solving subproblem e_1 . In many cases solving e_1 and reusing solutions for e_1 to solve e_3 might be less costly in terms of resources like time and space. Thus, any solution can be derived by using functor F , functor G , monoidal product σ_p and intermediate Turing complete language U_p . Hence proved.

4 Formalization of Seed AI

Seed AI can be defined as the minimum algorithm which is capable of evolving and finding optimal solutions for wide range of problem environments under resource

constraints like time, space and prior knowledge. Following the abstract model, any arbitrary task environment can be represented in a Turing complete language U_e . The solutions can be symbolically represented in another Turing complete language U_p which can be simulated in U_e . The seed AI can be represented as just another program in U_e . We choose the pathway $G \circ \sigma_p \circ F \otimes F$ in our design of seed AI to map solutions to environment. Consequently, individual morphisms need to be implemented in U_e to realize a concrete implementation of seed AI.

The functor F can be understood as an estimator of any task environment e in U_e , represented as program p in U_p . Ideally p should be exactly semantically equivalent with respect to e . But in real world scenario exact description of e may not be known and e can be estimated only by interacting with it. The quality of estimation depends on other environmental factors, like resource constraints. Among multiple descriptions of the environment the selection needs to be done based on some score which measures fitness or utility of the solution found using the description of the environment p in the context of a specific task environment e . The functor uses a helper function v in U_e to measure the utility in a specific context. Implementation of F in U_e is defined as follows

$$f: e \times v \rightarrow p \quad (5)$$

σ_p denotes a function constructor using composition operator whose domain consists of all semantically unique and shortest programs in U_p . Considering U_p a functional programming language with a set of primitive functions or a generator set ω , making P a free monoid, all program equivalence can be calculated by algebraically simplifying program expressions due to absence of side effects. Implementation of σ_p in U_e is defined as,

$$\sigma_p: \begin{cases} p_1 \times p_2 \rightarrow p_2 \circ p_1, & \text{if } U_p(p_2 \circ p_1) \cdot U_p(p_1) \cdot U_p(p_2 \circ p_1) \cdot U_p(p_2) \\ p_1 \times p_2 \rightarrow p_2, & \text{if } U_p(p_2) \equiv U_p(p_2 \circ p_1) \\ p_1 \times p_2 \rightarrow p_1, & \text{if } U_p(p_1) \equiv U_p(p_2 \circ p_1) \end{cases} \quad (6)$$

The natural transformation η_F transforms the estimator function f of a solved task to find a new estimator for a new task environment by reusing the estimator of the solved task. The natural transformation in U_e is derived using the estimator function f and the function constructor σ_p . Implementation of η_F in U_e is defined as,

$$\eta_f \equiv \sigma_p \circ (f \times f) \quad (7)$$

The functor G computes the optimal solution program p_{sol} or the policy, using the environment estimation p and transforms it to an executable e_{sol} in U_e . This can be done by adding the interpreter of U_p written in U_e so that p_{sol} can be interpreted in U_e . G uses the helper function v to find p_{sol} and consequently e_{sol} . Implementation of G in U_e is given as following.

$$g: p \times v \rightarrow e_{sol} \quad (8)$$

4.1 Seed AI Algorithm

For any solvable task environment, the seed AI searches for a solution represented as a program in a Turing complete language. For any given task environment e , the seed AI generates multiple programs as estimated representation of task environment and allocates execution time proportional to the fitness of solution programs e_{sol} . This continues until a maximum age is reached. The fitness of a solution program is calculated by a helper function v , which might use the interaction history ($trace(e_{sol})$) of the program with the task environment to calculate the same. The fitness of solution program also determines the fitness of the estimated environments (p_1, p_2). f selects two of the best estimations based on fitness and σ_p combines them to produce a new estimation. In order to bootstrap, f is supplied with a generator set which represents the primitives of U_p . g evaluates the new estimation by deriving the solution program and running it for a fraction of the total allocated runtime (T_R) in a phase. The runtime allocation is based on expected fitness of the solution. If the program does not halt within the allocated runtime, it is interrupted. Similar to the Levin search [9] the dynamic runtime allocation to individual programs alleviates the halting problem and makes the algorithm computable. T_p is the total runtime of the program e_{sol} and \mathbb{E} represents the evaluator corresponding to U_e . After completion of each phase T the total runtime allocation is doubled and the same process repeats.

The Seed AI Algorithm

Function seedAI($e, v, f, \sigma_p, g, \text{maxage}$)

While $T < \text{maxage}$

$T_R = T$

While $T_R \leq 0$

$f: e \times v \times T_R \rightarrow p_1$

$f: e \times v \times T_R \rightarrow p_2$

$\sigma_p: p_1 \times p_2 \rightarrow p_3$

$g: p_3 \times v \times T_R \rightarrow \mathbb{E}(e_{sol}, T_R) \times trace(e_{sol}) \times T_p$

$T_R = T_R - T_p$

$T = T * 2$

4.2 Constructivism

Constructivism is at the heart of the seed AI. Thorisson [2] already justified the need of constructivism in building Artificial general intelligent systems. The seed AI achieves

autonomy and adaptability by constructing estimations of task environments based on action perception history, recorded as experience. One usual problem of constructivist approach is facing exponential search space in order to find the right architecture or solution for a problem environment. The seed AI may also face this problem at its birth with minimal experiential knowledge as it needs to construct all possible programs and test them in some order. In course of doing so, gaining experiential knowledge will help dampening the search space and with the proper choice of ν the agent will gradually converge toward an optimal or near-optimal solution. Capability of constructing solutions for any solvable task environment imparts a good generalization property to the seed AI. Yet, practically usable solutions may not be constructed from scratch in resource constrained environments. For example, a specific task environment may demand recognizing objects from captured images and pick up the objects which are recognized as apples using a robotic arm. Solving such an environment requires solving the subtasks of object recognition and robotic arm control. Creating solutions for these subtasks from scratch in time constrained environment may not be practically feasible. Deep neural networks are already proved excellent in handling vision processing tasks. Pretrained deep neural nets acting as object recognizers can be placed in the environment as a module. A robotic arm controller can also be placed as a library in the environment. The seed AI then constructs a program to integrate and control these modules in order to achieve the final goal. The addition of modules can be anytime without breaking the operation state of the seed AI, thus allowing integration of human intelligence with artificial intelligence. Paul et al. [18] already presented an integrative AI platform in line with this idea. However, the integration of modules needs to be done manually in the proposed platform. A seed AI may run another seed AI program which may construct a solution for a subtask and place it in the environment as a module. Other constructed solution programs may reuse that module and achieve a level of abstraction.

4.3 Learning

Learning is the process of gaining experience with constrained resources based on interaction history with an environment so that the environment can be modified in the desired way as much as possible. Learning is analogous to the prediction problem with constrained resources. It can be defined as finding the best possible machine model M for an environment with limited interaction history of action/perception, time and space. The learnt machine model M can be used to predict optimal sequence of actions for desired perceptions. Given a set of interaction histories an agent can find the best possible estimation of an environment using brute force. But it might use unlimited amount of time to find the model. Similarly, an expert system may tag outputs against inputs as rules, but it might need immensely large set of interaction history for any arbitrary environment to realistically model it, let alone assuming there is no effect of noise in measuring perceptions. In both the cases it can be comfortably said that there is least amount of learning involved. Learning is essentially an optimization

problem of finding patterns in interaction history with the environment, using limited resources and thereby model it to get maximum desired perceptions.

The seed AI have inherent learning capability as it tries to find an estimate of the task environment (p_3) so as to find out the program (e_{sol}) that generates desired set of actions. The usage of limited time to learn the model is already embedded in the logic, as time allocated to evaluate a model of the environment is directly proportional to the expected fitness of the solution. The choice of ν is important to drive the learning process. Proper choice of ν would allow progressive learning such that it balances exploration and exploitation and the learnt model improves with time. ν can also control the usage of limited space in the learning process. For example, ν considers models with smaller size more fit comparatively. The design of seed AI inherently adds metalearning capability. For example, suppose a task environment requires to find neural architectures which provides high accuracy with low training effort and using minimal training examples. Then in such scenario ν helps the seed AI to learn to find out neural architectures with improved learning capabilities. Also, suppose the objective of a task environment is to find out seed AI with better learning capabilities, it may generate a ν which may improve the child seed AI when evaluated by the parent one. This adds capability of metalearning of metalearning and so on.

4.4 Adaptability Across Environments

Let us assume, the seed AI acts optimally in task environment e_1 which transitions to environment e_3 by combining with e_2 in a certain way. By Eq. 1, there is a p_1 corresponding to e_1 in the representative language U_p . If seed AI already learnt p_1 , it needs to search for p_2 corresponding to e_2 which can be combined to form p_3 and eventually e_{sol3} which represents solution of e_3 . Thus, the seed AI effectively reuses solutions of solved task environments to adapt to new task environments if the search process is less costly for a given ν . Otherwise, it may directly search p_3 without reusing p_1 . In either case, the seed AI is capable of transitioning to the new solution program when the task environment changes, based on utility derived from action perception trace. Making an instance of seed AI callable from programs constructed by the parent seed AI makes the search hierarchical. A task may be automatically subdivided into subtasks and solved by child seed AIs and reused in the solution found by the parent seed AI.

A practical embodied intelligent agent is born in a single continuous environment. The embodiment itself serves as the internal environment for the seed AI of which seed AI is a part. It might include several sensors, actuators, knowledge bases, other soft skills etc. Sensors, actuators may be physical or virtual, but they are meant to communicate with the environment outside embodiment. However, there should be a task in its internal environment which should give the agent a drive for its intelligent behavior. It is solely an internal task within the embodiment. The root

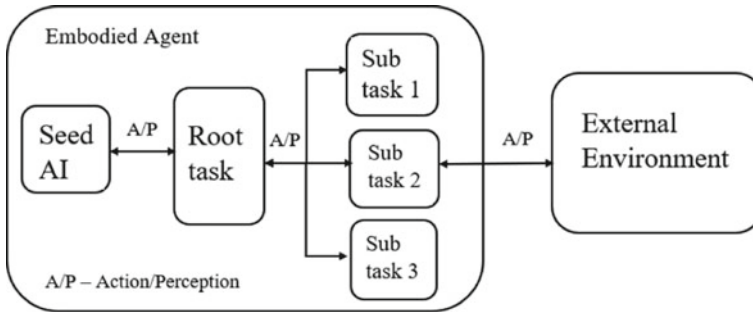


Fig. 2 Illustration of an embodied general intelligent agent

level seed AI communicates with this task environment and tries to search for the optimal solution. Now the question is what should be this task? This task can be anything for any arbitrary embodied agent, depending on the objective the agent is created to serve. For a general intelligent agent, the task can simply be “act optimally using all functional components present in the internal environment” or “strive to maintain a homeostatic condition of internal environment”. This gives an internal drive for the agent. The internal environment can be influenced and changed by external perception signals which indirectly effects the drive of the seed AI. But in a general intelligent embodied agent the seed AI never communicates with the external environment which makes it impossible to control directly by an external being.

Though creating an internal task environment for a general intelligent agent may sound simple yet it is an engineering challenge. The root task may consist of multiple tasks and components. Actions and perceptions need to be properly routed between subtasks and seed AI. Some perceptions may get priority over others or multiple perceptions may get merged and transformed. For example, a physical damage in any part of the agent may generate a strong punishment signal irrespective of the task in which the seed AI is currently engaged.

Figure 2 illustrates an embodied agent constructed using seed AI. The root task represents the primary task in the internal environment of the agent which may consist of multiple subtasks. Some subtasks may in turn communicate with external environment. The seed AI communicates with the root task and searches for solution.

4.5 Recursive Improvement

Self-improvement by self-modification of an intelligent agent is necessary to adapt across environments and continue acting optimally [19]. Arguments provided by Hall establishes the usefulness of self-improvement in achieving universal intelligence and renders the bootstrap fallacy as generalization of experience with brains and

systems below the level of universality [20]. The seed AI is just a program synthesis and scheduling algorithm which makes it a synthesizer, selector and evaluator of specific programs. Due to this property, self-modification is inherent to seed AI. If a specific program solves a specific task it is going to get maximum time allocation. On changing the task environment, the seed AI will eventually shift time allocation to some other program which serves as the solution for the new task.

Recursive self-improvement requires improving the self-improvement algorithm itself, aka the seed AI. Recursive self-improvements can be found by creating a task environment using the seed AI itself. Let us consider a task environment e_ξ , where ξ denotes an instance of seed AI. e_ξ can run another instance of seed AI ξ' with any ν, f, g and σ_p when interacting with ξ , considering ξ can generate/modify arbitrary ν, f, g and σ_p in U_e which serves as parameters for ξ' . e_ξ generates a positive reward if the fitness or utility of instance ξ' exceeds all previous fitness when measured with ν with which ξ is running. Considering ξ is also running within e_ξ , e_ξ generates reward if $\nu(\xi') > \nu(\xi)$. $\nu(\xi)$ measures the utility of ξ with respect to the root task. ξ' can be represented by a program generated by ξ while interacting with subtask e_ξ . Increment of the utility of ξ in general indicates improvement of ξ itself. ν can be a function whose one component can represent future expected reward which makes one of the goals of ξ is to maximize total future expected reward with respect to root task. If e_ξ is a part of root task then finding self-improvement of itself is a part of the goal of ξ . In such scenario, if ξ' performs better than ξ with respect to root task in general then $\nu(\xi') > \nu(\xi)$. Getting rewards for ξ' from e_ξ positively reinforces the value of $\nu(\xi')$ and its fitness gradually increases. As fitness of ξ' increases ξ allocates more time to ξ' and it may eventually converge to a limit. Thus, it essentially means ξ' evolved as the improved version of ξ and occupies major share of the execution time. This can happen recursively until no further improvements are found with respect to root task. Making the agent a part of the environment itself imparts the ability to self-modify [21]. Thus, the trick of achieving recursive improvement is making the solution part of the problem environment such that improvement of the solution can be automatically found by solving a task environment which essentially asks for improvement of the prior solution.

4.6 Agent as Environment

After introducing a structure of seed AI, naturally a question arises—What constitutes an intelligent agent? Is it the seed AI? No! The agent is the environment or rather the internal environment of an embodied agent of which the seed AI acts as a controller or integrator. An embodied agent may contain several sensors, actuators, internal knowledge base, reasoning capability etc. All having some functional interfaces along with some subtasks to train the agent to use these capabilities. The agent has a root task which gives the internal drive to do everything what an agent does.

Acting optimally against the root task may reuse solution programs for individual subtasks and integrate them with other programs. Interaction with external environment is optional, yet to make an agent useful in real world it is necessary. Communication with external environment is carried out with functionalities available in the internal environment of the agent. The embodied agent may be supplied with various inbuilt capabilities to act optimally against external environment and the list may be augmented from time to time. The seed AI achieves autonomy by learning to integrate and control the functional capabilities available within the embodied agent so as to make it useful in the external environment.

5 Seed AI Implementation with Universal Search

This section briefly describes a probable implementation roadmap of the seed AI using universal search. However, there can exist multiple different other approaches for constructing a seed AI. For example, a variant of evolutionary algorithm might also act as a seed AI as there are many structural similarities between the two. Universal search [12] is an asymptotically optimal method to solve machine inversion problem or time limited optimization problem. A wide variety of computational problems can be either transferred to machine inversion problem or time limited optimization problem [22]. The search process generates and tests arbitrary programs in the order of levin complexity until a solution is found for machine inversion problem or some time limit reaches for time limited optimization problem. The following expression states the Levin's complexity, where $\text{time}(p)$ is runtime of p in M .

$$\begin{aligned} Kt_M(y) &= \min\{l(p) + \log(\text{time}(p)): l(p) \\ &= \text{length of } p \text{ and } M(p) = y\} \end{aligned} \quad (9)$$

It can also be re-written as following where $P = 2^{-l(p)}$ is probability of program p , considering p can be represented by bit strings.

$$Kt_M(y) = \min \left\{ \log \left(\frac{\text{time}(p)}{P} \right) \right\} \quad (10)$$

Structurally seed AI and Universal search is quite similar. Universal search consists of a generator, selector and an evaluator function. It generates a program by combining two programs. It repeatedly selects a program based on program probability and tests with certain time allocation proportional to program probability.

5.1 *Programming Model*

Universal search constructs programs using a programming model. A functional dataflow graph-based programming model has been used for this purpose. The functions are represented by nodes, dataflow by edges and programs by directed acyclic graphs. Programs are constructed by composing functions. The details of the programming model can be found here [15]. The programming model is equipped with a set of primitives, aka the *generator set*. Programs are constructed by arbitrarily composing functions from this generator set.

5.2 *Metasearcher*

The metasearcher is the physical implementation of seed AI using universal search. The metasearcher implements the synthesizer, the fitness function and the evaluator. The evaluator serves as the interpreter of the programming model. The following functions demonstrates the implementation of each of the sub methods of the seed AI. The metasearcher generates a search graph to solve a problem environment. The search graph consists of a cluster of generated programs in the programming model. Individual programs are executed, perceptions from environment are recorded and reward signals are extracted for each program. The sequence of rewards constitutes the output trace of each program which in turn is used to update the fitness. The search graph loosely resembles a model of the problem environment which can be used to estimate expected reward on executing a sequence of actions as defined by a specific program graph.

The fitness of a program is calculated by the program probability which in turn indicates the fitness of a search graph. The maximum fitness among all programs in a search graph denotes the fitness of the search graph itself. The program probabilities are assigned and updated in the same strategy as mentioned by Paul et al. in [15]. Program probability distribution is updated incrementally based on the sequence of rewards received by individual programs. Probability distribution is updated using gradient ascent so as to maximize the total future expected reward, as defined by the objective function $J(p)$. x_k is a program in search graph with index k , i' represents the horizon and $p(r_i|x_k)$ represents conditional probability for gaining reward r_i when program x_k is extended to x_i .

The Metasearcher Algorithm

Function $fitness(X, x)$

If $x \notin X$

Assign default probability to x and
add in p_x^*

End If

$$J(p) = \sum_{\forall x_k \in X} \sum_{\forall i > k \wedge \forall i < i} p(r_i | x_k) r_i$$

$p_x^* \leftarrow \arg \max_p J(p)$

$p_x^* \leftarrow$ probability of x from p_x^*

Return p_x^*

Function $evaluator(X, T)$

$x = \arg \max_x fitness(X, x)$

$|\forall_{x \in X}$ and x is not executed $fitness(X, x) * T > 1$

$T_r \leftarrow eval(x, fitness(X, x) * T/c)$

If x halts in $fitness(X, x) * T/c$

Mark x as executed

End If

Return T_r

Function $synthesizer(X, T, y)$

$x = \arg \max_x fitness(X, \sigma(x, y))$

$|\forall_{x \in X}$ and x is executed $fitness(X, \sigma(x, y)) * T > 1 \wedge \sigma(x, y) \notin X$

If x exists

Add $\sigma(x, y)$ in X

End If

Return

Function $metasearcher(T_{limit}, Y)$

$T = 2$

While $T < T_{limit}$

$T_R = T$

While $T_R > 0$

For each y in Y

$synthesizer(X, T, y)$

$T_r = evaluator(X, T)$

$T_R = T_R - T_r$

End For

End While

$T = T * 2$

End While

The synthesizer selects program graphs (x) from search graph based on fitness and extends it by adding another selected function node (y). The $\sigma(\cdot, \cdot)$ is the composition operator which composes two program graphs and produces a third program graph. If the new synthesized program is not semantically redundant it is added in the search graph and resultantly modifies it. Semantically redundant programs with lower fitness are not added to the search graph to avoid the problem of over representation [15]. The synthesizer implements the natural transformation η_F in the seed AI. The evaluator selects the optimal program graph from the search graph, based on fitness and evaluates it using the interpreter of the programming model. The evaluator is allocated a runtime proportional to the fitness of the program graph, where c is a predefined constant value and $0 < c \leq 1$. If the runtime of the program exceeds the allocated time, it is interrupted else the program is marked as executed and runtime (T_r) is recorded.

The metasearcher implements the main body of the seed AI using the helper functions $fitness$, $synthesizer$ and $evaluator$. The metasearcher is executed in phases and in each phase a specific total runtime is allocated (T_R). Once the total allocated time is consumed it is doubled in each subsequent phase and this continues until the maximum age (T_{limit}) is reached.

6 Case Study

Experiments are conducted with the prototype in a heterogenous maze environment. Figure 3 illustrates the maze environment. The prototype acts as an agent in the maze environment whose objective is to traverse through the cells in the maze and reach the goal state. The rewards and images of direction in the vision zone helps guiding the search. This is a resettable environment and the agent is initially placed in [0,0] cell. Five actions are allowed in the environment, each shifting the agent in a specific direction. Namely, 1—front, 2—left, 3—right, 4—back, 0—return current cells observation. The vision zone returns base64 encoded image of the direction of the goal state. The agent searches through the program space and finds a program, executing which would generate sequence of actions to lead the agent to the goal state.

The task environment and components external to the programming model are implemented as microservices, as stated in [18]. All the microservices are configured to take an action request as input and returns observation, reward pair as output. The agent interacts with these microservices through an interface which serves as root task environment for the agent. The following functions are used as generator set for the metasearcher. *iW, 1.K, 2.K, 3.K, 4.K, maze, ocreco, lp*. The maze and ocreco are composite node functions. Each contains a similar program graph as illustrated in Fig. 4a. For ocreco the node maze.K is replaced with ocreco.K. The composite nodes act as functions to interact with the maze task and OCR component. The maze node sends the input argument as an action to the maze task and returns the observation. The cumulative reward obtained by running a program is used by the fitness function to update the probability distribution. The ocreco node takes an image encoded in base64 and returns the identified character in it. For any other input it returns 0. Table 1 enumerates the description of primitive function nodes used in solving the maze problem. Figure 4b. illustrates the program graph found by the metasearcher.

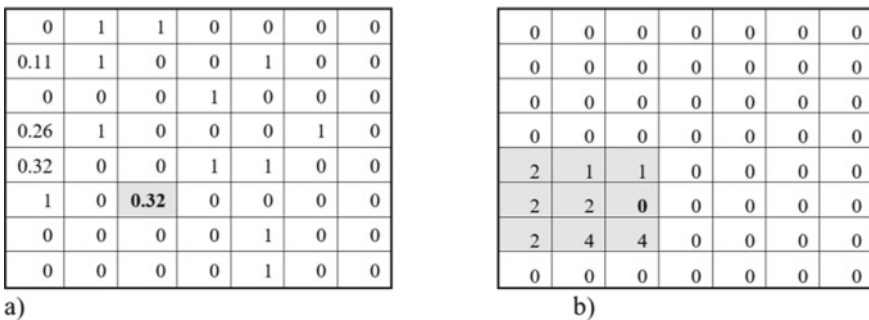


Fig. 3 Maze environment **a** reward distribution. The grey cell is the goal state. Cells containing 1 generate - 0.5 punishment. All other cells generate reward as specified in the illustration. **b** The grey region denotes the vision zone. Each cell in the grey region returns a base64 encoded image of the number specified as observation. All other cells return 0 as observation

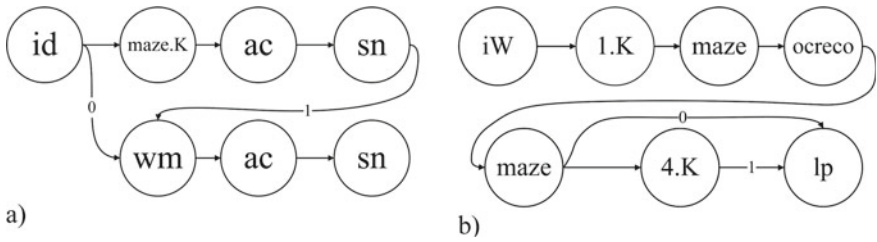


Fig. 4 Solution program graphs **a** program graph for the maze composite node. **b** Solution program graph found by the metasearcher for the maze problem environment

Table 1 Primitive function node descriptions

Node name	Description
id (identity)	Takes any input and returns the same
a.K (constant)	Takes any input and returns a constant value a
ac (actuator)	Takes any input and execute it as an action in the environment. Returns the modified environment
sn (sensor)	Takes environment as an input and returns the observation sensed from the environment
wm (worldmerger)	Returns the 1st argument as-is embellished with the environment object received as the 2nd argument
iW (initial world)	Initializes the environment and returns 0 embellished with the environment
lp (loop)	Executes the parent program graph for n times, where n is received as 2nd argument

There is a pattern in the solution which has been identified. The movement pattern is repeated 4 times in a loop to reach the goal state. The solution program took direct movement actions in the maze environment as well as used the OCR component where necessary, to solve this heterogenous problem environment optimally.

7 Conclusion

The presented formal architecture of seed AI provides a guidance to implement any general intelligent system. It has been proved that the presented model of the seed AI is theoretically capable of finding a solution for any solvable task. The seed AI algorithm states that a properly designed synthesizer of arbitrary computation logic and a scheduler is sufficient to achieve general AI. This alleviates the problem of finding an AGI system design from scratch through different approaches, as any system could evolve into AGI system if they are structurally similar to the seed AI. The seed AI guarantees to achieve generality by adapting across different solvable

task environments. It has also been discussed how the seed AI could achieve different properties of general intelligent systems, like learning, self-improvement, constructivism etc. An implementation roadmap of the seed AI is also demonstrated and the developed prototype is experimented in a toy problem. However physical implementation of a full-fledged general intelligent system is still an engineering challenge, specially designing an appropriate fitness function, the core internal components, the subtask environments and the root task. Multiple subtasks need to be designed based on situatedness of the system which would allow communicating with the external environment. On principle the seed AI should be able to generate subtasks. But its engineering setting, feasibility and applicability needs to be investigated. The limitations of the seed AI in an embodied agent also needs to be investigated.

References

1. Thórisson KR, Benko H, Arnold A, Abramov D, Maskey S, Vaseekaran A (2004) Constructionist design methodology for interactive intelligences. *AI Mag* 25(4):77–90
2. Thórisson KR (2012) A new constructivist AI: from manual methods to self constructive systems. In: *Theoretical foundations of artificial general intelligence*. Atlantis Press, Paris, pp 145–171
3. Yudkowsky E (2007) Levels of organization in general intelligence. In *Artificial general intelligence*. Springer, Berlin, Heidelberg, pp 389–501
4. Nivel E, Thórisson KR, Steunebrink BR, Dindo H, Pezzulo G, Rodríguez M, Hernández C, Ognibene D, Schmidhuber J, Sanz R, Helgason HP, Chella A (2014) Bounded seed-AGI. In: *International conference on artificial general intelligence*. Springer, Cham, pp 85–96
5. Hutter M (2003) A gentle introduction to the universal algorithmic agent AIXI. Technical report, IDSIA, Manno-Lugano, Switzerland
6. Schmidhuber J (2007) Gödel machines: fully self-referential optimal universal self-improvers. In: *Artificial general intelligence*. Springer, Berlin, Heidelberg, pp 199–226
7. Wang P, Li X, Hammer P (2018) Self in NARS, an AGI system. *Front Robot AI* 5:20
8. Goertzel B (2017) A formal model of cognitive synergy. In: *International conference on artificial general intelligence*. Springer, Cham, pp 13–22
9. Thórisson KR (2020) Seed-programmed autonomous general learning. In: *International workshop on self-supervised learning*. PMLR, pp 32–61
10. Steunebrink BR, Thórisson KR, Schmidhuber J (2016) Growing recursive self-improvers. In: *International conference on artificial general intelligence*. Springer, Cham, pp 129–139
11. Schmidhuber J (2004) Optimal ordered problem solver. *Mach Learn* 54(3):211–254
12. Li M, Vitányi P (2013) An introduction to Kolmogorov complexity and its applications. Springer Science & Business Media
13. Paul SK, Gupta P, Bhaumik P (2018) Learning to Solve single variable linear equations by universal search with probabilistic program graphs. In: *ICIBCAs*. Springer, Cham, pp 310–320
14. Paul SK, Bhaumik P (2020) A reinforcement learning agent based on genetic programming and universal search. In: *2020 4th ICICCS*. IEEE, pp 122–128
15. Paul SK, Bhaumik P (2021) Solving partially observable environments with universal search using dataflow graph-based programming model. *IETE J Res*. <https://doi.org/10.1080/03772063.2021.2004461>
16. Steunebrink BR, Schmidhuber J (2012) Towards an actual Gödel machine implementation: a lesson in self-reflective systems. In: *Theoretical foundations of artificial general intelligence*. Atlantis Press, Paris, pp 173–195

17. Milewski B (2018) Category theory for programmers. <https://cdn.jsdelivr.net/gh/it-ebooks-0/it-ebooks-2018-11to12/Category%20Theory%20for%20Programmers.pdf>
18. Paul SK, Jana S, Bhaumik P (2021) On solving heterogeneous tasks with microservices. *J Inst Eng (India) Ser B*:1–9
19. Nivel E, Thórisson KR, Steunebrink BR, Dindo H, Pezzulo G, Rodriguez M, Hernández C, Ognibene D, Schmidhuber J, Sanz R, Helgason HP (2013) Bounded recursive self-improvement. [arXiv:1312.6764](https://arxiv.org/abs/1312.6764)
20. Hall JS (2007) Self-improving AI: an analysis. *Mind Mach* 17(3):249–259
21. Everitt T, Filan D, Daswani M, Hutter M (2016) Self-modification of policy and utility function in rational agents. In: *International conference on artificial general intelligence*. Springer, Cham, pp 1–11
22. Solomonoff RJ (1989) A system for incremental learning based on algorithmic probability. In: *Sixth ICAI, computer vision and pattern recognition*, pp 515–527

The Effective Learning Approach to ICT-TPACK and Prediction of the Academic Performance of Students Based on Machine Learning Techniques



T. Saravanan, N. Nagadeepa, and B. Mukunthan

Abstract Objectives: To analyze and compare different techniques of machine learning for the student's academic performance to find an effective learning environment (ICT-TPACK). **Methods:** A descriptive study design was adopted among 3000 BCA and B.Sc. (computer science) students from various locations at Bharathidasan University affiliated colleges in the Karur region, Tamilnadu, India. The data was collected through internal marks from June 2019 to April 2021. The internal mark or grades are calculated for the time of submission of the assignment, attendance, evolution of internal examination answer script, MCQ test, and student's study seminar. **Findings:** This research paper was analyzed for ANN, LSTM, MLP, and our proposed methods (IP-LSTM). The proposed algorithms (IP-LSTM) can better reflect the academic performance of the students; With the LSTM method, some information can be forgotten. Our proposed method achieves 91% accuracy and 94% recall, which outperforms the other methods. **Novelty:** Introducing a new algorithm in ML-called IP-LSTM for knowing the effective learning techniques to improve students' academic performance to create an effective environment for learning.

Keywords ICT-TPACK · Machine learning · Algorithm · MLP · Student's academic performance

1 Introduction

The many researchers go to great lengths to improve student performance for various purposes, such as identifying students at risk, ensuring student loyalty, assigning courses and resources, and much more [1]. The goal of this paper is to forecast

T. Saravanan (✉) · B. Mukunthan
Department of Computer Science, Jairams Arts and Science College, Karur 639003, India
e-mail: saran.gpz@rediffmail.com

T. Saravanan · N. Nagadeepa · B. Mukunthan
Bharathidasan University, Tiruchirappalli 620024, India

N. Nagadeepa
Sri Sarada Niktan College of Science for Women, Karur 639003, India

student success based on a variety of academic and demographic factors. Machine learning algorithms may be used to forecast a student's performance and identify pupils who are in danger, allowing appropriate action to be taken to enhance that student's performance [2]. As a result, multiple machine learning approaches are employed to develop the model and forecast student performance in this study work.

This research has a minor student record to investigate the outcome. This research used a survey methodology to collect the data set and build the model. The student data set is collected by the Faculty of Computer Science, Department of Computer Applications. This data set contains information on 3000 students. This research follows various activities to create, normalize, and examine the results from the data sets such as data acquisition, preprocessing, identification of evaluation patterns, and output generation from five machine learning models including proposed IP-LSTM methodologies. The main objective of the research is to find the best model and analyze the results [3].

The LSTM architecture is a kind of sophisticated recurrent neural network (RNN) It makes use of an input gate, a forget gate, as well as an output gate as part of its triggering mechanism These gates aid in determining if whether data within the present state should be regarded or if it should be retained in the preceding state. As a result, the LSTM mechanism contributes to the solution of the long-term storing information problem and extinction gradient problems that traditional RNNs face. The powerful capabilities of LSTMs to extract information from data are an important part of data classification. The scope of LSTM [4] has significantly increased in recent years, and numerous academics have proposed a variety of approaches for updating LSTM to enhance its accuracy.

2 Related Works

Using socio-demographic and academic characteristics, [5] developed built a model to predict the academic achievement of incoming pupils with the use of data mining methods, the naive Bayes classifier, as well as Rapid miner tool, which the author created a model that was over 60% accurate. His model was used to predict how well the next generation of kids will do academically. Nominal as well as numerical data types dictated which classification algorithms were studied. These included k-NN, IBk, decision trees, as well as naive Bayes. The latter, according to previous research, may provide the greatest results. Singh et al. [6] evaluated how well previous academic achievement, entrance examinations, as well as interviews may help predict progress in a medical school. 706 university students from three cohorts were evaluated on their overall performance of Queensland's four-year graduate-entry medical program was studied. The GPA, graduate Australian medical school admissions exam scores, as well as interviews score, were all taken into account in the selection process. All assessments included questions about students' academic progress, in first and fourth-year assessments, as well as in individual written, ethical, and clinical components. Fei and Yeung [7] presented the process of creating multiple

intelligences-based student profiling as well as learning and effective methods to find the most relevant traits for engineering students' academic achievement. The foundation for two surveys which were completed with 618 undergraduate engineering students to determine their student profiles was established and was separated into eight dimensions for each of the two constructions. Instructors may utilize the definition of relevant metrics to assess their pupils' cognitive abilities, emotional, as well as self-regulation profiles in terms of the implementation of successful teaching practices in their classrooms.

Qu et al. [8] used a decision tree method using parameters such as information on academic performance as well as extracurricular activities of students to investigate a student's academic performance. We obtained data on 22 students enrolled in a private higher education institution in Oman who were enrolled in the spring 2017 semester and were studying at the undergraduate level. The suggested work will help students and improve their results in the module. Assisting stakeholders, with the analysis and evaluation of module delivery and outcomes. Both at the institutional and module levels, early identification and solutions are possible. This research examined data from students' activity as well as performance in a level 3 module of a Middle East College computing degree courses. This study utilized data from the spring of 2017 and a sample of 22 undergraduate students enrolled in the program in question.

Rodrigues et al. [9] addressed the online tutorial gap by analyzing a wide sample of tutorials against numerous characteristics of pedagogical efficacy developed from the learning sciences and education literature. Examined what and how learners were taught in 30 popular and different online coding courses. Tutorials typically taught identical information, were structured bottom-up, and gave goal-directed activities with instant feedback, according to the findings. Brinton et al. [10] introduced a novel prediction method for assessing student performance in academia. It was developed utilizing both classification and clustering methods and evaluated in real time on a student data set from numerous academic subjects at Kerala's higher educational institutes in India. The findings show that its hybrid procedure, which combines clustering and classification techniques, produces considerably greater results in terms of accuracy in predicting student academic achievement. Cui et al. [11], the author introduced a novel prediction method for assessing student performance in academia, which was built using both classification and clustering approaches and tested in real time using a student data set from several academic disciplines at higher educational institutions in Kerala, India. The findings show that the hybrid method, which combines clustering as well as classification techniques, produces considerably greater results in terms of accuracy in predicting student academic achievement.

3 Methodology

3.1 *ICT-TPACK Pedagogical Methods*

The TPACK framework identifies the different categories of knowledge that instructors need to properly incorporate technology into their courses. It suggests that instructors should understand the links between technology, pedagogy, and material. Specifically, how various domains of knowledge interact and impact one another in distinct and specific contexts [12]. In terms of technology-assisted teaching, it appears that it has an impact on not only what we teach but also how we teach. In the early 2000s, this concept was popular, and scholars were working on variations of it [13, 14].

In education which entails the involvement and adoption of general components of information and communication technology. These concepts and application of ICT in learning and teaching demand a new learning environment to effectively harness the power of ICT to improve learning. ICT has the potential to transform the nature of education like where, when, how and the way learning takes place. It will facilitate the emergence of responsible knowledgeable society emphasizing lifelong learning with meaningful and enjoyable teaching and learning experiences; the move from reproductive model of teaching and learning to an independent, autonomous learning model that promotes initiatives, creativity, and critical thinking within dependent research. Learners are expected to collect, select, analyze, organize, extend, transform, and present knowledge using ICT in authentic and active learning paradigm. Teachers are expected to create a new flexible and open learning environment with interactive, experimental, and multimedia-based delivery system. ICT helps teachers and learners to communicate and collaborate without boundaries, make learners autonomous and allow teachers to bring the whole world into classroom activities, especially the concept of online programs. It is ultimately important to understand the roles of ICT in promoting educational changes. A basic principle is that the use of ICT changes the distribution and ownership of information resources in the space of teaching and learning and thus changes the relationship among educational participants. While designing any innovative teaching and learning environment using ICT, the teacher should always keep the learning at the center of all activities, pedagogy should be at the heart; and integration of pedagogy-technology should be the central focus. Figures 1 and 2 represent the learner's roles and curricula and delivery [15, 16].

3.2 *LSTM in TPACK*

TPACK is the framework that is nothing but the knowledge needed by the teacher for the best pedagogical practices. The main role of this work is to improve effective teaching and learning skills, so this work proposed the LSTM in the pedagogical

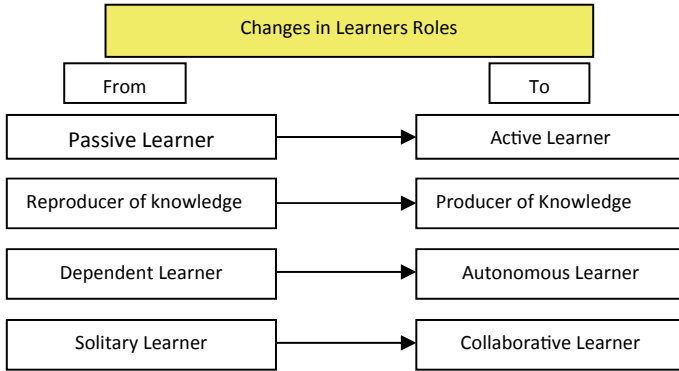


Fig. 1 Roles of learner's

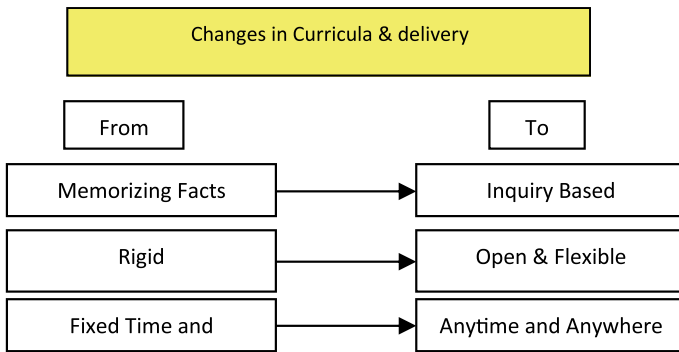


Fig. 2 Roles of curriculum and delivery

part. Because pedagogy has the power to improve the skill of the students and make them independent in the future with the effective knowledge [17]. It fully depends upon the way of teaching and methods used for teaching. Here, LSTM is used to improve the pedagogy that is the LSTM can predict the benefits of students in the future based on the teaching methods [18, 19].

By using the LSTM, the teacher can predict the output and it will help the teachers to change the way of teaching method to improve the student skills.

Deep learning employs a synthetic recurrent neural network architecture known as long short-term memory. For the first time in a neural network, LSTM has feedback connections instead of only forward connections. It can handle both whole data sequences and single data points. LSTM networks are well-suited to categorizing, processing, as well as predicting univariate time series, since significant events in time-series data may occur at indeterminate intervals. LSTMs were developed to address the issue of vanishing gradients that may arise when standard RNNs are trained.

LSTM networks are a kind of recurrent neural network, and it can learn order dependence in sequence prediction problems. The benefits of deep learning LSTMs are a difficult concept to grasp. Understanding the notion of LSTMs and how words like bidirectional and sequence-to-sequence relate to the area may be challenging [20].

- From multiple perspectives, the LSTM model has a collaret over the recurrent neural network and traditional feed-forward neural networks. This is due to their ability to remember specific designs for extended periods.
- With intricate regions, LSTM deep learning has a place. Getting your head everywhere to long short-term memory is far from a simple task. It manages algorithms that attempt to work in the same way as the human cerebrum while avoiding covering the basic connections in the given sequential data.

LSTM is mainly composed of cell state and three gates [4]:

- Forget gate
- Input gate
- Output gate.

The LSTM's memory is the cell state, which is updated by the forget gate and the input gate. In essence, the forget gate specifies what should not be stored in the memory, while the input gate specifies what should be stored there [21].

- The forget gate determines which pieces of information are irrelevant and should be ignored. The previous hidden state as well as the current time step make up the forget gate.
- What information should be included in the cell state is determined by the input gate. It is made up of both the previous hidden state and the current time step. The input gate considers two functions: the first uses a sigmoid function to filter the previous hidden state as well as the current time step.
- The output gate generates a new hidden state. The output gate uses the previous hidden state as well as the current time step, which is filtered using a sigmoid function. In parallel, the current cell state is extracted and filtered using a tanh function; both outputs are then combined to generate the new hidden state.

3.3 Prediction Method

We provide a strategy for predicting student academic achievement, as well as the attention process, in this part. With an ANN, MLP, multilayer LSTM, and proposed methods, the algorithm can accurately represent how pupils learn and absorb information. A better prediction may be gained by modifying the weights of the ANN neural network and IP-LSTM.

3.4 Artificial Neural Network

An ANN is made up of a huge number of processors that work in parallel and are stacked in layers. The raw input information is received by the first layer, which is equivalent to the optic neurons in human visual processing. In the same manner, as neurons farther away from the optic nerve get signals from those closest to it, each succeeding tier receives the output from the layer before it, rather than the raw input. The system's output is produced by the final layer.

Artificial neural networks are known for being adaptable, which means they change as they learn from their initial training and deliver more information about the environment in the future runs. Weighting the input streams, which is how each node weights the relevance of incoming data from each of its predecessors, is the simplest fundamental learning model. Inputs that help you achieve the proper responses are given more weight.

Even though the study's emphasis was on students' mental abilities, it turned out that this model presented a novel method to tackling technical problems that were not related to neurobiology. Clustering, pattern recognition, function approximation, and prediction systems have all used neural networks. There are many ANN architectures available. Feed-forward, feed-backward, single-layer, recurrent, radial basis function network, and self-organizing maps are examples of these. So ANN will also be the best model for analyzing the academic performance of students.

3.5 Multilayer Perception

A multilayer perception (MLP) is a kind of artificial neural network that has many layers. It is made up of many perceptions. They are made up of an input layer that receives the signal, an output layer that makes a judgment or prediction about the input, and an arbitrary number of hidden layers in between that comprise the MLP's real computational engine. Any continuous function may be approximated using MLPs with one hidden layer.

Multilayer perceptions are often used in supervised learning problems: they are trained on a collection of input–output pairs and learn to predict the correlation (or dependencies) between the inputs and outputs. To decrease error, the model's parameters, or weights and biases, are adjusted throughout training. Back propagation is used to change the weight and bias about the error, which may be quantified in several methods, including root mean squared error. Hence, this is also one of the best algorithms to analyze student performance.

3.6 Multilayer LSTM Neural Networks

We employ a multilayer neural network LSTM to predict as well as reflect student academic achievement. The LSTM neural network is capable of remembering critical information as well as forgetting irrelevant data. He overcomes the issue of gradient attenuation as well as gradient explosion in RNNs while dealing with extended sequences. An LSTM is made up of three gates: an input gate, an output gate, as well as a forget gate [22]. The gateway adds serial data, whereas the forgetting gateway consciously forgets the preceding node's information. The output gate specifies the data that may be transmitted to the next node.

A multilayer LSTM network is composed of three layers: (1) an input layer, (2) hidden layers, (3) hidden layers, (4) hidden layers, as well as (5) an output layer, in that order. Our LSTM employs a five-layer architecture, with three hidden layers in each of the levels.

4 Proposed Methods

We employ an IP-LSTM to this part will mine sequential patterns as well as create pattern adapters to underline the importance of critical behavioral data. Because the forget gate is used by LSTM devices and is programmed to delete any outdated or incorrect behavioral information, certain crucial additionally, behavioral data may be lost. Sequential patterns, on the other hand, include crucial behavior information that is shared by most sequences and might [23] increase the relevance of data.

There is no difference in performance between students who pass or fail the test. Students who failed the examination should be used as an example to ensure that its sequential patterns seen in students who failed the exam are rare in students who passed it. Otherwise, this accuracy of sequential pattern prediction will suffer significantly. When it comes to multi-classification sequential pattern prediction, the GSP method has flaws. As a result, we present IP-LSTM, an enhanced GSP method.

Equations (1) illustrates the steps involved in computing the output (7). The preceding unit's outputs h_{t-1} and x_t are used to create the new memory C_t . Before forgetting the gate f_t , the input gate manages the new memory's retention, the forget gate f_t regulates the retention of C_{t-1} , as well as the output gate O_t regulates its final memory's output.

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \quad (1)$$

$$f_t = \sigma(W_f X_t U_f h_{t-1} + b_f) \quad (2)$$

$$c_t = \tanh(W_2 X_t + U_z h_{t-1} + b_z) \quad (3)$$

$$C_t = i_t * z_t + f_t * C_{t-1} \quad (4)$$

$$O_t = \sigma(W_o X_t + U_o h_{t-1} + V_o C_t + b_o) \quad (5)$$

$$C_t = \tanh(C_t) \quad (6)$$

$$h_t = O_t * \tanh(C_t) \quad (7)$$

The time step in the input sequence is used when it is essential. x , and tries to are the LSTM cell's input, output, and cell state, respectively [24]. The forget selection vector is f . The input selection vector is denoted by the letter i . The weight variables in the forget, input, and output gates are W_f , W_i , W_c , and W_o . The bias variables in the forget, input, and output gates are b_f , b_i , b_c , and b_o . The sigmoid function is denoted by σ . The hyperbolic tangent function is denoted by \tanh (C_t). (*) indicates element-by-element multiplication.

This last layer of LSTM produces the prediction output. The output gate produces the hidden state, and then, the output layer produces the output of the current timestamp by applying the Softmax activation on the hidden state H_t .

$$\text{Output} = \text{Softmax}(H_t)$$

Hence, the output layer produces the prediction output by using the Softmax function. With the help of the output gate and the output layer as well as the long-term memory, it obtains the result H_t . To get it as a current timestamp output use the Softmax function, this will display the independent learner as an output. By using the LSTM, the pedagogical will be improved by getting the future results of a learner based on the current role of learner and depending on the teaching methodology.

Algorithm 1. Proposed Student Prediction Model (Improvement pedagogical Long-Short Term Memory (IP-LSTM))

Input:

T_{ik} : The set of candidate sequential patterns in the k_{th} iteration of classification i

P_{ik} : The set of frequent sequential patterns of classification i

V : The range of values for the feature sequences

rr and ar Adjustable parameters to control for the accuracy and recall of obtaining sequential patterns

Output:

CP_i The classification sequential patterns of classification i

// generating candidate sequential patterns with a length of 1 according to the range of values in the sequences

Step 1: $T_{i1} \leftarrow \text{Initialize } ()$

//If the recall (RCCi) of candidate sequential pattern f in T_{i1} is greater than rr , then f is added to P_{i1}

Step 2: **for** ($f \in T_{i1}$)

Step 3: **if** ($RCC_i(f) \geq rr$)

Step 4: $P_{i1} \leftarrow P_{i1} \cup f$

Step 5: **for** ($k = 2; P_{i,k-1} \neq \emptyset; k++$) **do**

// For sequential patterns of length $k-1$ satisfying rr , candidate sequential patterns of length k are generated.

Step 6: **for** ($f \in P_{i,k-1}$)

Step 7: **for** ($v \in V$)

Step 8: $T_{ik} = T_{ik} \cup \text{Con}(f, v)$

//The function Con connects the sequence f and value v

Step 9: **endfor**

Step 10: **endfor**

//If the RCCi of candidate sequential pattern f in T_{ik} is greater than rr , then f is added to P_{ik}

Step 11: **for** ($f \in T_{ik}$)

Step 12: **if** ($RCC_i(f) \geq rr$)

Step 13: $P_{ik} \leftarrow P_{ik} \cup f$

//In the sequence satisfying the recall requirement of classification i , if the accuracy of prediction (ACC_i) in all classifications is greater than that of ar , it is added to the final

Step 14: result CP_i .

Step 15: $CP_i = CP_i \cup \{ \{ f \} \mid f \in P_{ik}, ACC_i(f) \geq ar \}$

Step 16: **endfor**

Step 17: **return** CP_i

We can use the IP-LSTM algorithm to get many behavior models, which are the corresponding pattern adapters. For sequential patterns, we can construct an n -dimensional vector, where there are n patterns. A sequence of functions will correspond to these sequential models separately. If the match is successful, the corresponding vector position is 1. If the match fails, the corresponding vector position is 0. A student's behaviors can turn into n -dimensional vector via pattern adapters, which can be trained by a simple neural network to predict student academic performance.

4.1 Tools of Data Collection

In this part, we present the data set and detail the characteristics that were retrieved from it. After that, the distribution of data is discussed in detail. The information comes from a course named ‘CS as well as BCA BDU Core papers’, which ran from 2017 to 2020 and was taught in English. 3250 students took part in this study in total. The course comprises an MCQ exam, assignments that must be submitted within a certain time frame, attendance percentages, as well as an internal test. The students were required each unit-wise calculate academic performance.

4.2 Evaluation Metrics

The accuracy, *F1*-score, precision, as well as recall of the models employed in the experiment were all assessed using four different assessment criteria. These are the parameters that are specified as follows:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (8)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (9)$$

$$F1 \text{ Score} = 2 * (\text{Precision} + \text{Recall}) / (\text{Precision} + \text{Recall}) \quad (10)$$

True positive (TP) is the amount of information categorized as positive within the data tagged as positive in Eq. (8), as well as (8) true negative (TN) is the number of data classed as negative within the information categorized as negative [22, 24]. False-negative (FN) data are those categorized as negative but were marked as positive within the data set, while false positive (FP) data are those classified as positive and were tagged as negative in the data set [25]. The recall is the percentage of documents labeled as positive by the system out of all positively tagged data in the real world. Precision is defined as the proportion of documents labeled as positive by the model, and *F1*-score is defined as the average of precision-recall [22].

5 Results and Analysis

As seen in Table 1, the MLP approach with global parameters has a maximum accuracy of 75%. The accuracy of a multilayer LSTM technique is 89%. The findings indicate that suggested algorithms (like ANN as well as multilayer LSTM) may more accurately represent students’ learning processes as well as provide accurate predictions. In the MLP technique, the value of the variable of global feature would

Table 1 Analogies as well as contrasts between our technique as well as other methods MLP is an abbreviation for multilayer perception

Model	ANN	MLP	ML-LSTM	Proposed
Accuracy (%)	86	72	89	91
Epoch	5	15	15	20
F1-score (%)	88	77	88	90
Recall (%)	85	75	92	94
Precision (%)	82	70	89	90.5

ANN artificial neural networks, ML-LSTM multilayer LSTM, and proposed methods

be the same for distinct behaviors. The proposed method achieves an accuracy of 91%.

The MLP model demonstrates that it retains a low specificity of 70%. Finally, up to epoch 10, the suggested model displays a sharp rise inaccuracy, which then stabilizes after epoch 15. ANN, MLP, as well as multilayer LSTM performance, varies as the number of epochs grows while data size is set at 15 k, as seen in figure below. To demonstrate, how accurate it is, see Fig. 3. Up until epoch 10, the ANN model exhibits a decreasing accuracy rate of change, but beyond that point, accuracy remains constant. Up to epoch 15, the multilayer LSTM model improves accuracy. After that, the rate of improvement remains steady. The performance of the model changes as a data volume is fixed at 15 K.

The F1-score is shown in Fig. 4. The F1-score was seen in the ANN, multilayer LSTM, as well as suggested models, but only the MLP model demonstrated that the speed of change did not stabilize. The efficiency of ANN, MLP, multilayer LSTM, as well as the suggested model varies as the number of epochs grows. When the data size is set at 15 k.

The model developed in this study outperformed previous models in terms of performance, as well as accuracy grows as data quantity and training data increase. Existing models suffer from data loss and long-term dependence issues as data size increases. This method fixes such issues. Because it takes more training data as well

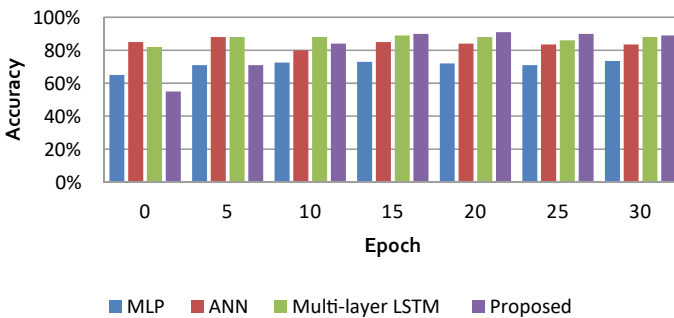


Fig. 3 Experimental results according to optimal accuracy, epoch

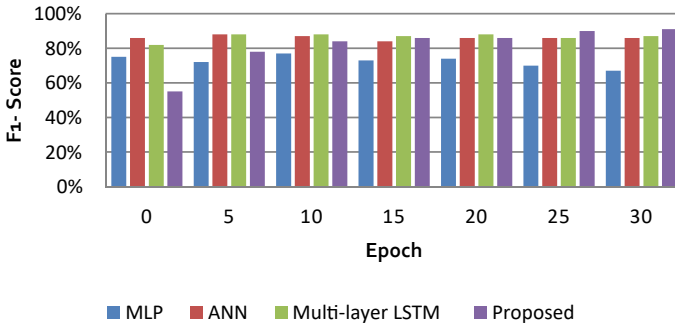


Fig. 4 Experimental results according to optimal $F1$ -score, epoch

as practice time than the previous models, the new model has just one drawback. Text classification in current years has become a representative area of sentiment analysis [26] as well as text classification for specialized disciplines, despite this constraint. However, despite this limitation, text classification may be successful in classes that need a large amount of training data (which require more inferences). Because the $F1$ -score was also not optimized, it is proven that MLP’s accuracy is much lower than other models employed in the experiment. This is because the data is a categorization of sentiment that includes both positive and negative values. Because the categorization standards are unclear, it is been determined that the learning process has not been completed. Due to recent advances in text categorization, the present multi-class MLP cannot be used for sentiment analysis.

The accuracy and recall are shown in Fig. 5. Although proposed methods The LSTM approach may cause some material to be lost since it may better represent students’ learning processes. LSTM and behavior pattern methods were used to depict the process of learning as well as emphasize the significance of essential information. Our technique outperforms the competition with a 91% accuracy rate as well as a 94% recall rate.

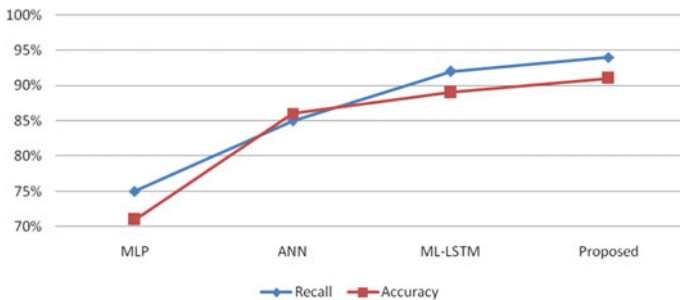


Fig. 5 Performance of models accuracy vs recall

6 Conclusion and Future Work

TPACK is one of the frameworks which visual composition and interactivity also play a role in effective learning. The TPACK combine with the machine learning technique (IP-LSTM) to predict the better results. We want to see how the learning process affects college success, and that's why we are doing this study. To predict student academic achievement, researchers built a system including an attention mechanism that incorporated data preprocessing, a technique we presented, as well as an LSTM-based neural network. IP-LSTM neural network was utilized to represent students learning processes as well as essential information relevance. In forecasting student achievement, our technique has a 91% accuracy rate as well as a 94% recall rate. As a result, our research contributes significantly to the body of knowledge on the relationship between the learning process as well as student achievement. Developing an improved algorithm for forecasting academic achievement based on recognized student profiles might be a worthwhile project in the future. This would allow instructors to provide students with more tailored and adaptable learning experiences, whether they are in person or online.

Due to the advancement of new social media tools, there are possibilities exists to develop the current ICT-TPACK enabled learning. Combination of cloud computing machine learning and artificial intelligence in the existing information and communication technology environment will give more benefit for the learners' community as well as for the educators. Effective learning can be achieved by many ways. This research work can be further enhanced by the application of student's factors such as personality, intelligence, emotional intelligence, social intelligence, socio-economic difference, and learning style difference.

References

1. Meier Y, Xu J, Atan O, Van Der Schaar M (2015) Personalized grade prediction: a data mining approach. In: Proceedings of the 2015, IEEE international conference on data mining, Atlantic City, pp 907–912
2. Yang TY, Brinton CG, Joe-Wong C, Chiang M (2017) Behavior-based grade prediction for MOOCs via time series neural networks, pp 716–728
3. Faucon L, Kidziski L, Dillenbourg PS (2016) Markov model for simulating MOOC students. In: Proceedings of the 9th international conference on educational data mining, EDM, Raleigh, USA, pp 358–363
4. Baier C, Haverkort B, Hermanns H, Katoen JP (2003) Model-checking algorithms for continuous-time Markov chains. *IEEE Trans Softw Eng*:524–541
5. Gers FA, Schmidhuber J, Cummins F (2000) Learning to forget: continual prediction with LSTM. *Neural Comput* 12:2451–2471
6. Singh B, Marks TK, Jones M, Tuzel O, Shao M (2016) A multi-stream bi-directional recurrent neural network for fine-grained action detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, Las Vegas, NV, USA, pp 1961–1970
7. Fei M, Yeung DY (2005) Temporal models for predicting student dropout in massive open online courses. In: Proceedings of the 15th IEEE international conference on data mining workshop, ICDMW 2015, Atlantic City, NJ, USA, pp 256–263

8. Qu S, Li K, Zhang S, Wang Y (2018) Predicting achievement of students in smart campus. *IEEE Access* 6:60264–60273
9. Rodrigues RL, Ramos JLC, Silva JCS, Gomes AS (2018) Discovery engagement patterns MOOCs through cluster analysis. *IEEE Lat Am Trans* 14:4129–4135
10. Brinton CG, Buccapatnam S, Chiang M, Poor HV (2016) Mining MOOC click streams: video-watching behavior vs. in-video quiz performance. *IEEE Trans Signal Process* 64:3677–3692
11. Cui T, Yuanxin O, Wenge R, Jingshuai Z, Zhang X (2018) Time series model for predicting dropout in massive open online courses. In: *Proceedings of the artificial intelligence in education. In: 19th international conference, AIED, Switzerland, 2018*, pp 353–357
12. Conijn R, Van den Beemt A, Cuijpers P (2018) Predicting student performance in a blended MOOC. *J Comput Assist Learn*:615–628
13. Pineda FJ (1987) Generalization of back-propagation to recurrent neural networks. *Phys Rev Lett*:22–29
14. Hochreiter S, Schmidhuber J (1997) Long short-term memory. *Neural Comput* 9:1735–1780
15. García EPI, Mora PM (2011) Model prediction of academic performance for first year students. In: *2011 10th Mexican international conference on artificial intelligence*, pp 169–174
16. Wilkinson D, Zhang J, Byrne GJ, Luke H, Ozolins IZ, Parker MH, Peterson RF (2008) Medical school selection criteria and the prediction of academic performance. *Med J Aust* 188(6):349–354
17. Ding M, Yeung DY, Yang K, Pong TC (2019) Effective feature learning with unsupervised learning for improving the predictive models in massive open online courses. In: *Proceedings of the 9th international conference on learning analytics and knowledge, LAK2019, Tempe, AZ*, pp 135–144
18. Cerezo R, Sánchez-Santillán M, PauleRuiz MP, Núñez JC (2016) Students' LMS interaction patterns and their relationship with achievement: a case study in higher education. *Comput Educ*:42–54
19. Kahan T, Soffer T, Nachmias R (2017) Types of participants behavior in a massive open online course. *Int Rev Res Open Distrib Learn*:1–18
20. Zhang L (2019) Evaluation of teaching effectiveness based on Gray Markov chain in the context of MOOC. In: *Proceeding SOFTHE2019 2nd international conference on advanced materials, intelligent manufacturing and automation, Zhuhai, China*, pp 42–52
21. Xiong F, Zou K, Liu Z, Wang H (2019) Predicting learning status in MOOCs using LSTM. In: *Proceedings of the 2019 ACM turing celebration conference-China, ACM TURC 2019, Chengdu, China*, pp 1–5
22. Hasan R, Palaniappan S, Raziff ARA, Mahmood S, Sarker KU (2018) Student academic performance prediction by using decision tree algorithm. In: *2018 4th international conference on computer and information sciences (ICCOINS)*, pp 1–5
23. Xiong F, Zou K, Liu Z, Wang H (2019) Predicting learning status in MOOCs using LSTM. In: *Proceedings of the 2019 ACM turing celebration conference-China, ACM TURC 2019, China*, pp 1–18
24. Gonzalez-Nucamendi A, Noguez J, Neri L, Robledo-Rella V, García-Castelán RMG, Escobar-Castillejos D (2021) The prediction of academic performance using engineering student's profiles. *Comput Electr Eng* 93:107288
25. Francis BK, Babu SS (2019) Predicting academic performance of students using a hybrid data mining approach. *J Med Syst* 43(6):1–15
26. Kassarnig V, Mones E, Bjerre-Nielsen A, Sapiezynski P, Dreyer Lassen D, Lehmann S (2018) Academic performance and behavioral patterns. *EPJ Data Sci* 7:1–16

Verification of Iris with Consideration of Constraints



Sayan Das and Biswajit Kar

Abstract In recent days, identification of people has long been an attractive target in automated reliable way. Considering this scenario, nowadays biometric person identification systems are extremely valued currently due to their terribly helpful security applications. There are numerous biometric technologies; however, the iris recognition system has been deemed the foremost reliable as a result of human irises are distinctive and can't be simply forged. One of the most used biometric recognition systems is the iris recognition system. Earlier, several ways are considered to improve the effectiveness of this systems. However, at the moment, the bulk of present recognition systems severely limit the subject's movement and make it stationary throughout the process for acquiring a good image, instead of the application of an very precise algorithm for matching patterns utilized in the verification process. Here, the various problems those are found during the iris segmentation and verification process has been discussed. However throughout the verification process, all presently on the market systems impose significant restrictions on subject movement. The existing procedure should be further developed when the iris image is not ideal, i.e., the image is taken remotely, in motion, with incorrect illumination and low resolution.

Keywords Biometrics · Ideal iris · Non-ideal iris · Segmentation · Noise removal · Threshold · Iris localization · Verification · Matching · Acquisition · Hough transform · Normalization · 1D log Gabor wavelet · Hamming distance

1 Introduction

Biometrics are machine-controlled strategies of recognizing an individual supported physiological or behavioral characteristics [15]. Biometrics involves the measure and analysis of biological knowledge or traits of the human body resembling fingerprint

S. Das (✉) · B. Kar

Central Institute of Technology Kokrajhar, Kokrajhar, Assam, India
e-mail: ph19ie1904@cit.ac.in

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
H. Sharma et al. (eds.), *Communication and Intelligent Systems*, Lecture Notes in Networks and Systems 461, https://doi.org/10.1007/978-981-19-2130-8_8

95

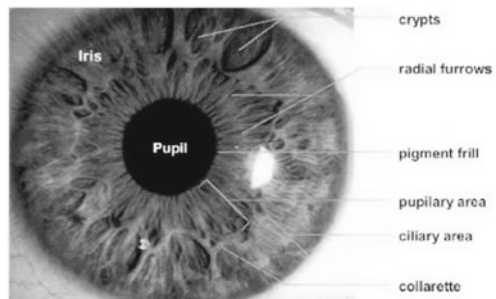
analysis, hand geometry measurements, palm print pattern, nailbeds pattern, voice characteristics, facial geometry, DNA pattern, eye retina pattern and iris pattern, and out of them, iris picture verification is that the one that is simply a great system and offers good efficiency and accuracy in distinctive people verification among the other verification systems available. Therefore, the advantages of biometrics may be classified as follows: Universality—Each person should have several of these Biometrics characteristics. Uniqueness—The biometric characteristics between two individuals are unique. Stability—The selected characteristics must be invariable within a given period of time. Easy to collect—The feature collection should be readily available. In 1987, iris recognition was first proposed by Flom and Safir [5]. In digital image processing research industries, the analysis of the iris verification systems has achieved a very nice growth. At present, iris verification system is an interesting research topic for researchers. This identification systems have lots of applications and plays a very important role in following areas, viz. managing the immigration access and records, identification and verification of person crossing sensitive national or international border, missing child identification, distribute welfare to right people, national identity card, etc.

2 The Iris

Iris is that the donut-shaped ring structure which is situated between pupil and sclera of human eyes.

It is a musculature that controls the illumination coming into the eye, with tortuous details that may be measured, resembling striations, pits, and furrows (Fig. 1). The iris recognition biometric technology uses the measurable options of the iris to form mathematical algorithms of the iris. The algorithms are then hold on and later compared with new algorithms of irises given to a capturing device for either identification or verification purposes [15].

Fig. 1 Closer look on iris



2.1 Key Advantages in Using Iris Recognition are as Follows

There are very good calculation advantages of an iris pattern, because of its completely different structure varies between people [3]. The iris pattern is very stable in a given era of time, as it is enclosed (yet outwardly visible) eye organ which is well shielded from our atmosphere. Iris is a planar object. It shows good image consistency in various angle of illumination. The localization of iris in a face is an easy process as it is a part of eye. Mainly one can achieve good features from its distinctive donut-shaped structure and size. The structure and the unique features are formed by the 8th month of pregnancy and that's why this verification process is applicable to babies and children too. It's complicated pattern will contain several distinctive options resembling zigzag collarette, ridges, arced ligaments, freckles, crypts, furrows, corona and rings [3]. Available industrial systems applies John Daugman algorithm developed on 1995 [3] for identification of stationary iris. Present systems restrict the subject movement throughout the verification and identification process, to get stable images of an iris. Researchers applies simple pattern matching algorithms for the matching process. So, the prevailing procedure desires any development once the iris image isn't the best one, i.e., image is taken from distance, in motion, improper illumination and low resolution.

3 Literature Survey

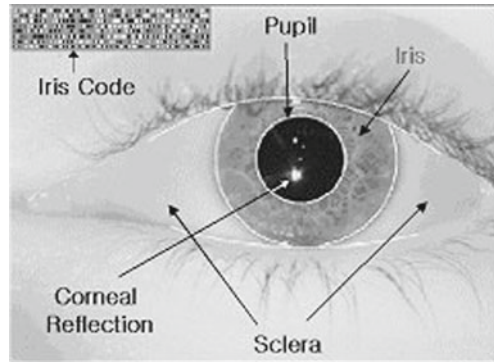
A reliable iris localization formula planned by Jan et al. [7] throughout which they focuses on two primary noises, i.e., eyelids and eyelash occlusions. They found out the pupil and iris boundaries to segment iris properly. Zhang et al. [20] proposed in their work to reconstruct iris pixels those are occluded by eyelashes mistreatment info from their non-occluded neighbors and reconstruct the information to generate informative iris code. Shirke et al. [11] proposed to work with super resolution when the images may be degraded because of absence of texture, blur or low resolution. Purva M. Barve, in his work, [16] focused on non-ideal iris pre-processing part where the author developed an algorithm in pre-processing stage to handle the irises' images which are acquired in non-ideal way.

4 Iris Verification System

It has been shown that among various biometric processes, iris verification process shows very good reliability, accuracy and performance.

A good iris recognition algorithm depends mainly on the proper segmentation of the donut-shaped iris hence the reliability and accuracy increase eventually. Typically, iris verification process comprises four modules, viz. segmentation of iris from an

Fig. 2 Iris, pupil, eyelid detection and iris code



image, normalization of the segmented image to polar representation, proper extraction of good features and matching of two iris patterns [9]. For real-time application, the process time of the system ought to additionally consider. Iris code [15] is that the hexadecimal representation of the iris, keep a 512-byte template as shown in Fig. 2. Iris of an individual is generally of about 11 millimeter in diameter. In standard algorithms, iris code has 3.4 bits of data per square millimeter. One can extract distinct 266 features from an iris image. In following sections, step-by-step iris verification steps are discussed. The corresponding flowchart is shown in Fig. 3.

4.1 Iris Acquisition

This is the first and most important step in iris identification system. One camera is used to acquire the image which is mounted between 90 and 300 cm far from the person of interest. Then the high-definition multiple iris image shots are captured [15]. Image generally taken by infrared camera, because infrared image will give lucid patterns and blocks visible light reflections and as well as the illumination using visible spectrum may make irritation to the eye which makes problem in proper iris image acquisition. Image acquisition depends on the type of recognition technique used and can be said that image acquisition can be defined as acquiring an image from the subject of interest. For further processing, the captured image is then converted into gray-scale image composed of shades of gray (neutral), varied from black to white, where black represents the weakest intensity and white side represents the strongest.

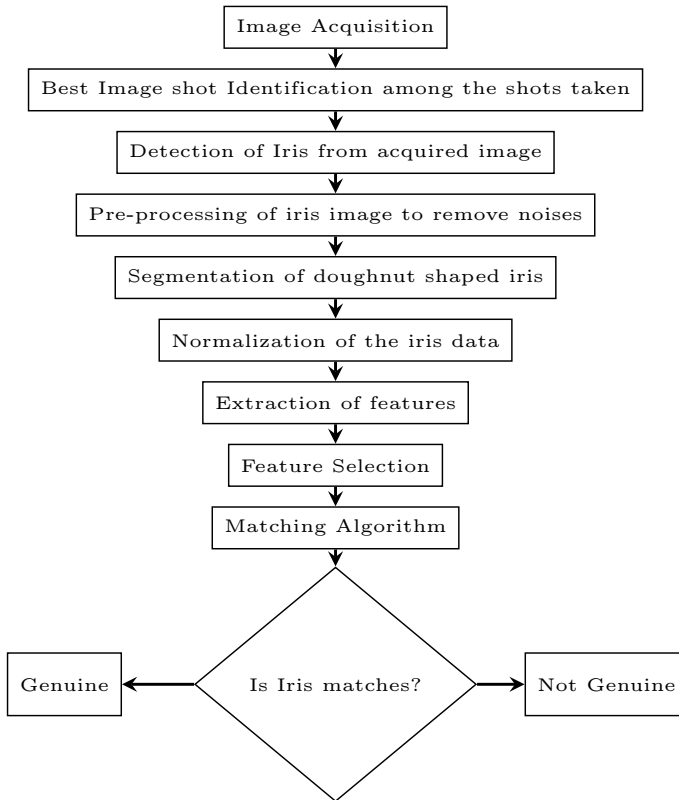


Fig. 3 Basic iris verification stages

4.2 Iris Segmentation

Iris segmentation is that the method to discover the boundaries of iris and pupil so the donut-formed iris is often segmented from the image. There are some strategies can be used for the segmentation, those are as follows:

- Circular Hough transform—The circular Hough transform [19] are often used to deduce the donut-shaped iris by finding the radius and center coordinates of the both, i.e., the pupil and iris regions.
- Hough transform for line for eye lid detection—Detecting the eye lids and masking them [15] will also reduce the unwanted edges and results in better accuracy of recognition further.
- Daugman's integro-differential operator—Daugman's integro-differential operator [4] uses a circular integral to search for the parameters of circles formed where the integral derivative is at its maximum. The integro-differential operator is modeled as

$$r = a \cos \theta + b \sin \theta \quad (1)$$

$$-(x - h_l) \sin \theta_l + (y - k_l) \cos \theta_l)^2 = a_l((x - h_l) \cos \theta_l + (y - k_l) \sin \theta_l) \quad (2)$$

$$\max(r, x_q, y_0) \left| G_\sigma(r) * \frac{\partial}{\partial r} \oint_{r, x_0, y_0} \frac{Im(x, y)}{2\pi r} ds \right| \quad (3)$$

where $Im(x, y)$ = Acquired eye image,

r = Radius to search for,

$G_\sigma(r)$ = Gaussian smoothing function, and

s = Contour of the circle given by (r, x_0, y_0) .

4.3 Normalization

Normalization has to be done because of the inconsistency of pupil radius due to variation of light intensities and have to make donut-shaped iris image a dimensional inconsistent to variation of intensity to make the donut-shaped iris a planar one. Daugman's rubber sheet model mainly used to perform this operation.

Daugman's rubber sheet model—Using the Daugman's rubber sheet model, the iris region nullifying the effects of dimensional inconsistencies is represented. With the help of this algorithm, polar conversion of the iris image is performed. Daugman uses linear interpolation method to perform the same. It separates the donut-shaped iris annular ring. This process eliminates pupil and the other non-interesting regions.

Daugman's rubber sheet model [12] may be used to perform this operation. The iris region Cartesian coordinates (x, y) representation is remapped into the normalized non-concentric polar illustration by the use of the following mathematical model:

$$I(x(r, \theta), y(r, \theta)) \rightarrow I(r, \theta) \quad (4)$$

with

$$x(r, \theta) = (1 - r)x_q(\theta) + rx_1(\theta) \quad (5)$$

$$y(r, \theta) = (1 - r)y_q(\theta) + ry_1(\theta) \quad (6)$$

4.4 Feature Extraction and Encoding

Any of the following methods or multiple methods are combinedly used to extract and encode the distinct feature:

- Zero crossings of the 1D wavelet
- Log Gabor filters
- Wavelet encoding
- Gabor filters
- Haar wavelet
- Laplacian of Gaussian filters

4.5 Matching

Matching technique is used to match the trial template with the reference template to verify the genuineness. Several methods are there and out of some of them are described below:

- The Hamming Distance (HD)—The HD algorithmic program measures the similarity percentage between 2 bit patterns. Using the HD of 2 bit patterns, a decision is made, whether two irises are same or they are different. If two bit patterns are taken, i.e., pattern X and pattern Y, then, HD, is outlined as the sum of the XOR between pattern X and pattern Y over whole number N within that bit pattern.
- Weighted Euclidean distance (WED)—Two feature templates are compared in the WED, particularly if the template consists of integer values. Zhu et al. utilized this method for matching the templates [18] and is specified as

$$\text{Hamming Distance(HD)} = \frac{1}{N} \sum_{l=1}^N X_l (XOR) Y_l \quad (7)$$

$$\text{WED}(k) = \sum_{j=1}^N \frac{(f_j - f_j^{(k)})^2}{(\sigma_j^{(k)})^2} \quad (8)$$

where $f_j = j$ th feature of the unknown iris,

$f_j^{(k)} = j$ th feature of iris template k and

$\sigma_j^{(k)} = \text{SD}$ of the j th feature in the iris image template k , where SD is the standard deviation.

If WED is minimum in template k , then the unknown iris template is matched to the iris template k .

5 Specific Constraints in Iris Segmentation

Iris recognition is now a commercially demanding biometric verification system. But there are some constraints which are described as follows:

5.1 Image Acquisition

Image acquisition is the most significant stage [17] in iris recognition system. Because all the successive stages rely highly on the image quality. If image quality is not the ideal one then a good pre-processing algorithm is required before applying the segmentation algorithm. Proper image acquisition depends on some parameters discussed as following [17] such as:

- **Illumination**—Proper illumination is required to capture the image of the iris, entirety. The visible light span of a human eye is between the wavelength ranging between 400 and 760 nm [20], so to avoid the user uneasiness, the light source is kept in the nearby infrared band, i.e., between 700 and 900 nm.
- **Focal length**—Focal length usually determines the angle of view. By varying the angle, the image can be magnified. Longer focal length produces the higher magnification. Shorter focal length produces lower magnification during image acquisition.
- **Aperture**—This determines how much light will enter into the lens by opening or closing the camera lens using shutter.
- **Depth of field (DOF)**—DOF determination of an image is that the measure of the distance between the closest and also the objects which are farthest in the image, that are in tolerably sharp in focus in associated image. The depth of field calculation is done by determining the following parameters, i.e., aperture of camera, focal length of camera, distance to subject from camera and the suitable circle of confusion size.
- **Pixel of Resolution**—Image resolution determines how much details an image is holding.

$$\text{DOF} = \frac{2\mu^2 Fc}{f^2} \quad (9)$$

where c = circle of confusion,

f = focal length,

F = f -number, and

μ = distance to subject.

5.2 *Factors to Be Considered in Pre-processing of Non-ideal Iris*

- Angle estimation—In case of non-ideal iris if the iris is off-angled then it has to be corrected or some measure to be taken to detect the iris.
- Reflection handling—During the iris image acquisition the eye is illuminated by proper light source which introduce reflection on the iris image and that is treated as a noise which should be taken care.
- Eyelid/eyelash Occlusion—As per Al Zubi et al., they classified the eyelashes into two types [1]. One is separable eyelashes. These eyelashes are in isolated form in the acquired image, and the second is the multiple eyelashes, which are causing the main problems as they are bunched together and overlap in the iris image. Upper and lower eyelid is also detected and removed during segmentation.
- Illumination compensation—Good illumination on eye make acquisition easier.
- Motion deblurring—When the iris is in movement, the image acquisition process is so complicated. The motion blurring is a key factor to be corrected using the deblurring filters.
- Super-resolution—Iris recognition process requires an input image of good resolution, where the iris size should be greater than 200 pixels to achieve good performance [8]. The camera which is equipped with the zoom lens of longer focal length and the camera sensor of the greater number of pixels is used for iris recognition at a distance [10]. To decrease costs and size, a camera without a zoom lens can be used in an iris recognition system, but this causes a reduction in recognition accuracy by capturing a low-resolution image. Therefore, image restoration is needed in case of use of a camera without a zoom lens and this can be achieved using the super-resolution image restoration algorithm.
- Image localization—Iris localization in a minimal execution time is another key factor to be considered.

6 Results and Discussion

The ideal and non-ideal iris images are collected from various biometric databases [2, 6, 13, 14], and the following algorithm has been applied on randomly selected images out of them.

1. Step 1: Iris segmentation using circular Hough transform where the iris part is extracted from the eye.
2. Step 2: To generate iris code the normalization has been done using Daugman's rubber sheet model. Feature extracting process for feature extraction process and matching:
3. Step 3: Image registration by Wildes method.
4. Step 4: Encoding of significant features by 1D log Gabor wavelets.
5. Step 5: Matching of significant features by Hamming distance.

Table 1 Each step output in image form for the work done on ideal iris





















Image	Original image	Segmented iris	Upper and lower eyelid detection	Noise removal	Normalization
1					
2					

Table 2 Each step output in image form for the work done on non-ideal iris

Image	Original image	Segmented iris	Upper and lower eyelid detection	Noise removal	Normalization
1					
2					

The algorithm shows a good verification result when two ideal iris images of same person is compared. Following are the glimpse of some results that have found after applying the algorithm mentioned with ideal and non-ideal iris data.

Following are the glimpse of some results that have found after applying the algorithm mentioned with ideal and non-ideal iris data.

In Tables 1 and 2, step-wise output is shown after application of the above said algorithm on an ideal iris image and non-ideal iris image, respectively.

7 Conclusion

Work done on both the iris data, i.e., ideal and non-ideal iris. In this iris verification algorithm, the image segmentation algorithm is used. That algorithm would localize the iris region from an eye image, isolate eyelashes, reflection areas and detect eyelids and remove them. From the eye image, the circular Hough transform localize the donut-shaped iris. Threshold algorithm is employed here for isolating eyelashes and reflections. When algorithm is applied on ideal iris where the iris images are taken with intensive care, so that, they are not occluded, none or less eyelash affected, having good illumination, it is found that this algorithm works very efficiently on

them. In this work, iris images which are off-angled, having large eyelashes or they are taken from distance, i.e., with poor illumination and low resolution. Whenever the algorithm is applied stated above, it could not localize the iris properly during localization. It is seen that during iris localization or segmentation more or less area rather than area of interest is localized and segmented which adds impurities in iris code hence resulting less accurate iris verification.

References

1. Al-Zubi RT, Darabkh KA, Al-Zubi N (2015) Effect of eyelid and eyelash occlusions on a practical iris recognition system: analysis and solution. *Int J Pattern Recogn Artif Intell* 29(8):1–24. <https://doi.org/10.1142/S0218001415560169>
2. Centre for Biometrics and Security Research (CBSR). CASIA iris database. <http://www.cbsr.ia.ac.cn/english/IrisDatabase.asp>
3. Daugman J (2009) How iris recognition works. *Essential Guide Image Process* 14(1):715–739. <https://doi.org/10.1016/B978-0-12-374457-9.00025-1>
4. Djoumessi M (2016) Iris segmentation using Daugman’s integro-differential operator
5. Flom L, Safir A. Iris recognition system
6. Institute of Automation, Chinese Academy of Sciences. CASIA iris image database. http://english.ia.cas.cn/db/201610/t20161026_169399.html
7. Jan F, Usman I, Agha S (2013) Reliable iris localization using Hough transform, histogram-bisection, and eccentricity. *Signal Process* 93(1):230–241. <https://doi.org/10.1016/j.sigpro.2012.07.033>
8. Matey JR, Naroditsky O, Hanna K, Kolczynski RA, Loiacono DJ, Mangru S, Tinker M, Zappia TM, Zhao WY (2006) Iris on the move: acquisition of images for iris recognition in less constrained environments. *Proc IEEE* 94(11):1936–1946. <https://doi.org/10.1109/JPROC.2006.884091>
9. Paul PP, Monwar MM, (2007) Human iris recognition for biometric identification. In: 10th international conference on computer and information technology. ICCIT. <https://doi.org/10.1109/ICCITECHN.2007.4579354>
10. Shin KY, Kang BJ, Park KR (2010) Super-resolution iris image restoration using single image for iris recognition. *KSH Trans Internet Inf Syst* 4(2):117–137. <https://doi.org/10.3837/tiis.2010.04.003>
11. Shirke SD, Rajabhushnam C (2019) Biometric personal iris recognition from an image at long distance. In: Proceedings of the international conference on trends in electronics and informatics, ICOEI 2019, Apr 2019. ICOEI, pp 560–565. <https://doi.org/10.1109/ICOEI.2019.8862640>
12. Tian Q, Liu Z, Li L, Sun Z (2006) A practical iris recognition algorithm. In: 2006 IEEE international conference on robotics and biomimetics, ROBIO 2006, pp 392–395. <https://doi.org/10.1109/ROBIO.2006.340208>
13. UBIRIS Noisy Visible Wavelength Iris Image Databases. <http://iris.di.ubi.pt/>
14. UPOL Iris Database. Phoenix—UPOL. <http://phoenix.inf.upol.cz/iris/>
15. Vacca JR. Biometric technologies and verification systems. Elsevier
16. Virginia W (2005) Robust pre-processing techniques for non-ideal iris images
17. Wibowo EP, Maulana WS (2009) Real-time iris recognition system using a proposed method. In: International conference on signal processing systems. ICSPS 2009, June, pp 98–102. <https://doi.org/10.1109/ICSPS.2009.9>
18. Yong Zhu TT, Wang Y (2006) Biometric personal identification based on iris recognition. In: 2006 international conference on computer engineering and systems, ICCES’06 (59825105), pp 208–213. <https://doi.org/10.1109/ICCES.2006.320449>

19. Yuan X, Shi P (2004) An iris segmentation procedure for iris recognition. *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics)* 3338:546–553. https://doi.org/10.1007/978-3-540-30548-4_62
20. Zhang D, Monro DM, Rakshit S (2006) Eyelash removal method for human iris recognition. In: *Proceedings—international conference on image processing, ICIP*, pp 285–288. <https://doi.org/10.1109/ICIP.2006.313181>

Theoretical Validation of Data Warehouse Requirements Metrics Based on Agent Goal Decision Information Model Using Zuse's Framework



Tanu Singh and Manoj Kumar

Abstract Requirement's data models' quality is one of the main data models that influence quality of information stored in data warehouse to take major decisions in organization. Thus, it becomes significant for an organization to maintain and assure the information quality of data warehouse. Very few proposals were seen in the literature for assuring quality of requirements data model of DW. However, no theoretical validation of DW requirements metrics using Zuse's framework was seen in the literature. Hence, in this paper, theoretical validation of requirements traceability metrics (based on agent goal decision information model) is done by applying Zuse's framework. Results indicate that all traceability metrics are valid and correct. Thus, requirements traceability metrics can be used for assuring quality of DW requirements model.

Keywords Agent goal decision information model · Data warehouse quality · Requirements engineering · Requirements traceability metrics · Theoretical validation · Zuse's framework

1 Introduction

Data warehouse (DW) is a data-driven powerful tool for quick decision-making architecture in an organization [1]. As this information is required for making organizational decisions, hence, it is important to uphold DW information quality [2]. Mainly, information quality of DW is based on the quality of its data models (i.e., requirements, conceptual, logical, and physical) [3, 4]. Hence, many researchers

T. Singh (✉)

University School of Information, Communication and Technology, Guru Gobind Singh Indraprastha University, Delhi, India

e-mail: tanusnigam@gmail.com

M. Kumar

Department of Computer Science and Engineering, Netaji Subhas University of Technology, East Campus (Formerly Ambedkar Institute of Advanced Communication Technologies and Research), Delhi, India

e-mail: manoj.kumar@nsut.ac.in

proposed development methodologies for data models' quality, but these were not sufficient for evaluating information quality of DW.

Hence, these methodologies were theoretically and empirically validated using metrics. Accordingly, to evaluate the data models quality at physical model level [1, 5], at logical model level [6, 7] at conceptual model level [8, 9], and at requirements model level [10–14] have been witnessed in the literature.

Theoretical validation includes mathematical properties which need to prove by each metric, whereas experimental validation of metric assists to illustrate the practical usefulness to measure the external attributes, i.e., maintainability, understandability, etc. Hence, in this direction, theoretical validation of traceability metrics based on agent goal decision information (AGDI) model [15] using Zuse's framework [16] is carried out to prove that (i) they are defined correctly and (ii) they are related with external attributes in predictable ways which can be further used for evaluating the DW quality.

The paper structure is represented as: related work of requirements data model quality of DW is discussed in Sect. 2. The explanation of requirements traceability metrics are done in Sect. 3. Section 4 explained the theoretical validation of traceability metrics using Zuse's properties. The results of theoretical validation are discussed in Sect. 5, and in last, Sect. 6 highlight the conclusion of the paper.

2 Related Work

From literature, a lot work was witnessed for estimating the data models quality at physical, logical, conceptual, and requirements model level. As this paper only focuses on the requirements data model quality; hence, theoretical validation of only DW requirements data model is discussed below.

Requirement's traceability metrics was validated formally by Briand's property-based framework [11] where only size measure was applied. In addition, cohesion and coupling property of Briand's framework were investigated formally on traceability metrics by [13]. Recently, requirements completeness metrics were formally validated by employing Briand's framework [14]. From above, it is witnessed that formal validation of metrics was performed only using Briand's framework. However, other theoretical frameworks are not still proved on these traceability metrics. This inspired us to validate the traceability metrics theoretically formed on AGDI model [15] by implemented Zuse's framework properties. The requirements metrics which are used in this study are explained in the next section.

3 Metrics Used for Validation

This section explains the requirements traceability metrics presented by [10]. The requirements traceability metrics based on AGDI model and are majorly in twofold (i) coverage (COV) metrics and (ii) depth and height coverage (DHCOV) metrics. The AGDI model was designed using requirements engineering (RE) approach to capture both requirements, i.e., early and late. Here, organization and decision modeling activities fall under early requirements, and information modeling activities fall under late requirements.

We have created various requirements schemas of different verticals out of all which are taken into considerations are manufacturing, library, banking, university, railways, ecommerce-company, IT industry, etc. Here, in this paper, to explain AGDI

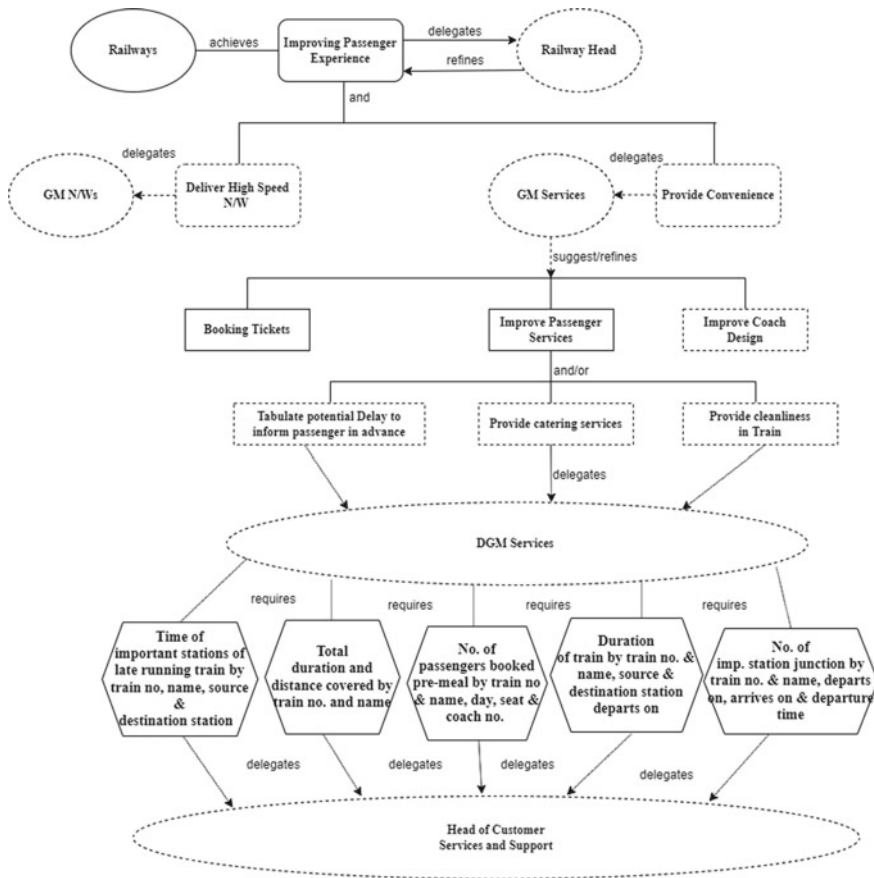


Fig. 1 Railways requirements model of DW

Table 1 Computed values of traceability metrics for railways requirements model

Requirements traceability metrics	Description	Values	
COV metrics	NGD	Number of simple goals tracing downwards to the simple decisions	1
	NDI	Number of simple decisions tracing down to the information requirements	3
	NDG	Number of simple decisions tracing toward up to simple goals	3
	NID	Number of information requirements which trace up to simple decisions	5
	NDGI	Number of decisions (simple) tracing toward up to the simple goals and down to information requirements	3
DHCOV metrics	NIG	Number of information which trace up to the simple or complex goals	5
	NGI	Number of complex or simple goals tracing down to the information	1

model, we have chosen railways requirements model (see Fig. 1) and computed the values of these traceability metrics with its description (as depicted Table 1).

4 Theoretical Validation of Metrics Using Zuse's framework

This section describes the properties of Zuse's framework [16] which is satisfied by the requirements traceability metrics. This framework implements the notion of measurement theory to discover the scale corresponding to the metric. However, in this concept, one should know the tendency to calculate empirical relations for, e.g., "greater than," "greater than or equal to," "less than," etc. between the objects. Let's assume, an empirical relational system $(A, \bullet \geq, \circ)$ where A belongs to set of not empty objects, $\bullet \geq$ belongs to binary empirical relation to A , and \circ is a concatenation function, i.e., closed binary relation on A . Now, let's assume a numerical relational system $(B, \geq, +)$ where B belongs to set of real numbers, \geq belongs to binary relation to B , and $+$ is a binary operation on B . Therefore, for measure m the definition of a mapping function is as follows:

$m: A \rightarrow B$ such that $\forall A_1, A_2 \in A$ then below mentioned two conditions are true, i.e.,

- a. $A_1 \bullet \geq A_2 \iff m(A_1) \geq m(A_2)$
- b. $m(A_1 \circ A_2) = m(A_1) + m(A_2)$

Thus, (A, B, m) is defines as a scale obtaining to measure m .

Table 2 Mathematical properties of Zuse's framework [16]

Zuse's properties	Definition
MES—Axiom 1	$(A \bullet \succeq)$ (weak order)
MES—Axiom 2	$A1 \circ A2 \bullet \succeq A1$ (positivity)
MES—Axiom 3	$A1 \circ (A2 \circ A3) \approx (A1 \circ A2) \circ A3$ (weak associativity)
MES—Axiom 4	$A1 \circ A2 \approx A2 \circ A1$ (weak commutativity)
MES—Axiom 5	$A1 \bullet \succeq A2 = > A1 \circ A \bullet \succeq A2 \circ A$ (weak monotonicity)
MES—Axiom 6	If $A3 \bullet > A4$ then for any $A1$ and $A2$, then there exists a natural number n , such that $A1 \circ nA3 \bullet > A2 \circ nA4$ (Archimedean axiom)
Independence C1	C1: $A1 \approx A2 = > A1 \circ A \approx A2 \circ A$ and $A1 \approx A2 = > A \circ A1 \approx A \circ A2$
Independence C2	C2: $A1 \approx A2 \iff A1 \circ A \approx A2 \circ A$ and $A1 \approx A2 \iff A \circ A1 \approx A \circ A2$
Independence C3	C3: $A1 \bullet \succeq A2 = > A1 \circ A \bullet \succeq A2 \circ A$, and $A1 \bullet \succeq A2 = > A \circ A1 \bullet \succeq A \circ A2$
Independence C4	C4: $A1 \bullet \succeq A2 \iff A1 \circ A \bullet \succeq A2 \circ A$, and $A1 \bullet \succeq A2 \iff A \circ A1 \bullet \succeq A \circ A2$ where $A1 \approx A2$ if and only if $A1 \bullet \succeq A2$ and $A2 \bullet \succeq A1$, and $A1 \bullet > A2$ if and only if $A1 \bullet \succeq A2$ and not $(A2 \bullet \succeq A1)$
MRB1	$\forall A, B \in \mathfrak{S}: A \bullet \succeq B$ or $B \bullet \succeq A$ (completeness)
MRB2	$\forall A, B, C \in \mathfrak{S}: A \bullet \succeq B$ and $B \bullet \succeq C = > A \bullet \succeq C$ (transitivity)
MRB3	$\forall A \supseteq B = > A \bullet \succeq B$ (dominance axiom)
MRB4	$\forall (A \supset B, A \cap C = \emptyset) \mathfrak{P} (A \bullet \succeq B = > A \cup C \bullet \succeq B \cup C)$ (partial monotonicity)
MRB5	$\forall A \in : A \bullet \succeq 0$ (positivity)

Also, there are three main mathematical properties of Zuse's framework [16] for measuring the software, i.e., modified extensive structures (MES), independence conditions (IC), and modified relation of belief (MRB). These mathematical properties are defined in Table 2. Based on these mathematical properties one can calculate the scale of metric. Hence, if the metric satisfy the MES properties, then it said as ratio scale. In second case, if the metric satisfy the IC properties but not satisfy the MES properties, it said as ordinal scale. In last case, if the metric satisfy the MRB properties, then it said as "above" ordinal scale. In addition, if metric classify as above ordinal scale then only it is considered to be effective and useful because we cannot do very much with ordinal numbers [16].

For validating requirements traceability metrics, assume an empirical relational system represented as $CG = (CG, \bullet \succeq, \circ)$, where CG denotes number of complex goals (non-empty set), $\bullet \succeq$ denotes binary empirical relation on CG and \circ denotes concatenation operation, i.e., closed binary relation on CG . The complex goals are further refines into simple goals and these simple goals are traced down to the decision and information requirements. Here, formal validation of two metrics are shown, i.e., NGD and NDI metrics using Zuse's framework [16] which is shown below. Further, NID, NIG, and NGI metrics can also be validated theoretically exactly similar to NGD metric and are "above" ordinal scale, whereas NDG and NDGI metrics can be validated theoretically exactly similar to NDI metric and are "above" ordinal scale.

4.1 Formal Validation of NGD Metric

According to Zuse’s framework, to illustrate an empirical relation [16] introduced new concatenation operations. Here, in this section, a closed binary concatenation (\circ) is defined in *GoalsConcat*. Using this operation, the concatenation of goals G_1 and G_2 leads to a goal G which includes all the G_1 and G_2 features, suppressing commonalities (if it exist) in the main goals, i.e., in resulting goal G show common features only once (see Fig. 2).

The NGD traceability metric is measured as a function of mapping: $NGD: G \rightarrow D$ (G and D denotes the number of complex goals (as shown in Fig. 2, as bold rounded rectangle) and decisions, respectively), such that it holds all the relations between G_1 and G_2 .

$$G_1 \text{ and } G_2 \in G: G_1 \bullet \geq G_2 \Leftrightarrow NGD(G_1) \geq NGD(G_2)$$

where G_1 and G_2 are simple goals (as shown in Fig. 2 as dotted rounded rectangle).

According to the concatenation rule, $NGD(G_1 \circ G_2) = NGD(G_1) + NGD(G_2) - v$, i.e., the number of goals tracing down to decisional hierarchy achieved after concatenation of G_1 and G_2 , equals to the total number of goals tracing down to decisional hierarchy of G_1 and G_2 , considering the common decisional hierarchy to G_1 and G_2 , only once and are not affected in other cases.

NGD as an MES Six axiom of MES are described below and also explained in Table 2.

Axiom 1 (Week order): Let’s assume three goals, i.e., G_1 , G_2 , and G_3 of a requirements model and it is obvious that:

$$NGD(G_1) \geq NGD(G_2) \text{ or } NGD(G_2) \geq NGD(G_1) \text{ and also, 'if } NGD(G_1) \geq NGD(G_2) \text{ and } NGD(G_2) \geq NGD(G_3) \Rightarrow NGD(G_1) \geq NGD(G_3).$$

Hence, NGD satisfy the first axiom of MES.

Axiom 2 (Positivity): This axiom is satisfied by the NGD metric. The concatenation of G_1 and G_2 is always equal to or greater than $NGD(G_1)$, if there exist non-hierarchical decisions in G_2 . Hence, can be written as

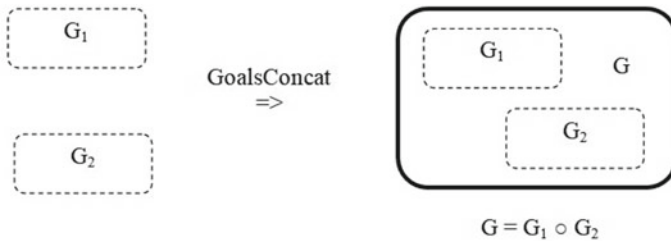


Fig. 2 Concatenation of two goals: *GoalsConcat*

$$\text{NGD}(G_1 \circ G_2) \geq \text{NGD}(G_1).$$

Axiom 3 (Weak associativity) and *Axiom 4 (Weak commutativity)* are satisfied by the NGD metric because the natural join operation is both commutative and associative as the definition of *GoalsConcat*. The order in which the goals are concatenated; it will not affect the count of goals tracing downwards to the decisional hierarchy.

Axiom 5 (Weak monotonicity): This axiom is not satisfied by NGD metric. As, on concatenating the goal (G) to both G_1 and G_2 , $\text{NGD}(G_1 \circ G)$ value may not be greater than or equal to $\text{NGD}(G_2 \circ G)$ value; because some decisional hierarchy of G common to G_1 and not G_2 exist, leading to suppressing $\text{NGD}(G_1 \circ G)$ value in comparison to value of $\text{NGD}(G_2 \circ G)$.

Axiom 6 (Archimedean): NGD metric not satisfied this axiom as goal concatenation; i.e., *GoalsConcat* is idempotent (operation applied multiple times without changing the result). Thus, concatenation of G_1 with G_2 (i times) is equal to concatenation of G_1 with G_2 (one time).

NGD as an IC The properties of IC are described below and also explained in Table 2.

As, *axiom 5 (Weak monotonicity)* is not satisfied, then the NGD metric does not satisfy the $C1$ and it cannot satisfy $C2$ condition. Similarly, the third condition, i.e., $C3$ does not satisfy because axiom 5 was not fulfill and if it does not satisfy $C3$, it cannot satisfy the $C4$ condition. Hence, NGD metric does not satisfy these IC properties.

NGD as an MRB Five properties of MES are described below and also explained in Table 2.

As NGD satisfy axiom 1, then the first (*MRB1*) and second (*MRB2*) property, i.e., completeness and transitivity are satisfied.

NGD satisfy property (*MRB3*), i.e., dominance as the overall decisions of G_2 are comprised in G_1 , this means all the hierarchical decisions of G_2 are in G_1 . So, we can say that,

$$\text{NGD}(G_1) \geq \text{NGD}(G_2).$$

MRB4, i.e., partial monotonicity is satisfied by the NGD metric because $G_1 \subset G_2$, then $\text{NGD}(G_1) > \text{NGD}(G_2)$. Now, assume G_3 having non-hierarchical decisions with G_1 . Then, G_3 cannot have common decision hierarchy with G_2 also. Hence, we can say that

$$\text{NGD}(G_1 \circ G_3) > \text{NGD}(G_2 \circ G_3).$$

At last, *MRB5* is satisfied by NGD metric, as if G has no decisional hierarchy, then $\text{NGD}(G) = 0$, but never can be negative.

Hence, NGD metric is “above” the ordinal scale as this metric satisfy all the properties of MRB. Further, NID, NIG, and NGI traceability metrics can be validated theoretically exactly like NGD metric and said to be “above” ordinal scale.

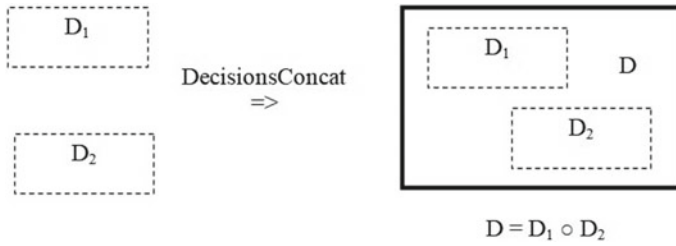


Fig. 3 Concatenation of two decisions: *DecisionsConcat*

4.2 Formal Validation of NDI Metric

Based on Zuse’s framework, for demonstrating an empirical relation, a new concatenation operation was introduced. Here, in this section, a closed binary concatenation (\circ) is defined in *DecisionsConcat*. Using this operation, the concatenation of decisions D_1 and D_2 leads to a decision D which includes all the D_1 and D_2 features, suppressing commonalities (if it exist) in the main decisions; i.e., it shows all common features only once in decision D (see Fig. 3).

The NDI traceability metric is defined as a function of mapping: $NDI: D \rightarrow I$ (D and I denotes the number of complex decisions (as shown in Fig. 3, with bold rectangle) and information requirements, respectively), such that it holds all the relations between D_1 and $D_2 \in D: D_1 \bullet \geq D_2 \iff NDI(D_1) \geq NDI(D_2)$, where D_1 and D_2 are simple decisions (as shown in Fig. 3, with dotted rectangle).

According to the concatenation rule, $NDI(D_1 \circ D_2) = NDI(D_1) + NDI(D_2) + v$, i.e., decisions which trace downwards to hierarchy of information requirements achieved after concatenation of D_1 and D_2 , equals to the total decisions which trace downwards to hierarchy of information requirements of D_1 and D_2 , considering the common information hierarchy to D_1 and D_2 , only once and are not affected in other cases.

NDI as an MES Six properties of MES are explained below and also explained in Table 2.

Axiom 1 (Week order): Let’s assume three decisions, i.e., D_1, D_2 , and D_3 of a requirements model and it is obvious that

$$\begin{aligned}
 & NDI(D_1) \geq NDI(D_2) \text{ or} \\
 & NDI(D_2) \geq NDI(D_1) \text{ and also, if} \\
 & NDI(D_1) \geq NDI(D_2) \text{ and} \\
 & NDI(D_2) \geq NDI(D_3) \Rightarrow NDI(D_1) \geq NDI(D_3).
 \end{aligned}$$

Hence, the first axiom of MES is satisfied by NDI metric.

Axiom 2 (Positivity): This axiom is satisfied by the NDI metric. The concatenation of D_1 and D_2 is always equal to or greater than to $NDI(D_1)$, if non-hierarchical informations exist in D_2 . Hence, can be written as

$$\text{NDI}(D_1 \circ D_2) \geq \text{NDI}(D_1).$$

Axiom 3 (Weak associativity) and *Axiom 4 (Weak commutativity)* are satisfied by the NDI metric because the natural join operation is both commutative and associative as the definition of *DecisionsConcat*. The order in which the decisions are concatenated; it will not affect the sum of decisions which trace down to the hierarchical information requirements.

Axiom 5 (Weak monotonicity): This axiom is not satisfied by NDI metric. As, on concatenating the decision (D) to both D_1 and D_2 , $\text{NDI}(D_1 \circ D)$ value may not be greater than or equal to $\text{NDI}(D_2 \circ D)$ value; because there exist some hierarchical information of D which is common to D_1 but not to D_2 , leading to suppressing $\text{NDI}(D_1 \circ D)$ value in comparison to value of $\text{NDI}(D_2 \circ D)$.

Axiom 6 (Archimedean): NDI metric not satisfied this axiom as decision concatenation; i.e., *DecisionsConcat* is idempotent (operation applied multiple times without changing the result). Thus, concatenation of D_1 with D_2 (i times) is equal to concatenation of D_1 with D_2 (one time).

NDI as an IC The properties of IC are described below and also explained in Table 2.

As, axiom 5 (Weak monotonicity) is not satisfied by NDI metric, then it does not satisfy condition $C1$ and it cannot satisfy $C2$ condition. Similarly, the third condition, i.e., $C3$ does not satisfy because axiom 5 was not fulfill, and if it does not satisfy $C3$, it cannot satisfy the $C4$ condition. Hence, NDI metric does not satisfy these IC properties.

NDI as an MRB Five properties of MES are described below and also explained in Table 2.

As NDI satisfy axiom 1, then the first (*MRB1*) and second (*MRB2*) property, i.e., completeness and transitivity are satisfied by NDI metric.

NDI satisfy property (*MRB3*), i.e., dominance as overall informations of D_2 are comprised in D_1 , this means that all hierarchical information of D_2 are in D_1 . So, we can say that, $\text{NDI}(D_1) \geq \text{NDI}(D_2)$.

MRB4, i.e., partial monotonicity is satisfied by the NDI metric because $D_1 \subset D_2$, then $\text{NDI}(D_1) > \text{NDI}(D_2)$. Now, assume D_3 having non-hierarchical information requirements with D_1 . Then, D_3 cannot have common information hierarchy with D_2 also. Hence, we can say that

$$\text{NDI}(D_1 \pm D_3) > \text{NDI}(D_2 \circ D_3).$$

At last, *MRB5* is satisfied by NDI metric, as if D has no information requirements hierarchy, then $\text{NDI}(D) = 0$, but never can be negative.

As NDI metric satisfy all the properties of MRB, hence, NDI metric is “above” the ordinal scale. Further, theoretical validation of NDI metric can also be implemented like NDG and NDGI metrics and said to be “above” ordinal scale.

5 Results and Discussions

Here, this section shows the results of formal validation of requirements traceability metrics (see Table 3) by applying properties of Zuse’s formal framework [16]. The whole theoretical validation of NGD and NDI metrics proves the Zuse’s framework properties in Sect. 4. Additionally, in the same way like NGD metric; validation of NID, NIG, and NGI metrics can be done and are said as “above ordinal scale”. Similarly, theoretical validation of NDI metric can also be implemented like NDG and NDGI metrics and is characterized as “above” ordinal scale. It is found that metric should “above” ordinal scale, because we cannot do much analysis with ordinal numbers.

Hence, according to the results of Zuse’s framework [16], it can be concluded that the requirements traceability metrics presented by [10] are accurately defined and validated theoretically (see Table 3). Therefore, traceability metrics proves the theoretical utility which devotes during the DW requirements model design, development, and improvement.

Table 3 Requirement’s traceability metrics results using Zuse’s formal framework [16]

Traceability metrics →	Coverage metrics					Depth and height coverage metrics	
	NGD	NDI	NDG	NID	NDGI	NIG	NGI
Zuse’s properties ↓							
MES—Axiom 1	✓	✓	✓	✓	✓	✓	✓
MES—Axiom 2	✓	✓	✓	✓	✓	✓	✓
MES—Axiom 3	✓	✓	✓	✓	✓	✓	✓
MES—Axiom 4	✓	✓	✓	✓	✓	✓	✓
MES—Axiom 5	✗	✗	✗	✗	✗	✗	✗
MES—Axiom 6	✗	✗	✗	✗	✗	✗	✗
Independence C1	✗	✗	✗	✗	✗	✗	✗
Independence C2	✗	✗	✗	✗	✗	✗	✗
Independence C3	✗	✗	✗	✗	✗	✗	✗
Independence C4	✗	✗	✗	✗	✗	✗	✗
MRB1	✓	✓	✓	✓	✓	✓	✓
MRB2	✓	✓	✓	✓	✓	✓	✓
MRB3	✓	✓	✓	✓	✓	✓	✓
MRB4	✓	✓	✓	✓	✓	✓	✓
MRB5	✓	✓	✓	✓	✓	✓	✓

6 Conclusion

Theoretical validation is implemented on requirements traceability metrics formed on DW railways requirements model (as shown in Sect. 3) in this paper. Here, the overall mathematical process is done to validate two metrics (NGD and NDI traceability) metrics theoretically that proves almost every property of MEB and MRB is satisfied from the outlook of Zuse's framework (see Sect. 4). In general, it is believed that MRB measures is important and is said as "above" ordinal scale. Based on results, it is concluded that every traceability metrics are "above" ordinal scale, and hence, are accurately defined and valid (see Sect. 5). Thus, traceability metrics can be utilized for estimating the quality of DW requirements model.

Also, various theoretical validation frameworks would be employed in the future to theoretically validate the requirements metrics for DW requirements model.

References

1. Inmon WH (2005) Building the data warehouse. Wiley
2. Rizzi S, Abelló A, Lechtenböcker J, Trujillo J (2006) Research in data warehouse modeling and design: dead or alive? In: 9th ACM international workshop on data warehousing and OLAP, pp 3–10
3. Serrano M, Trujillo J, Calero C, Piattini M (2007) Metrics for data warehouse conceptual models understandability. *Inf Softw Technol* 49(8):851–870
4. Kumar M, Gosain A, Singh Y (2012) Quality-oriented requirements engineering approach for data warehouse. *Int J Comput Syst Eng* 1(2):127–138
5. Kimball R, Ross M, Thornthwaite W, Mundy J, Becker B (2008) The data warehouse lifecycle toolkit. Wiley
6. Serrano M (2004) Definition of a set of metrics for assuring data warehouse quality. Univeristy of Castilla, La Mancha, Spain
7. Gaur H, Kumar M (2014) Assessing the understandability of a data warehouse logical model using a decision-tree approach. *ACM SIGSOFT Softw Eng Notes* 39(5):1–6
8. Gosain A, Singh J (2017) A formal approach for evaluating data warehouse metrics. In: International conference on smart trends for information technology and computer communications. Springer, Singapore, pp 236–243
9. Kumar M, Gosain A, Singh Y (2014) Empirical validation of structural metrics for predicting understandability of conceptual schemas for data warehouse. *Int J Syst Assur Eng Manage* 5(3):291–306
10. Kumar M, Gosain A, Singh Y (2013) On completeness and traceability metrics for data warehouse requirements engineering. *Int J Comput Syst Eng* 1(4):229–237
11. Kumar M (2015) Validation of data warehouse requirements-model traceability metrics using a formal framework. In: 2015 2nd international conference on computing for sustainable global development (INDIACom). IEEE, pp 216–221
12. Singh T, Kumar M (2020) Empirical validation of requirements traceability metrics for requirements model of data warehouse using SVM. In: 17th India council international conference (INDICON). IEEE, New Delhi, pp 1–5
13. Singh T, Kumar M (2021) Formally investigating traceability metrics of data warehouse requirements model using Briand's framework. In: 2021 5th international conference on intelligent computing and control systems (ICICCS). IEEE, pp 1203–1209

14. Singh T, Kumar M (2021) Investigating requirements completeness metrics for requirements schemas using requirements engineering approach of data warehouse: a formal and empirical validation. *Arab J Sci Eng*:1–20
15. Kumar M, Gosain A, Singh Y (2010) Stakeholders driven requirements engineering approach for data warehouse development. *J Inf Process Syst* 6(3):385–402
16. Zuse H (1998) A framework of software measurement. Walter de Gruyter

Corpus-Based Hashing Count Frequency Vectorization of Sentiment Analysis of Movie Reviews



M. Shyamala Devi, R. Aruna, Y. Lakshmi Akshitha, G. Chandana, G. Bhavisha, B. Lohitha, and M. Anusha

Abstract Sentiment analysis is textual context—specific extraction that identifies and extracts qualitative information from various information, assisting businesses in understanding the social emotion of their model, product, brand, or service while measuring online conversations. With the advancement in entertainment scenario, the analysis of customer real-time pulse while watching movie is essentially needed for rating the movie and to analyze the success of movie toward marketing industry. The machine learning can be used for predicting the sentiment of the movie review given by the customers. The movie review dataset from KAGGLE repository with ‘117211’ movie reviews is used for analyzing the type of movie based on the review comments. The movie review dataset is preprocessed by removing the stop words and unwanted tokens from the dataset. The tokens are extracted from the text using NGram method. The emotional labels are substituted for the tokens, and the machine is trained to identify the sentiment emotions during learning phase. The sentimental emotional labels are converted into features to form corpus text for predicting the emotions in the movie reviews. The corpus is splitted to form training and testing dataset and fitted to Count Vectorizer, Hash Vectorizer, TFID Transformer, and TFIDVectorizer to extract the important features from the text conversation. The extracted features from the movie reviews are subjected to all the classifiers to analyze the performance of the emotion prediction. The scripting is written in Python and implemented with Spyder in Anaconda Navigator IDE, and the experimental results show that the decision tree classifier with Hash Vectorizer is exhibiting 98.7% of accuracy toward predicting the sentiment from the movie reviews.

Keywords Machine learning · Tokens · Corpus · Vectorizer · TFID · Accuracy

M. Shyamala Devi (✉) · R. Aruna · Y. Lakshmi Akshitha · G. Chandana · G. Bhavisha · B. Lohitha · M. Anusha

Computer Science and Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, Tamil Nadu, India

e-mail: shyamaladevim@veltech.edu.in

1 Introduction

Thoughts and feelings have a significant impact on people's lives and contours who they are. They provide viewers with pertinent information about our current state and some well. Corporations and individuals must be capable of understanding the critical sentiment expressed by people in order to provide relevant recommendations to meet the individual needs of buyers in delivering efficient services to them. Sentiment analysis is a synergistic relationship of feelings also recognized as contributing factors and development that derives its essential premise from incorporating emotion-defined sophisticated technology to numerous different aspects of providing amazing findings. Emotion recognition will contribute significantly in artificial intelligence, advancement in the field of interacting components. It is a method for identifying and assessing the thoughts and feelings of conversation and messages, that also implies that people's choices feelings and emotions can be straightly obtained, and these methods can be used in a variety of cultural networking websites and business applications.

2 Literature Review

With the advancement of World Wide Web, knowledge representation and evaluation have grown in importance in terms of investment performance. It allows network providers continue providing customers with advanced features. Due to the ease with which information can be collected and the various services it provides, numerous studies are being conducted in the field of textual analysis and results. The above idea of sentimental analysis from message and discusses the various methodologies being used speech sentiment warning system [1]. The primary goal of sentiment classification for sales forecasting is to ascertain consumer attitudes of current alternatives. The purpose of this article should provide an overview and comprehensive evaluation of different text analytics tools and methodologies with opinion sentimental analysis and to describe the drawbacks of conventional research and career directions for emotion research methodologies with opinion sentimental analysis [2].

The user's perceptions could be substantiated with proven empirical evidence, and literally thousands of questionnaire items till after his or her context are properly identified. It appears to be trying to refine subconscious mind unless they are in a state of depression where participants were asked to respond by supplying supplements based on the knowledge accumulated regarding them. The advanced strategy will act as guidelines for the physician conducting interpretation and as favored empowering system [3]. Recognizing emotional responses at the smoother stage of imparted attitudes is essential for continuous improvement. The above critical inferences can indeed be procured completely across knowledge massive data emotion classification; as a result, conversation sentiment analysis using cognitive computing in large datasets from online platforms has considered as a leading range

of speech recognition research [4]. The vast variety of possible research is linear in nature, characterizing phrases as favorable or unfavorable without examining the thoughts and emotions that contribute to that characterization. Nevertheless, existing rules for analysis and interpretation, associated with the uncertainties and multi-dimensional components of the human views and experiences, have delivered such options available obsolete. Due to these limitations, scientific literature highlights specifically notifying thoughts and feelings instead of purely expressing feelings in a given document [5].

The ability to directly procure the majority's thoughts and feelings about social interactions, innovative operations, advertising strategies, and product preferences has attracted the attentions of both academicians and researchers, who are stimulated by the remarkable advantages and threats, and the finance sector, which is captivated by the achieving competitive advantages in product accounts and investment business forecasting. As a result, the fields of object recognition and text analytics have emerged, which use social communication, knowledge discovery, and multi-dimensional data information processing to derive strong thoughts from vast volumes of internet-based information [6]. This study investigates theoretical constructs that regard people's emotions as interpretations, visualizations, impression management consequences, social practices, biologically based printed circuit brands and commodities, and emotional understandings of feelings and emotions [7]. Munjal et al. [8] developed a framework sentiment analysis opinion mining [9, 10]. Sentiment emotion detection could help with emotion recognition, interpersonal behavior, information processing, Web-based learning, recommendation engines, and neuroimaging [11].

Subjective communications are categorized into two stages: one conveys offensive messages that could be about anything or very little and the other communicates with inherent notifications about the presenters itself. Both speaking and innovation have worked hard to fully comprehend the very first, concise and prompt path; however, the next one was quite well acknowledged. Acknowledging the feelings and emotions of the other group is one of the most important responsibilities associated with the second, underlying broadcaster. To finish the assignment, transmission processing and decision-making methods, as well as emotional and lingual assessments of thoughts and feelings, must be established [12]. A memetic approach is also used by Shekhawat et al. [13] for spam review detection with higher efficiency. A boolean object recognition prediction model was used to implement a framework for completely automated sentiment analysis. Because emotional reactions emerged to be lexicalized continuously and actively, researchers hypothesized that lexical and syntactic personal characteristics might be an adequate means of organizing the information. Simulated data oversampling was used to identify the best illustrations for everyone of the emotional reactions. To decrease language alteration, language processing modification was applied to input data [14].

3 Our Contributions

Figure 1 depicts the overall design of the work. This work includes the following contributions.

- The movie review dataset is normalized by discarding stop words and unnecessary tokens.
- The NGram technique is used to extract the tokens from the message. During the learning stage, sentimental labels are interchanged for tokens, and the machine is designed to detect sentiment emotional responses.
- The sensual sentimental tags are transformed into attributes, which are then used to generate corpus text for forecasting the emotional reactions in movie reviews.

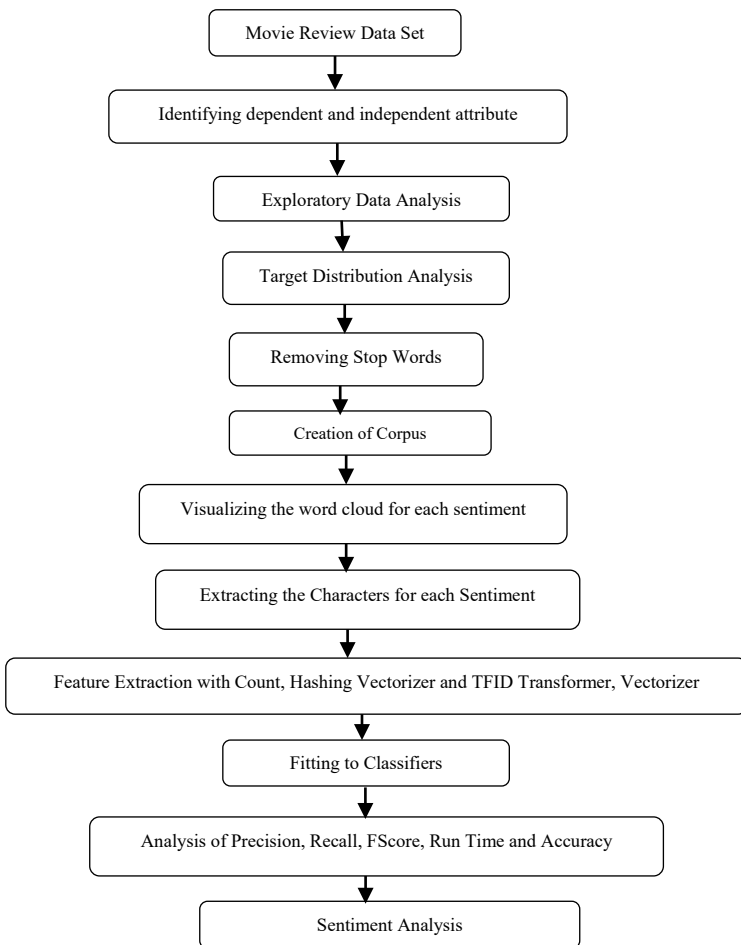


Fig. 1 Architecture system workflow

```

RangeIndex: 117211 entries, 0 to 117210
Data columns (total 2 columns):
#   Column      Non-Null Count  Dtype
---  ---        -
0   Phrase      117211 non-null object
1   Sentiment   117210 non-null float64
dtypes: float64(1), object(1)
memory usage: 1.8+ MB

```

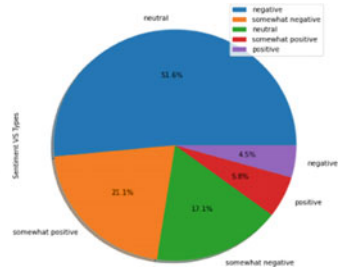


Fig. 2 Movie review dataset information and sentiment target distribution

- The corpus text is then divided to form training and testing dataset and fitted to Count Vectorizer, Hash Vectorizer, TFID Transformer, and TFIDVectorizer to take the key points from the text conversation and summarize them
- The features are extracted from the movie reviews are therefore successfully applied to all of the classifiers in order to assess the performance of the sentiment forecasting.

4 Implementation Setup

The movie review dataset from the KAGGLE repository with 117,211 incidences is then applied to data preprocessing. Figure 2 depicts the data source details as well as the emotion dissemination.

5 Results and Discussion

The movie review dataset is normalized by expelling stop words and unnecessary tokens. The target words for each sentence are discovered and evaluated as a word cloud, as shown in Fig. 3.

The corpus is then divided to create training and testing dataset and then processed with Count Vectorizer, Hash Vectorizer, TFID Transformer, TFIDVectorizer to expel the significant attributes from the message chat discussion. The removed attributes from the movie reviews are then processed with the classifiers to investigate the emotion detection and are shown in Fig. 5 and Tables 1, 2, 3, and 4.

Figure 4 depicts the mean length of the desired emotion tokens derived from movie review dataset information.



Fig. 3 Sentiment target word cloud

Table 1 Metrics coefficient analysis with Count Vectorizer

Classification	Prec.	Recall	FSc0.	Accu.	RTime
LReg	0.9433	0.9432	0.9432	0.9533	0.31
KNearest	0.9423	0.9443	0.9443	0.9435	0.17
KernelSVM	0.9477	0.9337	0.9446	0.9567	1.56
GNB	0.9787	0.9787	0.9777	0.9787	0.14
Decision	0.9667	0.9667	0.9667	0.9667	0.16
Extra	0.9117	0.9117	0.9117	0.9117	0.18
Rforest	0.9273	0.9173	0.9273	0.9273	0.20
Rid	0.9577	0.9557	0.9557	0.9557	0.18
RidCV	0.9227	0.9227	0.9227	0.9227	0.24
SGDclass	0.9333	0.9223	0.9223	0.9223	0.23
Passive	0.9331	0.9011	0.9011	0.9011	0.17
Bagging	0.9212	0.9222	0.9222	0.9222	0.18

6 Conclusion

This study investigated the effectiveness of emotion classification inside the movie review data source. It aims at evaluating classification results in comparison with multiple different feature extraction techniques. The emotion of film ratings is analyzed by retrieving phrases from the corpus using multiple feature extraction

Table 2 Metrics coefficient analysis with Hash Vectorizer

Classification	Prec.	Recall	FSc.	Accu.	RTime
LReg	0.9117	0.9367	0.9167	0.9267	0.21
KNearest	0.9433	0.9432	0.9432	0.9533	0.17
KernelSVM	0.9423	0.9443	0.9443	0.9435	1.16
GNB	0.9477	0.9337	0.9446	0.9567	0.14
Decision	0.9873	0.9773	0.9773	0.9873	0.16
Extra	0.9117	0.9117	0.9117	0.9117	0.08
Rforest	0.9273	0.9173	0.9273	0.9273	0.12
Rid	0.9021	0.9021	0.9121	0.9121	0.08
RidCV	0.9233	0.9233	0.9233	0.9232	0.06
SGDclass	0.9133	0.9133	0.9213	0.9133	0.11
Passive	0.9021	0.9075	0.9075	0.9075	0.09
Bagging	0.9233	0.9233	0.9233	0.9232	0.14

Table 3 Metrics coefficient analysis with TFIDVectorizer

Classification	Prec.	Recall	FSc.	Accu.	RTime
LReg	0.9323	0.9343	0.9333	0.9335	0.30
KNearest	0.9477	0.9437	0.9446	0.9567	0.27
KernelSVM	0.9787	0.9787	0.9777	0.9787	1.16
GNB	0.9467	0.9447	0.9567	0.9567	0.23
Decision	0.9217	0.9127	0.9117	0.9117	0.25
Extra	0.9163	0.9166	0.9273	0.9273	0.19
Rforest	0.9211	0.9111	0.9211	0.9221	0.33
Rid	0.9577	0.9557	0.9557	0.9557	0.29
RidCV	0.9227	0.9227	0.9227	0.9227	0.18
SGDclass	0.9333	0.9223	0.9223	0.9223	0.22
Passive	0.9331	0.9011	0.9011	0.9011	0.19
Bagging	0.9212	0.9222	0.9222	0.9222	0.23

methodologies such as Count Vectorizer, Hash Vectorizer, TFID Transformer, and TFIDVectorizer. According to the experimental results, the decision tree classifier with Hash Vectorizer predicts emotional states from film ratings with 98.7% accuracy.

Table 4 Metrics coefficient analysis with TFID Transformer

Classification	Prec.	Recall	FScO.	Accu.	RTime
LReg	0.9217	0.9127	0.9117	0.9117	0.18
KNearest	0.9163	0.9166	0.9273	0.9273	0.07
KernelSVM	0.9211	0.9111	0.9211	0.9221	1.02
GNB	0.9007	0.9007	0.9117	0.9017	0.12
Decision	0.9127	0.9127	0.9137	0.9127	0.18
Extra	0.9067	0.9063	0.9066	0.9067	0.08
Rforest	0.9233	0.9232	0.9232	0.9333	0.15
Rid	0.9323	0.9343	0.9333	0.9335	0.06
RidCV	0.9477	0.9437	0.9446	0.9567	0.12
SGDclass	0.9787	0.9787	0.9777	0.9787	0.13
Passive	0.9021	0.9075	0.9075	0.9075	0.09
Bagging	0.9233	0.9233	0.9233	0.9232	0.14

Average length of negative messages:

65.20041635124906 characters

Average length of somewhat negative messages:

49.850774612693655 characters

Average length of neutral messages:

28.041969383533306 characters

Average length of somewhat positive messages:

47.07720975451673 characters

Average length of positive messages:

59.98280675973549 characters

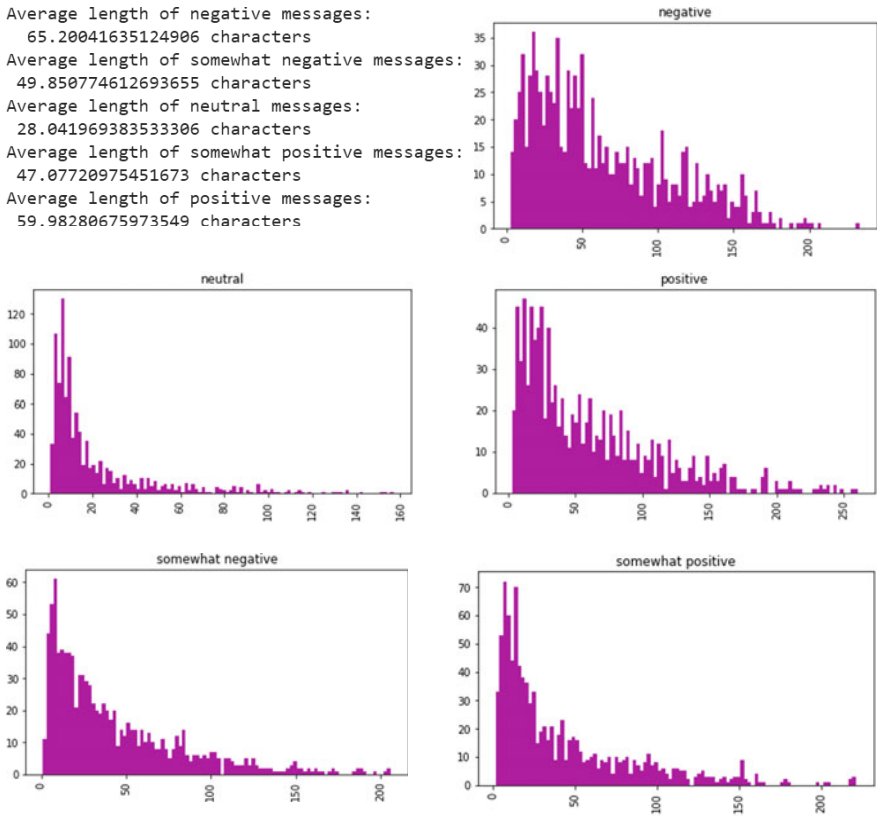


Fig. 4 Average length of sentiment target messages with analysis of negative, neutral, positive, somewhat negative, and somewhat positive sentiments

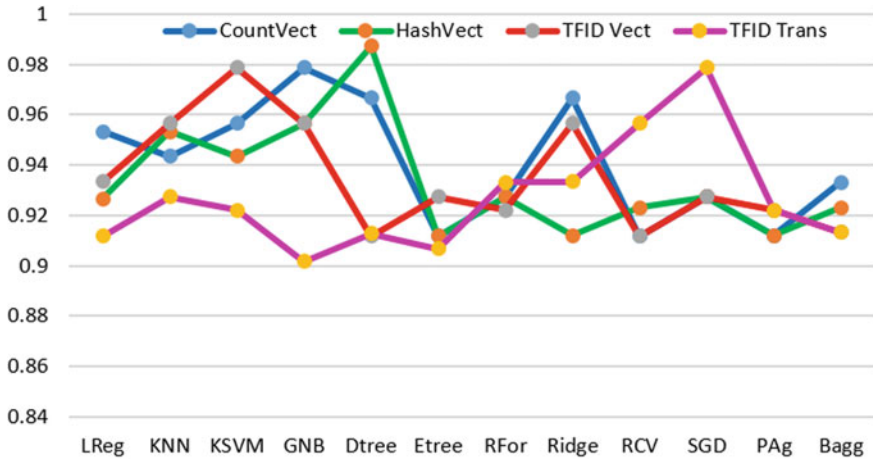


Fig. 5 Accuracy comparison of classifier with Count, Hash, TFIDVectorizer, and Transformer

References

1. Acheampong FA, Wenyu C, Nunoo-Mensah H (2020) Text-based emotion detection: advances, challenges, and opportunities. *Rev Rep*
2. Ahire V, Borse S (2022) Emotion detection from social media using machine learning techniques: a survey. In: *Applied information processing systems. Advances in intelligent systems and computing*, vol 1354. Springer
3. Sekhar C, Rao MS, Nayani ASK, Bhattacharyya D (2021) Emotion recognition through human conversation using machine learning techniques. In: *Machine intelligence and soft computing. Advances in intelligent systems and computing*, vol 1280. Springer
4. Kusal S, Patil S, Kotecha K, Aluvalu R, Varadarajan V (2021) AI based emotion detection for textual big data: techniques and contribution. *Big Data Cogn Comput* 5(3)
5. Krommyda M, Rigos A, Bouklas K, Amditis A (2020) Emotion detection in Twitter posts: a rule-based algorithm for annotated data acquisition. In: *Proceedings of international conference on computational science and computational intelligence*, pp 257–262
6. Cambria E (2016) Affective computing and sentiment analysis. *IEEE Intell Syst* 31(2):102–107
7. Calvo R, Mello SD (2010) Affect detection: an interdisciplinary review of models methods and their applications. *IEEE Trans Affect Comput* 1(1):18–37
8. Munjal P, Narula M, Kumar S, Banati H (2018) Twitter sentiments based suggestive framework to predict trends. *J Stat Manag Syst* 21(4):685–693
9. Munjal P, Kumar L, Kumar S, Banati H (2019) Evidence of Ostwald Ripening in opinion driven dynamics of mutually competitive social networks. *Phys A* 522:182–194
10. Munjal P, Kumar S, Kumar L, Banati A (2017) Opinion dynamics through natural phenomenon of grain growth and population migration. In: *Hybrid intelligence for social networks*. Springer, Cham, pp 161–175
11. Schuller B (2011) Recognising realistic emotions and affect in speech: state of the art and lessons learnt from the first challenge. *Speech Commun* 53(9/10):1062–1087
12. Cowie R, Douglas-Cowie E, Tsapatsoulis N, Votsis G, Kollias S (2001) Emotion recognition in human-computer interaction. *IEEE Sig Process Mag* 18:32–80

13. Shekhawat SS, Sharma H, Kumar S (2021) Memetic spider monkey optimization for spam review detection problem. *Big Data*. <https://doi.org/10.1089/big.2020.0188>
14. Cambria E, Olsher D, Rajagopal D (2014) SenticNet 3: a common and common-sense knowledge base for cognition-driven sentiment analysis. In: *Proceedings of 28th AAAI conference on artificial intelligence*, pp 1515–1521

Anime Scene Generator from Real-World Scenario Using Generative Adversarial Networks



Le Xuan Huy, Bui Thi Bich Ngoc, and Phan Duy Hung

Abstract This study illustrates a method for image cartoonization and style transfer, which entails converting a real-world image or video into an aesthetic, anime-style frame. By noticing the animation painting behaviors, this work proposes to independently distinguish three feature maps from pictures: the surface representation that contains smooth color shading characteristic of animation, the construction depiction that resembles flattened global content and clear boundaries in a typical anime frame, and the texture representation that reflects high-frequency surface, forms, and details in animated pictures. All the extracted information will be fed into the generator with the help of a VGG-based discriminator to learn how to cartoonize a real-world photo. This technique's training objectives are based on each extracted feature map independently, making the model controllable and adjustable. Experiments demonstrate that this method can generate high-quality animation graphics from real-world photos and outperforms many existing methods.

Keywords Style transfer · Image translation · Generative adversarial networks

1 Introduction

Humanity had learned to recreate daily natural sceneries on rocks and wood since the start of time. Humans had come from simple, straight-line sketches to realistic pictures and even cartoon and artistic drawings. This paper focuses on anime/cartoon style drawings and how to generate them from the input: real-life footage. One primary approach when it comes to domain transfer problems is using generative adversarial networks [1].

L. X. Huy · B. T. B. Ngoc · P. D. Hung (✉)
Computer Science Department, FPT University, Hanoi, Vietnam
e-mail: hungpd2@fe.edu.vn

L. X. Huy
e-mail: huylxhe140555@fpt.edu.vn

B. T. B. Ngoc
e-mail: ngocbtbhe140996@fpt.edu.vn

The rise of a new study in the last decades—the generative adversarial networks (GANs)—has indeed shaken the world of computer vision and neural style transfer. A wide range of applications is found by learning to translate data from its original domain to another, such as style transferring, image colorization, image restoration, and superresolution.

These days, GAN has shown its prominence with practical solutions in domain transfer and image generation problems. AnimeGAN image generation uses the deep convolutional GAN model trained on a 143,000 anime character faces dataset to generate new anime faces [2]. CartoonGAN proposes image translation with unpaired training data, significantly reducing the effort needed for data preprocessing [3]. Nonetheless, its black-box model is the enemy of generalization. CycleGAN introduces the cycle-consistent adversarial networks with cycle-consistent loss and full-cycle transform [4]. Their methodologies are expanded and refined in many subsequent studies. Meanwhile, Comixify works with videos and tries to convert them into comics [5]. They extract keyframes from the input videos, convert them into comics, and intend to incorporate speech recognition in the future.

The main focus of this research is on converting real-life footage into anime/cartoon style. In addition, it also involves a new algorithm and manually collected dataset to improve the generated animation.

2 Related Works

Image smoothing, in certain cases, is used as the first step of input image preprocessing. He et al. use a guided filter-modified bilateral filtering with better performance on edges [6]. Farbman et al. discard the old ways by utilizing a weighted least squares optimization framework to tighten up the edge-preserving operator [7]. Meanwhile, this study adjusts a trainable-guided filter to extract the top smooth filter [8] and a graph-based image representation to develop an efficient segmentation algorithm for extracting structure representation [9].

Generative adversarial networks [1] is a data generation method introducing adversarial training between the G —Generator and the D —Discriminator to achieve impressive results. Multiple variants of GANs have been introduced to solve specific problems in image processing, such as conditional GANs [10] and progressive-growing GANs [11]. The chosen method here is GANs because it is indeed powerful for image generation, especially domain and style translation tasks.

Image-to-image translation focuses on translating images from a source domain to another target domain. By using unpaired data, this approach becomes a general-purpose solution for many image processing tasks. Zhu et al. introduced CycleGAN, which uses bidirectional models to perform the transformation in various styles [4]. UNIT by Liu et al. creates a shared latent space by mapping source and target domain images to learn the joint distribution between them in an unsupervised manner [12].

All the above studies prove that interdomain translation is always an attractive field. Many interesting and effective methods were proposed to solve various problems in style transferring and image translation. However, some problems still require answers, such as unclear results caused by outliers [2], insufficient data [4], or poor style generalization caused by partial images segmentation of specific types.

An independent anime dataset was collected to conduct this study. New algorithms to calculate texture loss are introduced to reduce noises and unwanted edges that many methods have faced. The obtained results demonstrate that this model is lighter and can perform style transferring faster and easier than other models.

3 Methodology

This GAN framework contains two types of CNNs. The first one is a generator G , which learns to produce fake and indistinguishable input compared to the real ones through training. The discriminator D classifies whether the image is from the real target manifold or synthetic. This paper designs the generator and discriminator networks in a way to suit the particularity of cartoon images and can be easily fine-tuned and modified. The generator is U-Net based, which is capable of generating cartoon images in a short amount of time. After going through the generator, images are decomposed into the surface representation, the structure representation, and the texture representations. Three independent losses are proposed to extract information, as shown in Fig. 1.

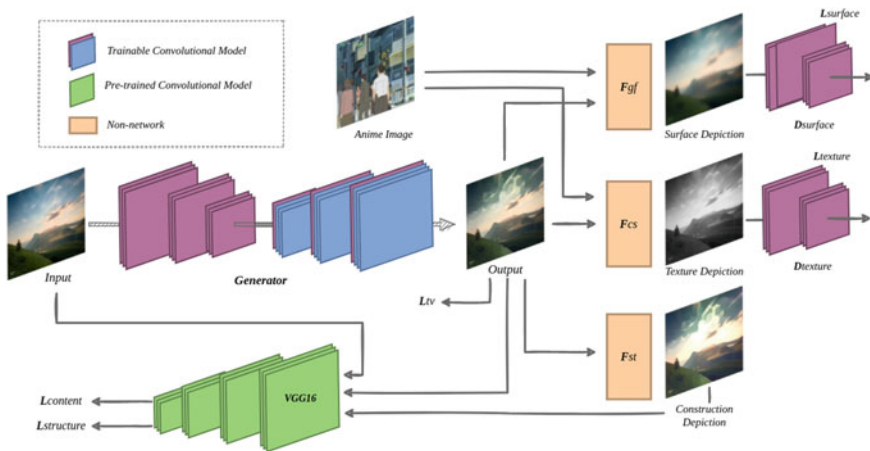


Fig. 1 Model architecture

Fig. 2 Hierarchical grouping algorithm

Input: (color) image
Output: Set of object location hypotheses L
 Obtain initial regions $R = \{r_1, \dots, r_n\}$
 Initialize similarity set $S = \emptyset$
foreach *Neighboring region pair* (r_i, r_j) **do:**
 Calculate similarity $s(r_i, r_j)$
 $S = S \cup s(r_i, r_j)$
While $S \neq n$ **do:**
 Get highest similarity $s(r_i, r_j) = \max(S)$
 Merge corresponding regions $r_t = r_i \cup r_j$
 Remove similarities regarding r_i : $S = S \setminus s(r_i, r_x)$
 Remove similarities regarding r_j : $S = S \setminus s(r_j, r_x)$
 Calculate similarity set S_t between r_t and its neighbors
 $S = S \cup S_t$
 $R = R \cup r_t$
 Extract object location boxes L from all regions in R

3.1 Structure Loss

The structure loss aims to imitate the animated style of clear edge, high-level simplification and abstraction, and sparse color blocks. After experiencing different solutions, our work finally achieved high-performance results with Felzenszwalb segmentation and hierarchical grouping as shown in Fig. 2 [9, 13]. In the first step, the Felzenszwalb algorithm is used to segment the image into separate regions. After that, a greedy algorithm is used to iteratively group regions together. Then, the similarities between all neighboring regions are calculated. The two most identical regions are grouped, from that new similarities will be computed between the newly achieved region and its neighbors. The process of grouping the most similar regions is repeated until the remainings are equal to the given segmentation number.

In order to enforce spatial constraints between the result and the extracted structure representation, a pre-trained *VGG16* feature extractor is utilized as a structure discriminator. Let F_{sr} be the extracted structure representation, and the final loss is formulated as:

$$L_{\text{structure}} = VGG(G(I_x)) - VGG(F_{sr}(G(I_x))) \quad (1)$$

3.2 Surface Loss

The surface loss will try to force the model to learn the animation painting style where artists usually draw the draft first with coarse brushes and have rather smooth

surfaces. In order to do that, this model uses a differentiable-guided filter from [14] for edge-preserving filtering. The filter denoted as F_{gf} will take an image as input and return a smooth, blur version of it, $F_{\text{gf}}(I_x, I_y)$ with texture and details removed, where I_x is the input photo and I_y represents the reference cartoon images. A simple discriminator D_s is used to decide if the generated output having the same surface as the animated picture to help the generator G . The formulation can be written as follow:

$$L_{\text{surface}} = \log D_s(F_{\text{gf}}(I_y, I_y)) + \log(1 - D_s(F_{\text{gf}}(G(I_x), G(I_x)))) \quad (2)$$

3.3 Texture Loss

Along with the color and luminance factor, which have been focused on with the two losses above, cartoon styles also have unique characteristics with high-level simplification, high-frequency features that can be treated as key objectives to make it easy to distinguish between cartoon images and real-world photos. Due to this, a simple random color shift algorithm F_{cs} is used to convert the image to a grayscale feature map that still contains information about all the high-frequency textures.

$$I_{\text{grayscale}} = \frac{\beta_1 \cdot I_r + \beta_2 \cdot I_b + \beta_3 \cdot I_g}{\beta_1 + \beta_2 + \beta_3} \quad (3)$$

where $\beta_1, \beta_2, \beta_3$ is trainable parameters. A simple discriminator D_t also separates the grayscale feature map extracted from the generated and cartoon images.

$$L_{\text{texture}} = \log D_t(F_{\text{cs}}(I_y)) + \log(1 - D_t(F_{\text{cs}}(G(I_x)))) \quad (4)$$

3.4 Total-Variant Loss, Superpixel Loss, and Full Model

In order to make the content of the generated photo to be of high quality, the total-variation loss L_{tv} is used to impose spatial smoothness on generated images and also reduces high-frequency noises such as salt-and-pepper noise. For H, W , and C represent spatial dimensions of images, it looks like:

$$L_{\text{tv}} = \frac{1}{H \cdot W \cdot C} \nabla_x(G(I_x)) + \nabla_y(G(I_x)) \quad (5)$$

Finally, a superpixel loss L_{sp} is proposed to maintain important content from the input photo. It is used to ensure that the cartoonized results and input photos are

semantically invariant. A pre-trained *VGG16* model works to calculate it:

$$L_{sp} = VGG(G(I_x)) - VGG(I_x) \quad (6)$$

With all of the losses mentioned above, the final loss function can be written as:

$$L_{generator} = \beta_1 \cdot L_{tv} + \beta_2 \cdot L_{surface} + \beta_3 \cdot L_{structure} + \beta_4 \cdot L_{texture} + \beta_5 \cdot L_{sp} \quad (7)$$

where the parameters $\beta_1, \beta_2, \beta_3 \dots$ can be changed for separate uses.

4 Experiments and Results

4.1 Implementation

This GAN model is implemented in TensorFlow [15]. The trained models in experiments are available to facilitate the evaluation of future methods. All experiments were performed on an NVIDIA 1060Ti GPU. The model uses Adam's algorithm [16] with a learning rate of 1.5×10^{-4} , trained with batch size 16 for 20,000 iterations.

4.2 Dataset

The training dataset contains four folders, real and cartoon images for scenery and human face. The animation data is collected manually by authors, cropped from real anime videos, primarily from Shinkai Makoto's films. For the cartoon data, the project uses 10,000 for scenery and 5000 for human faces. In terms of the real-world data, there are 10,000 scenery images crawled from the Internet and 5000 human faces from the FFHQ dataset [17]. Furthermore, 1000 real-life images and 1000 cartoon images were collected from the Internet for the validation dataset.

4.3 Time Performance and Model Size

This method has a much lower number of parameters and running time compared to some related works as given in Table 1. On the Nvidia GTX 1060Ti GPU, it reaches the time of 15 ms per image, which is faster than other related works and can be capable of real-time high-resolution video processing tasks.

Table 1 Parameters and time performance comparison

Methods	AnimeGAN [18]	CartoonGAN [3]	CycleGAN [4]	Ours
HR, GPU (ms)	45.53	148.02	106.82	15.23
Parameter (m)	3.96	11.38	11.13	1.48

4.4 Evaluation Metrics

For qualitative evaluation, this paper will present results for different objects compared to the results from other related works. Moreover, Frechet inception distance (FID) is proposed for quantitative evaluation by comparing the generated images with the target images [19].

4.5 Result Demo

As shown out above in Fig. 3, this model could indeed be generalized to diverse use cases. It can be applied in different real-world scenery, including human faces, animals, foods, and city street images.



Fig. 3 Models' result on multiple categories

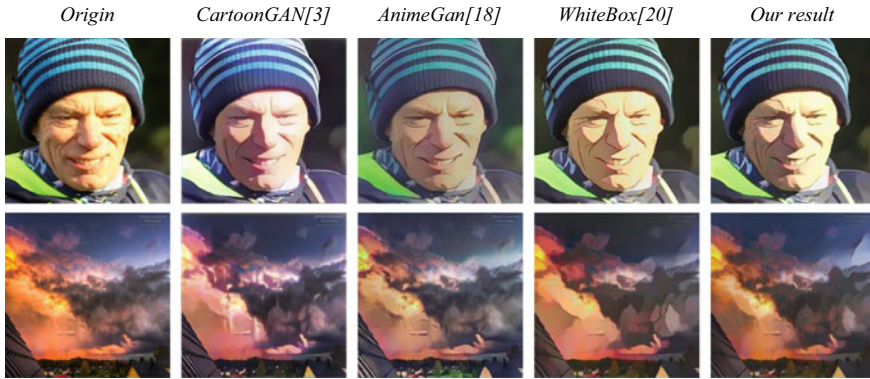


Fig. 4 Original images with their results using a different model

4.6 Qualitative Comparison

Comparison between this method and some related works is shown in Fig. 4. CartoonGAN generates quality results with good texture and clear edges but lacks abstraction and tends to distort colors. This model, on the other hand, prevents improper color modifications. AnimeGAN generates darkened images and cannot really show out the cartoon/anime style. The white-box model has clear boundaries but focuses a little too much on smoothing the picture with color blocks and enhancing the edge, leading to the lack of small details from the original image and creating noises. Finally, this model is not perfect, but it gives a balanced result in color, texture and still generates cartoon feel-like images. To conclude, it outperforms previous methods in generating images with harmonious color, clean edges, fine details, and less noise.

4.7 Quantitative Comparison

FID is widely used to evaluate the quality of synthesized images quantitatively. This work also proposes Heusel’s method to calculate the distance between two image distributions [19], namely the generated and real anime images. FID results used to calculate this study’s performance and performance of related works are given in Table 2. This method generates images with the smallest FID score, proving that it has the most cartoon/anime style result, outperforming the others.

Table 2 Performance evaluation based on FID

Method	Real photo	CartoonGAN [3]	AnimeGAN [18]	whitebox [20]	Ours
FID to cartoon	160	125	130	118	110

5 Conclusion and Future Works

This study presents a lightweight and controllable approach for image cartoonization by converting actual footage into animation. Frechet inception distance is used as its evaluation metric to compare the generated images to target images and has received promising results. The method pays close attention to the animation painting process, encouraging us to distinguish three feature maps from pictures for independent handling. The generator takes in the surface representation, construction depiction, texture representation, and a VGG-based discriminator to learn the cartoonization process. Unpaired datasets are prepared for training the model, allowing it to be fine-tuned more effortlessly. Experimental results show that this model can generate better results by surpassing performance compared to many state-of-the-art models.

In the future development, we would like to expand the application of this technique to videos in order to create seamless, anime-style cuts. Details on the portrait and facial expression are also considered for improvement, so that the character's emotion and sentiment would be more well described in further study.

References

1. Goodfellow IJ, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial networks. [arXiv:1406.2661](https://arxiv.org/abs/1406.2661)
2. Lei J (2017) AnimeGAN. <https://github.com/jayleicn/animeGAN>
3. Chen Y, Lai Y, Liu Y (2018) CartoonGAN: generative adversarial networks for photo cartoonization. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 9465–9474
4. Zhu J, Park T, Isola P, Efros AA (2017) Unpaired image-to-image translation using cycle-consistent adversarial networks. [arXiv:1703.10593](https://arxiv.org/abs/1703.10593)
5. Peřsko M, Svystun A, Andruszkiewicz P, Rokita P, Trzciński T (2018) Comixify: transform video into a comic. [arXiv:1812.03473](https://arxiv.org/abs/1812.03473)
6. He K, Sun J, Tang X (2013) Guided image filtering. *IEEE Trans Pattern Anal Mach Intell* 35:1397–1409. <https://doi.org/10.1109/TPAMI.2012.213>
7. Farbman Z, Fattal R, Lischinski D, Szeliski R (2008) Edge-preserving decompositions for multi-scale tone and detail manipulation. *ACM Trans Graph* 27. <https://doi.org/10.1145/1360612.1360666>
8. Wu H, Zheng S, Zhang J, Huang K (2018) Fast end-to-end trainable guided filter. [arXiv:1803.05619](https://arxiv.org/abs/1803.05619)
9. Felzenszwalb PF, Huttenlocher DP (2004) Efficient graph-based image segmentation. *Int J Comput Vision* 59:167–181. <https://doi.org/10.1023/B:VISI.0000022288.19776.77>
10. Mirza M, Osindero S (2014) Conditional generative adversarial nets. [arXiv:1411.1784](https://arxiv.org/abs/1411.1784)
11. Karras T, Aila T, Laine S, Lehtinen J (2017) Progressive growing of GANs for improved quality, stability, and variation. [arXiv:1710.10196](https://arxiv.org/abs/1710.10196)
12. Liu M, Breuel T, Kautz J (2017) Unsupervised image-to-image translation networks. [arXiv:1703.00848](https://arxiv.org/abs/1703.00848)
13. van de Sande KEA, Uijlings JRR, Gevers T, Smeulders AWM (2013) Selective search for object recognition. *Int J Comput Vision* 104(2):154–171
14. JoyceMar GJ, Begum A (2015) Rijuvana: guided filter smoothing for third order edge mask. *Int J Comp Appl* 120:36–42. <https://doi.org/10.5120/21403-4424>

15. Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, Devin M, Ghemawat S, Irving G, Isard M, Kudlur M, Levenberg J, Monga R, Moore S, Murray DG, Steiner B, Tucker P, Vasudevan V, Warden P, Wicke M, Yu Y, Zheng X (2016) TensorFlow: a system for large-scale machine learning. [arXiv:1605.08695](https://arxiv.org/abs/1605.08695)
16. Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. [arXiv:1412.6980](https://arxiv.org/abs/1412.6980)
17. Karras T, Laine S, Aila T (2018) A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4401–4410
18. Chen J, Liu G, Chen X (2020) AnimeGAN: a novel lightweight GAN for photo animation. In: Li K, Li W, Wang H, Liu Y (eds) Artificial intelligence algorithms and applications, ISICA 2019. Communications in computer and information science, vol 1205. Springer, Singapore
19. Heusel M, Ramsauer H, Unterthiner T, Nessler B, Hochreiter S (2017) GANs trained by a two time-scale update rule converge to a local nash equilibrium. In: Advances in neural information processing systems, pp 6626–6637
20. Wang X, Yu J (2020) Learning to cartoonize using white-box cartoon representations. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)

Employing AI for Development of a Smart Entry Log System at Entry Gates



Anusha Gadgil, Arjun Thakur, Mihir Gohad, Rahee Walambe, and Ketan Kotecha

Abstract Artificial intelligence is said to perform all our manual tasks with higher efficiency. But it may have a greater role in opening new doors to new methods of inventions that can completely change the innovation process of research and development. Smart entry logs aim to automate and simplify the process of making entries of the vehicles that come and go in a certain area using pre-existing CCTV feeds. This, in return, helps in reducing human effort and head toward a bright future with the use of artificial intelligence. In this work, we have developed an end-to-end smart entry log system to systematically save various data features of individuals passing through an entry point being monitored. We employed Opencv2 to detect vehicle number plates and got an accuracy of 90% on test data, tesseract to read the text on the plates with an accuracy of 85% on unseen data and SSD to identify if a rider is wearing a helmet or not with 87.7% model accuracy. Besides this, we used convolutional neural network for front- and rear-view detection of car and vehicle classification into multiple categories with 73.7 and 60% model accuracy, respectively, on Python version 3.6. We also saved the color of the number plate, so it is easy to identify the type of vehicle from the video recording. All this is stored in an excel sheet with timestamps which can be used later for records.

Keywords Helmet detection · License plate detection · Optical character recognition · Pose estimation · Vehicle classification

A. Gadgil · A. Thakur · R. Walambe (✉)

Department of ENTIC, Symbiosis International Deemed University, Pune, Maharashtra, India
e-mail: rahee.walambe@sitpune.edu.in

M. Gohad

Department of Mechanical, Symbiosis International Deemed University, Pune, Maharashtra, India
e-mail: gohad.mihir.btech2018@sitpune.edu.in

K. Kotecha

Faculty of Engineering, Symbiosis International Deemed University, Pune, Maharashtra, India

1 Introduction

The rapid improvement in the field of artificial intelligence [13] has helped in reducing human efforts. The “smart entry logs” system aims at creating a log using a system that will help users to record the entry and exit of vehicles at the gate of institutions with the time details by using object detection [1] algorithms. The system automates the work, and we have comprehensive features with very good accuracy.

We have built a six layer end-to-end security system for all two- and four-wheelers entering and exiting any gated area. The system works at multiple angles and elevations, providing details of incoming traffic like vehicle license plate recognition with its color detection, vehicle classification, pose estimation, and rider helmet detection along with timestamp of all vehicles. Our system works on videos recorded at entry points with high accuracy, thereby provide additional security layer. All the details are stored in Excel Sheets so that we can keep a track record. All the six models have been made to work in four parts on video feeds with a very high accuracy, and the object detection output is directly saved in these Excel Sheets. The license plate detection, OCR (tesseract), and registration color plate part in a Excel Sheet, helmet detection in the second Excel Sheet, vehicle classification into six classes in the third Excel Sheet and lastly pose estimation of front and rear side in the fourth Excel Sheet. Each entry in the Excel Sheets is accompanied with a timestamp. These Excel Sheets store the details of vehicles in real time from video feeds which can be accessed at any given time.

It has widespread applications in traffic monitoring, law enforcement, and smart parking and hence, is an integral part of intelligent transportation systems [18]. We believe this system will be highly useful, as noting down vehicle details by temporarily stopping the car that is entering wastes the driver’s time using advance artificial intelligence techniques. Our system can be used in a variety of places ranging from hospitals, schools, colleges, housing societies, public buildings, etc., with very improved accuracy and with many features. It reduces manpower, i.e., the work of a security guard. In times of a pandemic, this project will be very useful as the driver will not have to roll down his/her vehicle window and communicate with a guard. It ensures the safety of the security guard as well as the driver during a pandemic as it helps to maintain a safe distance. The basic idea is to use live CCTV [15] footage and detect various details of the incoming vehicles using OpenCV library [19], tesseract engine [16], single shot detection [10], CNN algorithm [12], on Python [11] platforms like Google Colaboratory.

In this paper, we have used different machine learning concepts to construct our smart and innovative system that works 24/7 (without human intervention) for logging details of all types of vehicles entering the institution where this system can be installed.

1.1 Related Work

Two specific techniques for smart entry logs are reported in literature.

1. RFID: Basically, radio-frequency identification technology is a technology where data gets encoded digitally as RFID tags and radio waves are used by a reader to detect the data [8]. Figure 1 shows the flowchart for the wireless entry system using RFID Tags.
2. ANPR: Automatic number plate recognition (ANPR) can read the number plate present on vehicles capturing images at fast speed in the absence of human input. By importing “ALPR” in Python, we can do real-time license plate recognition with “openALPR” using a video file as input. The steps involved in automatic number plate detection are shown in Fig. 2.

However, these approaches have certain limitations. These technologies are only capable of noting down the registration plates, and in case of a broken number plate or dirty number plate, fake number plate ANPR will not be useful. Existing technologies use RFID tags to store the details of all kinds of vehicles, these tags are expensive, and we must go through the hassle of distributing them. To that end, in this work, we have presented an end-to-end framework based on AI-based methods. Our model

Fig. 1 RFID gate entry system [3]

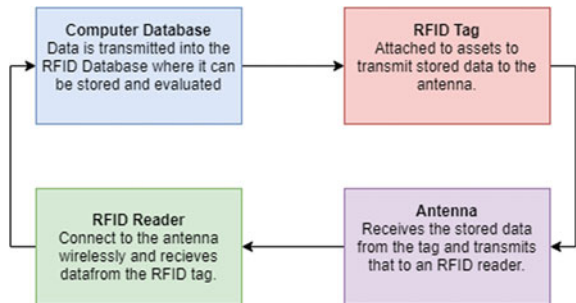
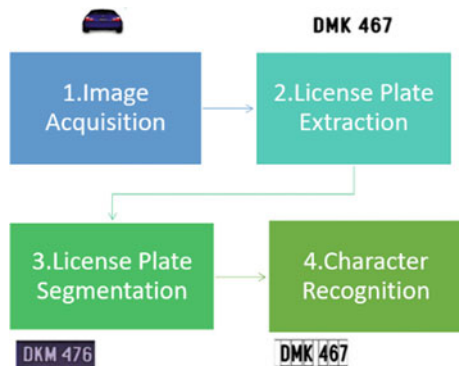


Fig. 2 License plate recognition [19]



records many details of the vehicles entering and leaving the institutions while other models have only focused on storing the license details. Our system will produce a complete log of incoming traffic not just note down registration plates as the existing technologies do, our system includes other details as well so we can still manage to note down significant details of incoming vehicles which will help users to enhance security. This model provides us with many details of all type of vehicles with very high accuracy. Our model gives us details as precise as the color of the license plate to identify the type of vehicles. This model can successfully differentiate between head and helmet for helmet detection [14] task, count bicycle and scooty helmet as a helmet, avoid misclassification of scooty and motorbike. We have manually collected 1000 images of auto-rickshaw (three-wheeler) which can be easily detected under vehicle classification. Our project is entirely AI-based and cost-effective as we will require minimal hardware.

This framework has multiple features, and the following parameters are recorded in an Excel Sheet:

1. Vehicle license number (text + color)
2. Vehicle type (motorbike/scooty/car/truck/auto/bus)
3. Rider helmet detection for two-wheeler (“Yes” for helmet detected, “No” for no helmet)
4. Four-wheeler front- and Rear-view pose estimations
5. Timestamp (for recording the time of entry and exit in the log).

To implement the above-specified features, we have experimented with multiple AI-based approaches. These are discussed in section system design.

The paper is organized as:—in Sect. 2, we first discuss the process of collection of the dataset, then we show the algorithms we have written section-wise mathematical formulation. Following this, in Sect. 3, we have described the project pipeline, preprocessing steps [9] for the models we used. In Sect. 4, we displayed the results and accuracy chart. In the end, we have the results followed by the conclusion in Sect. 5.

2 Methods

2.1 Datasets

We created a training dataset containing 800 images of vehicles that we collected from our locality for license plate detection. For each image, we marked the position of the bounding box [17] on each license plate training image, labeled them and saved the contents in a .xml file [2]. We used screen-o-matic software to record 30 YouTube videos of 1 min each and saved them so that we could test our XML file on them. Figure 3 shows locally collected license plates images.



Fig. 3 Locally collected license plates images

Our team went on Pune streets, car parking’s and collected 250 images of random license plate of both two- and four-wheelers for training on OCR [7]. We made sure we did not use inbuilt Pytesseract and build our own so that we could make necessary modifications. We build our tesseract engine. We used a license plate-cropped image that was extracted from video clips for testing our OCR. Figure 4 shows images used to test on OCR.

For saving license plate color same as the color of Excel Sheet cell, we used OpenCV2 for training 60 images as well as testing our model on 50 images of different colored license plates. We collected the multicolor license plate dataset from different sources on the internet. Figure 5 shows different colors of the numbers plates in India.

We have used a 1200 images dataset of people with a helmet and 1000 images of people without helmet which we got from Kaggle on all kind of two-wheelers along with 18 video clips available on Google and YouTube for testing helmet detection. We had to perform data cleaning, and include all kinds of two-wheelers, namely



Fig. 4 License-cropped images for testing tesseract OCR



Fig. 5 Different color license plate seen on Indian roads

Fig. 6 People with and without helmet on all kinds of two-wheeler



motorbikes, scooters, and bicycles. Our team have created two folders for positive, negative images and divided the helmet, no helmet images and put them in these folders. Figure 6 shows people with or without helmets on various two-wheelers.

For the vehicle classification into six categories, namely car, auto-rickshaw, motorbike, truck, bus, and scooty, we have collected 10,000 images for the training dataset which includes 4000 augmented images and 1100 images for the testing dataset of this multi-class classification. We have collected the car dataset from motorbike and scooty dataset were extracted from the dataset used for helmet detection dataset and were manually collected for trucks, buses, and auto-rickshaws. The dataset for all six classes came up to 1833 images each. We collected 18 YouTube clips using screen-o-matic and tested them. Figure 7 shows examples of the six classes we can detect.

For front- and rear-side detections of the car, we collected 35,000 images of four-wheeler from the Indian vehicles license number plate dataset on Kaggle and other free sources on the net and then divide them into two classes, namely front view and rear view. We had to filter out images in unwanted format manually for the training and testing part and checked it is working on ten separate video clips. Figure 8 shows the front and rear possess of the cars.

2.2 Algorithm Used

Object Detection: This is a computer vision technique that is extremely useful for identifying objects and locating them in a video or image. It can be used to count objects, classify objects, and determine their position, speed.

There are two kinds of machine learning problems that we deal with, namely classification and regression problems. In our system, we dealt with problems involving the classification of objects. All the algorithms that we used in our system were



Fig. 7 Different categories of vehicles images



Fig. 8 Front-view and rear-view images of cars

supervised learning algorithms. The list of machine learning algorithms and datasets used are given below in Table 1.

3 System Design

Figure 9 is the layout of the whole system design.

Table 1 Details of algorithms and dataset

Sr. No.	Topic	Technique/algorithm used	Number of training samples	Number of testing sample	Number of classes
1	License plate detection	XML file (HAAR cascade)	800	200	6
2	OCR on license plate	Built out tesseract engine	250	50	6
3	Storing license plate color	OpenCV2	60	50	6
4	Helmet/no helmet	CNN + mobile NET SSD	1800	400	2
5	Vehicle classification	KNN + CNN	10,000	1100	6
6	Vehicle pose estimation	CNN	28,000	7000	2

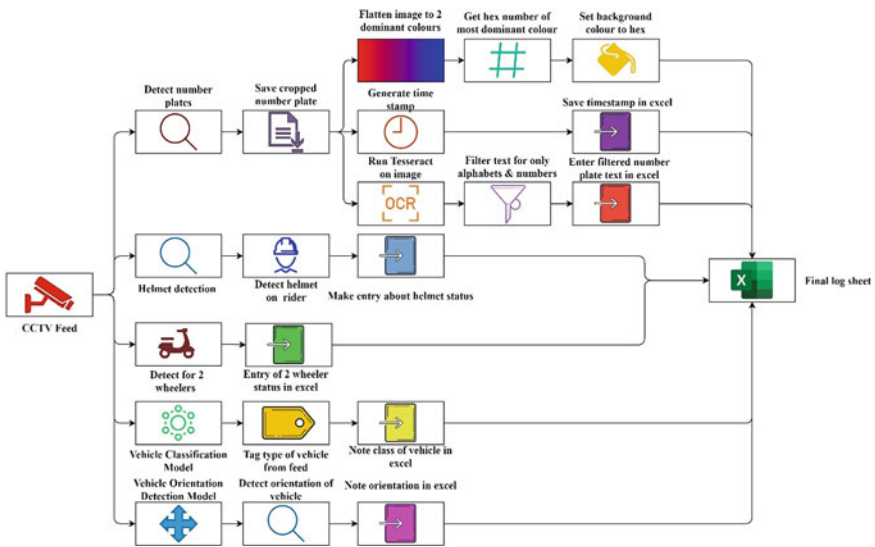


Fig. 9 End-to-end project pipeline

3.1 For License Plate Detection

Figure 10 shows all the preprocessing steps we used to improve our training accuracy. After cropping license plate, we did character recognition using our own OCR tesseract to detect the license plate.

Fig. 10 Preprocessing steps for license plate extraction

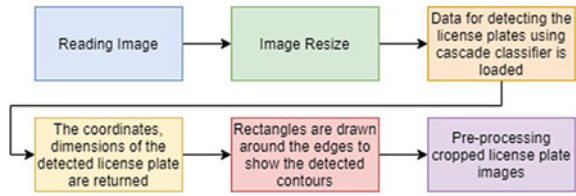
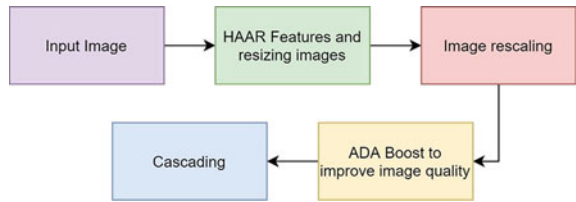


Fig. 11 Flow chart of HAAR cascading to create an XML file



All the task was performed on Google Colaboratory Python 3.6 using packages such NumPy, time, and cv2. We built our own cascade classifier and set up the minimum neighbors to seven and fixed scale factor to 1.3.

We used an XML file that shows class, frames, ID, height, covered by the object, and name the width of the box surrounding the object (here license plate). XML file in the data contains the information of all the bounding boxes [4]. A bounding rectangle box that acts as a reference point for detecting objects. We use data annotators to draw rectangles over images, X- and Y-coordinates are used to outline target objects in every image. We made our XML files for the images, which was used as a tool by the name Labelling that allowed us to draw visual boxes around objects in the images, and it automatically saved the XML files for our images. Figure 11 shows HAAR cascading.

3.2 Tesseract

The line finding algorithm is one such feature of tesseract that has already been worked on. It is designed to avoid losing image quality as a distorted page can be read without having to enhance the text. The key parts of the method are blob filtering and line construction. After the lines of text are found, we use a quadratic spline to fit the baselines more properly. This was never done before for an OCR system, and it helped tesseract to manage pages that had a curved baseline.

It is a standard artifact in scanning, and not just for book bindings. Tesseract tests of the text lines for checking out if they are fixed pitch or not. After tesseract finds the fixed-pitch text, the next step it does is cut words into individual characters by making use of pitch and disabling the chopper and the associator on these words for the proper word recognition step.

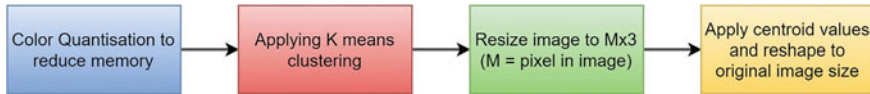


Fig. 12 Preprocessing steps for storing license plate color shade in excel

A reason for the popularity in any character recognition engine is to spot how a word can be segmented into several characters. The initial segmentation output from line finding is assessed first. For non-fixed-pitch text, we use the word recognition step. We implemented this task on the visual studio code platform.

Tesseract is a free open-source OCR engine. Text lines are checked for determining if they have fixed pitch by using tesseract. When tesseract gets the text with fixed pitch, it cuts the words in individual characters with the use of pitch and disables both associations as well as the chopper present on these words for the word recognition step. Tesseract will give output results as plain text, OCR or in a PDF, with text overlaid on the original image. Fixing text size, try to fix illumination of image, binarizing and de-noise image and trying to fix illumination of the image.

Check if the Excel Sheet for the day is made, if not make one. Check the folder where the detected images are stored and pick the oldest image. Run the image through tesseract and specify the language for greater efficiency. Filter the text so only alphabets and number are stored. Flatten the image into two colors so that we can the main color of the plate. Get the hex code of the dominant color. Make the entries into the sheet of the following things:

1. Timestamp.
2. Filtered text.
3. Set the background of the cell with the number plate with the respective color.

3.3 Saving Background Color of Excel Cell Same as License Plate Color Shade

Figure 12 is showing the method for saving license plate color same as the color of Excel Sheet Cell, we used OpenCV2.

3.4 Helmet/No Helmet

Refer Fig. 11 for preprocessing steps.

The model was saved and run on a video using Mobile Net SSD. We have used CNN algorithm with helmet/no helmet images and divided them into positive and negative class and used the model to make it work on high-resolution videos with the help of the Mobile Net SSD algorithm for two-wheeler detection and pedestrian detection.

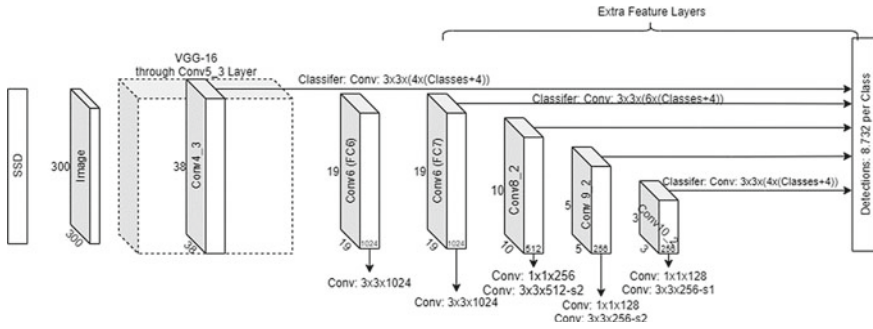


Fig. 13 SSD architecture [20]

Convolutional neural network algorithm (CNN) is a deep learning algorithm that can take in an input image and differentiate between different types of objects in it by assigning weights and biases to various aspects/objects in the image.

We have trained the CNN model using Python 3.6 and imported libraries like Numpy, Matplotlib, random, Keras, and cv2. As we have a large scale of images, so we resized the image to 50×50 . Activation function at hidden layer we used is rectified linear (ReLU) and at output sigmoid activation function with data format as channel last. We used a dropout of 0.5, 0.4, and “same” padding. For compiling the model, we chose binary cross-entropy as loss, Adam optimizer, metrics as accuracy with batch size of 64 and number of epochs were 200 for fitting the model perfectly.

Mobile Net SSD (also known as SSD 300).

Figure 13 shows SSD architecture.

We used Python 3.6 to load our CNN model and make it work on SSD 300 with packages like imutils, NumPy, time, cv2, and Keras [5]. We had to initialize the list of class labels. Mobile Net SSD was trained to detect different objects. We used rectangular bounding box colors for all the 21 classes which included categories like “background,” “bicycle,” “bus,” “motorbike,” and “person.” We used two classes from those above, namely “motorbike” and “person” to help us detect the helmet. For model compile, we used binary cross-entropy loss, RMSprop optimizer, accuracy as metrics. We kept the image size same as that of the CNN model, i.e., 50×50 .

Training Equation:

$$L(x, c, l, g) = 1/N \left(\begin{matrix} \underbrace{L_{conf}(x, c)}_{\text{confidence loss}} & + \alpha & \underbrace{L_{loc}(x, l, g)}_{\text{localization loss}} \\ \text{softmax loss} & & \text{Smooth } L1 \text{ loss} \end{matrix} \right) \quad (1)$$

is take as 1

Table 2 Comparison of SSD and YOLO algorithm

Algorithms	Mean average precision	Frames/sec	Batch size	Boxes	I/p resolution
YOLO (VGG16)	66	21	1	98	448*448
SSD mobile net	74	59	8	8732	300*300

N —number of default matched BBs

X —1 if the default box is matched to a determined ground truth box, 0 otherwise

l —predicted bb parameters

g —ground truth bb parameters

c —class.

Above Table 2 shows the compares the results of SSD and YOLO algorithms. Hence, it can be deduced that SSD is the superior algorithm of the two.

3.5 Vehicle Classification

Refer Fig. 11 for preprocessing steps.

We saved the model we trained using CNN and used the model on the video to detect six different classes of vehicles.

We used two algorithms on it, KNN and CNN algorithm. Out of the two, CNN gave us higher accuracy.

Distance Function:

$$\text{Euclidean : } \sqrt{\sum_{i=1}^k (x_i - y_i)^2} \quad (2)$$

$$\text{Manhattan : } \sum_{i=1}^k |x_i - y_i| \quad (3)$$

CNN:

The classification task was performed with Python 3.6 using NumPy, Pandas, Keras, Matplotlib, and PIL libraries with image resize dimension of 150X150 and a batch Size of 128. The class mode is set as binary with weight decay as small as 1e-4. Here, we used “same” padding, Activation function as ReLU for hidden layer and activation function as SoftMax at output using multiple dropouts like 0.2, 0.3, and 0.4 along with a learning rate of 0.001 and 30 epochs. The most suitable optimizer was Adam, with loss as categorical cross-entropy and metrics as accuracy.

K-nearest neighbors (KNN) algorithm is a nonlinear classifier and a machine learning algorithm that is supervised in nature. It is used for solving classification as well as regression problems. KNN is best for object detection. Classification of

objects is based on the training examples that are the closest in KNN feature space. It can predict the classification of the new sample point if the given data is of several classes. KNN algorithm requires no training before making predictions. New data can be added which will not impact the accuracy of the algorithm.

We have used sklearn, imutils, NumPy, cv2, glob, Nltk, Pickle, Matplotlib library for it. We resized the image to 64×64 with 42 random states [6].

3.6 Front and Rear-View Detection

For vehicle pose estimation, we did image cleaning, built the model, and saved the model, we trained using the CNN algorithm for binary classification and used the model on the video.

We used Python 3.6 platform using libraries such as NumPy, Keras, and Pandas. We had image width and height resized to 150×150 and a batch size 32 with class mode as binary. The activation function present in hidden layers is ReLU, the sigmoid activation function is present at the output. For model compile as we have two classes, we used loss: binary cross-entropy loss, optimizer as RMSprop, metrics as accuracy with 25 epochs.

Refer Fig. 11 for preprocessing steps.

4 Results and Analysis

4.1 Result

Given below we can see that the color of each vehicles license plate, its registration number, details of the presence of a helmet on a rider, front-rear side of vehicles, and vehicle classification based on its type.

Table 3 gives us an example of how the final Excel Sheet will look like with all the details logged for the vehicles passing the entry point.

Table 3 Final output on Excel Sheet (in table format)

Timestamp	Type of vehicle	Front or rear	Helmet status	Number plate
08-33-19.523896—28-08-2021	Car	Front	–	KL55R2473
08-33-19.533222—28-08-2021	Car	Front	–	MH20CS191
08-33-19.542994—28-08-2021	Car	Front	–	DL7CN5617
08-33-19.552687—28-08-2021	Bike	–	Yes	DL3CA9324
08-33-19.562572—28-08-2021	Truck	Rear	–	MH12CO6648
08-33-19.573216—28-08-2021	Tempo	Rear	–	UP14CP6748

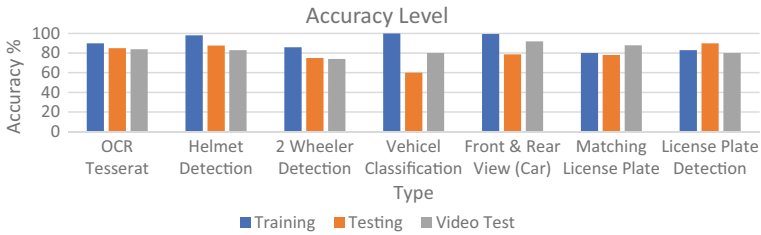


Fig. 14 Comparison of all model accuracy on the training dataset, testing dataset, and video clips

4.2 Accuracy Chart

Figure 14 is the accuracy chart showing the testing and training accuracy. We observed that we received the highest accuracy for the model having front and rear views of cars.

1. License plate detection: The license plate for each vehicle was detected the training accuracy was 83% and testing was 90%, cropped and saved in a folder for the OCR algorithm to be run on it. The license plate images were properly identified and stored. Figure 15 shows saved cropped images from video clips
2. Tesseract OCR for recognizing license plate in an Excel Sheet along with the date and time is given below. Figure 16 shows OCR output.

We observed that the OCR algorithm gave 90% accuracy for training and 85% accuracy while testing.

3. Helmet detection using CNN.

Given below we have plotted a chart showing final accuracy for training as 98.14% and loss as 7.13%, and validation accuracy at 87.76% and the loss is 17.23%. Figure 17 plots the accuracy.



Fig. 15 Saved cropped license plate from the video clip

Fig. 16 Recognized license plate text

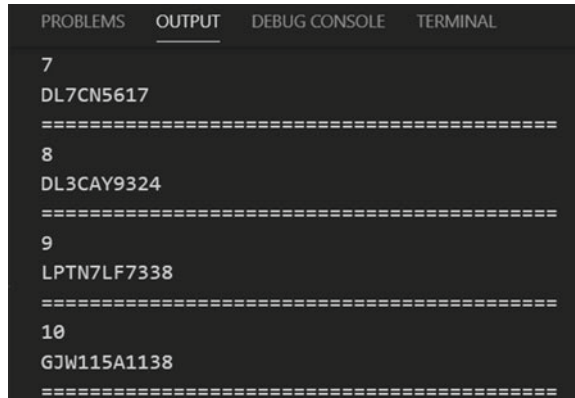
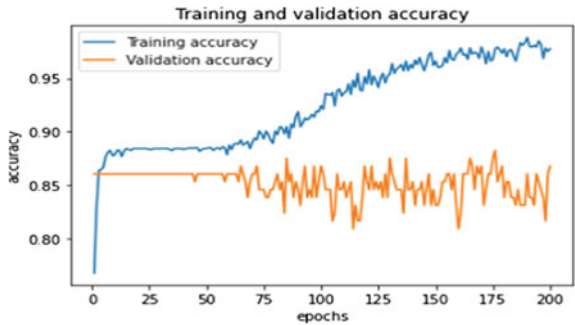


Fig. 17 Plot for accuracy chart



We used Mobile NET SSD for detecting two-wheeler with saved CNN model for helmet detection. Figure 18 shows detected helmets.

- 4. For classifying vehicles into six class.

We used K-nearest neighbor algorithm and convolutional neural network algorithm and successfully managed to place bounding boxes around vehicles in video clips to classify them.

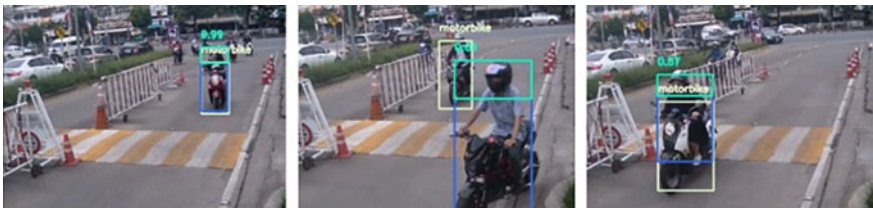


Fig. 18 Detecting presence of helmet on video clips with prediction accuracy

Table 4 Vehicles detected in the frames

Saved image 25	Car
Saved image 49	Car
Saved image 73	Car
Saved image 97	Car
Saved image 121	Bike
Saved image 145	Truck
Saved image 169	Scooty
Saved image 193	Car

Using CNN algorithm:

Table 4 shows the class of vehicle detected. CNN algorithm gave higher accuracy of 99.97% for the training dataset and 60% for the test dataset.

We have achieved validation accuracy by 51% using KNN supervised learning algorithm.

5. Front and rear side of car classification using CNN.

Table 5 shows the orientation of the vehicle detected. We got the final accuracy for training as 99.46% and testing accuracy at 73.7%.

6. License plate color is saved in Excel Sheet cell as background color (hex)

Figure 19 shows flattening of the image to detect the color of the plate. We detected the color of the license plate with an accuracy of 90% and stored it in the Excel Sheet along with the license registration plate successfully.

Table 5 Vehicle orientate in the frames

Saved image 25	Front
Saved image 49	Rear
Saved image 73	Front
Saved image 97	Front
Saved image 121	Rear
Saved image 145	Rear
Saved image 169	Front
Saved image 193	Front



Fig. 19 Input images to Output flat images conversion for storing license plate color in Excel sheet

5 Conclusion and Future Work

To summarize, we have tried to build a fully automated end-to-end system for storing all the entry log details of entrance and exit at a gate using machine learning as well as deep learning algorithms. The system works on live video recording at multiple angles and gives lots of details of vehicles at a single go. The system has been tested well and is giving an overall accuracy of 90% for vehicle license number (text + color), 60% for vehicle type (two-wheeler/four-wheeler/truck/auto/bus), 73.7% for vehicle pose estimation (front/ rear side), 87.7% for rider helmet detection for two-wheeler (“Yes” for helmet detected, “No” for no helmet), 100% for timestamp (for recording the time of entry and exit in the log).

The current system can be modified in the future to give a higher accuracy, as per requirements useful additions can be made to the logs like the system can be linked to sensors to make a record of whether the vehicle is adhering to the speed limit that is set while entering the campus. The system can be trained for vehicles in overseas countries with different license plates over a variety of vehicles. The future extension of this work involves implementing this in real time and using it at the gates of the building.

References

1. Arróspide J, Salgado L, Nieto M (2012) Video analysis-based vehicle detection and tracking using an MCMC sampling framework. *EURASIP J Adv Signal Process* 2012(1):1–20
2. Duong M (2010) Access control model and labelling scheme for efficient querying and updating XML data. Doctoral dissertation, Victoria University
3. Electronics T (n.d.) RFID: the technology making industries smarter. [online] Blog.ttelectronics.com.
4. Espinosa JE, Velastin SA, Branch JW (2018) Motorcycle detection and classification in urban scenarios using a model based on Faster R-CNN
5. Hines M, Davison AP, Muller E (2009) NEURON and python. *Front Neuroinform* 3(1)
6. Hui J (2020) SSD object detection: single shot multibox detector for real-time processing. Medium. Retrieved 21 Jan 2022, from <https://jonathan-hui.medium.com/ssd-object-detection-single-shot-multibox-detector-for-real-time-processing-9bd8deac0e06#:~:text=SSD%20is%20a%20single%2Dshot,offsets%20to%20default%20boundary%20boxes>
7. Jain P, Taneja K, Taneja H (2021) Which OCR toolset is good and why: a comparative study. *Kuwait J Sci* 48(2)
8. Kaur M, Sandhu M, Mohan N, Sandhu PS (2011) RFID technology principles, advantages, limitations & its applications. *Int J Comp Electr Eng* 3(1):151
9. Kotsiantis SB, Kanellopoulos D, Pintelas PE (2006) Data preprocessing for supervised learning. *Int J Comput Sci* 1(2):111–117
10. Li Y, Huang H, Xie Q, Yao L, Chen Q (2018) Research on a surface defect detection algorithm based on MobileNet-SSD. *Appl Sci* 8(9):1678
11. Linge S, Langtangen HP (2020) Programming for computations-python: a gentle introduction to numerical simulations with python 3.6, p 332. Springer
12. Liu T, Fang S, Zhao Y, Wang P, Zhang J (2015) Implementation of training convolutional neural networks. arXiv preprint [arXiv:1506.01195](https://arxiv.org/abs/1506.01195)

13. Ongsulee P (2017) Artificial intelligence, machine learning and deep learning. In: 2017 15th international conference on ICT and knowledge engineering (ICT&KE), pp 1–6. IEEE
14. Rehman SU, Ahmad M, Nawaz A, Ali T (2020) An efficient approach for vehicle number plate recognition in Pakistan. *Open Artif Intell J* 6(1)
15. Sanjana S, Shriya VR, Vaishnavi G, Ashwini K (2021) A review on various methodologies used for vehicle classification, helmet detection and number plate recognition. *Evol Intel* 14(2):979–987
16. Skogan WG (2019) The future of CCTV. *Criminol Public Policy* 18(1):161–166
17. Smith R (2007) An overview of the Tesseract OCR engine. In: 9th international conference on document analysis and recognition (ICDAR 2007), Vol 2, pp 629–633. IEEE
18. Szegedy C, Toshev A, Erhan D (2013) Deep neural networks for object detection. In: Burges et al (eds) *Advances in neural information processing systems*, Vol 26, Curran Associates, Inc. 2013
19. Veres M, Moussa M (2019) Deep learning for intelligent transportation systems: a survey of emerging trends. *IEEE Trans Intell Transp Syst* 21(8):3152–3168
20. Wagner P (2012) Machine learning with opencv2. Accessed 10 Mar 2016 from <http://www.bytefish.de>

Automated Spammer Detection for Limited Length Social Media



Shilpa Mehta

Abstract Today, various famous information sharing and social networking services like Orkut, Twitter, Facebook, and others allow users to exchange messages. Twitter restricts the messages to a character limit. It has an open nature and huge worldwide user base, and spammers utilize this feature. Cybercrimes like spreading rumors, cyberbullying, trolling, and stalking are commonplace. The approaches proposed here try to characterize users on the basis of interactions with their followers. This proves useful as a user can control one's own activities, but controlling the activities and features of followers is not possible. Three classifiers, viz., decision tree, Bayesian network, and random forest are used for learning and feature identification. Dataset from both genuine real users and spammers was used for testing. Study indicates that interactive features and community-based features prove good in identifying spam users, far better than metadata-based decisions.

Keywords Twitter · Spam detection · Social media · Cybercrime

1 Introduction

Twitter and other social media platforms make the real-time information about users available to the world. Twitter, Facebook, Orkut, and other such platforms are online social networks (OSNs), enabling users to share things of interest. This includes news, their own opinions about various things, family outings and other photographs and videos, celebrations, and their moods. Several arguments occur over different topics, like politics, important events, current world situations, and so on. When a user tweets anything, followers get and retweet it. This allows the outspread of information to a broad audience. OSNs necessitate study and analysis of behaviors on online platforms.

Twitter started in 2006. It has a policy of limiting length of posts. Normally, a tweet may have a maximum of 280 characters. Users can follow other people

S. Mehta (✉)

ECE, SoE, Presidency University, Bangalore, India

e-mail: shilpamehta1.official@gmail.com; shilpamehta@presidencyuniversity.in

(including celebrities) as per their personal liking. There are also options to follow and subscribe to news channels. Each registered follower receives the status updates of the subscribed accounts. OSNs like Twitter and Facebook were made basically for social purposes, but they are open and have huge user bases. They have convenient real-time message propagation. These two features attract social bots and cyber-criminals.

OSNs are prone to new sophisticated and complex attacks and threats, like cyber-bullying, spreading misinformation, radicalization, and various other illicit activities. There is also a problem of classical cyberattacks, like phishing and spamming. As detection methods were developed, parallelly attacks also evolved into smarter versions to avoid detection. Reports say that a large percentage of users as well as tweets are bots and spam, respectively. Spam bots copy human behavior to get trusted, and then use it for untoward activities [1].

2 Literature Survey

Both industry and academia researchers are working to defeat cybercriminals, and to enable users to have a safe and pleasant experience on OSNs. Multiple spam detection approaches are in use on all platforms, and new ones are constantly proposed. Authors in [2] discuss a neural technique for classifying intercepted e-mail communications with multi-layer perceptrons with backward propagation algorithm. The mails are classified into personal and official mails. The same technique is easily expandable to spam detection methods also. The paper [3] discusses a fuzzy logic-based techniques for classifying a set of mails from intercepted communication. It uses counts of official kind of words like “interview,” “campus,” etc., versus the counts of personal kind of words like “mother,” “birthday,” etc.

Most spam identification and detection approaches use features from user own profiles and activities. While this may be necessary in case of e-mail and SMS spamming, we have another tool at hand when it comes to social media, where there are interactions between users. Spammers can be identified using the techniques traditionally used for mail, but additionally, interaction and community-based identification is possible. Spammers try to avoid detection by changing the style and content of their own posts. Most users who interact with each other know each other’s real identity. Real people belong to various communities, as per their neighborhood, friend circle, office circles, and personal interests in real life. Such people follow them back in their turn. As opposed to this, spammers follow random users, due to which they get very low reciprocation. Additionally, less frequent connections can happen among such followers, which limit the interaction as well as reduce community-based features. To avoid this kind of detection approach, spammers do follow each other and build fake communities. But, this will defeat their basic purpose of spamming real people, as they get restricted to stay within spamming groups. Moreover, large percentage of members of these fake communities will be spammers and possess similar behavior, which increases the chances of detection overall.

Spams have created problems right from the days of communications. Even when we get newspapers, they are always stuffed with advertisement pamphlets, which have converted to spam in today's electronic communication era. When Internet was beginning, these spam messages were reported in ARPANET in 1978 itself. But at that time, nobody worried about it. E-mails were just beginning, and spam was not yet recognized as the menace it has become today. But, the problem intensified with time. Many techniques have been developed to be able to catch different forms of unsolicited spamming (spam e-mail/social bots/and spam bots). In paper [4], identity deception prevention is proposed. It utilizes common contribution network data and tries to identify deceptive accounts trying to enter a sub-community. Deception implies deliberately transferring false information to a person who is unaware of the falsity. This appears effective as both a detection and prevention mechanism. But, efficiency is low, especially in large networks.

WARNINGBIRD [5] was designed to classify suspicious URLs found in spammers' Twitter stream and to act as a near real-time system. Its intention is only to classify and not to check upon the pages led from there. Correlations can be extracted from multiple tweets. Attacker's resources are not unlimited and hence are essentially reused, so the chains usually share URLs. Detecting such correlated or shared URL redirect chains works well. The limitations are speed and low utilization of resources. Reference [6] proposes a method for spam detection, which utilizes Twitter's own spam policy. A Bayesian classification algorithm distinguishes behaviors of normal and suspicious types. Graph-based features are quite useful, because spam accounts necessarily follow a large number of users. Spam accounts will also usually have multiple duplicate tweets when we check content-based characteristics, which leads to detection. However, real users may also post tweets repeatedly, so it is not a fool proof method.

In [7], the authors discuss how social bots operate, and we can thereby track their behavior. The paper [8] discusses strategies for identification of spammers on social networks. In paper [9], the authors discuss the Sybil attacks, where a malicious user gets or creates multiple fake identities in the system. Thus, the malicious user controls multiple nodes and is able to "out vote" honest users in collaborative tasks. In [10], the authors discuss methods for discovering various categories in SMS messages, which paves the way for spam detection of short length messages on social media. Unfortunately, as detection techniques evolve, so do spammer techniques to evade detection. For example, in paper 3 above, keyword counts are utilized. But, spammers very easily get around this by using word substitutions [11]. Once again, the scientific community has found multiple ways for identifying such word substitutions also [12].

3 Proposed System: A Hybrid Approach

The proposed approach works on a combination of various approaches, including content, community features, and interactions. It emphasizes on followers and community structures in network aspects. Network category is subclassified to

Table 1 Symbol descriptions

Symbol	Description
\overleftarrow{u}	Set of users who follow user u
\vec{u}	Users who are followed by u
$N(u)$	Total number of tweets done by u
$u \overleftarrow{v}$	A follower “ v ” who is following “ u ”
$\vec{u} \overleftarrow{v}$	Following set of “ v ” who is follower of “ u ”

community-based and interaction types, which comes from interaction network. Metadata features come from users’ own tweets, and content-based ones use the posts’ text quality and posting behavior. Spammers easily adjust their own posts to avoid content and metadata-based detection, but find it difficult to bypass followers and community-based features.

3.1 Methodology and Dataset

We discuss methods using a Twitter dataset with both benign and spammers for experimental analysis and evaluation. Followers and followings lists are available for each labeled user, along with their profile information. It also contains tweets details. Unfortunately, most benign users do not have followers, hence community and interaction features become ineffective, and the classifier gives a biased detection. Hence, we take up only the users with followers (1000 spammers and 128 benign users), which in turn causes imbalance. Hence, we use the SMOTE technique (i.e., synthetic minority oversampling technique) to create artificial samples from given samples and their nearest neighbors. In this paper, we will repeatedly talk about the tweets done by user A , the users who follow user A , the users who are followed by the user A , and subsets among such users. The symbols used for these are shown in Table 1.

3.2 Metadata-Based Features

Metadata represents information describing the tweet attributes and may be useful in finding the information source. Four such features are identified for use. In all the following topics, $N(u)$ stands for total tweets by the user.

Retweet Ratio (RR)

Automated spammers do not tweet like humans. Bots normally retweet the tweets of others, use tweets from databases, or create tweets using probabilistic methods. That is because there is no human sending any of his/her thoughts, rather it is a

computer program acting like a human tweeting. This feature can be assessed with the RR which is the ratio of a given user's retweets to total tweets. This should be low for real people. But, spammers and automated spammers will usually be retweeting other people's tweets, so it will be high for spammers.

$$\text{Retweet Ratio} = \{RT(u)\}/\{N(u)\} \quad (1)$$

[here, $RT(u)$ denotes the count of retweets made by user u].

Automated Tweet Ratio (AR)

Some spamming accounts use the APIs provided by the OSNs. Twitter has a public API, and this is used by spammers with multiple accounts. Tweets originating from unregistered and third-party applications are called as automated tweets. The AR of a given user is the ratio of tweets from APIs to total tweets of u . Spammers are expected to use API to generate tweets quite often. Hence, while AR is low for real people, it will be high for spammers.

$$\text{Automated Tweet Ratio} = \{A(u)\}/\{N(u)\} \quad (2)$$

[here, $A(u)$ is the number of tweets by the user u made using API].

TSD—The Tweet Time Standard Deviation

Automated spammers fix activity time using random number generator algorithms, which use certain distributions. Time activation functions are used to activate bots at specific time. TSD captures the tweet time variations. Mathematically,

$$\text{Tweet Time Standard Deviation} = \frac{\sum_{j=0}^{N(u)} (t_j - T)^2}{N(u)} \quad (3)$$

Here, t_j is the j th tweet time, and T is the average time between consequent tweets. As automated spammers will use such APIs, they will have low TSD.

Tweet Time Interval Standard Deviation (TISD)

This feature uses patterns in consecutive activities. Bots have regular intervals, while humans have irregular behavior. Mathematically,

$$\text{Tweet Time Interval Standard Deviation} = \frac{\sum_{j=0}^n (T_j - \bar{T})^2}{N(u)} \quad (4)$$

Here, \bar{T} is the mean interval, while T_1, T_2, \dots, T_n represent the time that passes between consecutive tweets. Just like TSD, this is also expected to be low for automated spammers for the same reason.

3.3 Content-Based Features

Current methods use content quality also as one indicator. Spammers post tweets to tempt and deceive users, and typical characteristics may be utilized. Some features are:

URL Ratio (UR)

Real users of twitter generally use the social media platform to post their own views and thoughts according to their domains of interest, and sometimes share news articles and stories. Their tweets usually will not have any URLs. On rare occasions when they are talking of some specific reference, or in shared items, their tweets may have URLs to the corresponding source pages. But, a user injecting URLs continuously is definitely suspicious and indicates an attempt to lure the reader to certain unwanted spaces. To detect this, a feature called UR is utilized. UR is given as

$$\text{URL Ratio for user } u = \{U(u)\} / \{N(u)\} \quad (5)$$

Here, $U(u)$ is URLs used by u . Spammers have to necessarily use URLs to lure people to their targeted sites. So, a large ratio of their tweets contain URLs, and the UR value thereby approaches 1. Benign users rarely use URLs, so they have small UR value approaching 0.

Unique URL Ratio (UUR)

URLs used excessively is a suspicious situation by itself, but repeated usage of same URL intensifies this suspicion. Spammers try to trap users by using same URL repeatedly, to lure them to their targeted malicious sites. This is identified using unique URL ratio for uniqueness of URLs in a users' tweets.

$$\text{Unique URL Ratio for user } u = \{UU(u)\} / \{U(u)\} \quad (6)$$

As above, $U(u)$ is total URLs in user u 's tweets and $UU(u)$ is unique URLs. As spammers use the same URL again and again, the UUR will approach 1 for spammers.

Mention Ratio (MR)

Twitter allows its users to tag other users in their tweets using "@userid." Real users use this normally when they tag photographs or videos of events/occasions, where the tagged person may have been present or may be interested in that particular occasion. Spammers misuse this feature by mentioning random users in tweets, with an intention to tempt the tagged person to click on the tag to find out about the sender.

$$MR(u) = \{M(u)\} / N(u) \quad (7)$$

where $M(u)$ represents the count of mentions made by user u . This also will be low for real people as they usually mention others rarely. However, as spammers mention more frequently, it will be high for spam bots.

Unique Mention Ratio (UMR)

In real world, many people are acquainted with each other, but interactions do not happen with all acquaintances. Hence, it is not expected that a user will be tagging everyone in his circle. Conversely, spammers tag randomly. Mathematically,

$$\text{Unique Mention Ratio of user } u = \{UM(u)\}/\{M(u)\} \quad (8)$$

Here, $UM(u)$ is unique mentions made by user u . If a real user has 500 acquaintances, he may be tagging hardly ten of them, who are close enough to him/ her and wish to be tagged in such posts. Hence, UMR will be low for real people. But, spammers tag everyone they can find, so this ratio is expected to be high for spammers.

3.4 Interaction-Based Features

On open social media platforms, the interaction of users with other users provides lot of knowledge. This can be utilized for making decision about fraud detection and customers real-world identity. Additionally, analysis can be done about customer behavior and prediction of a user's behavior. A given user can choose to follow others or subscribe activities, but cannot force others to follow back.

Follower Ratio (FR)

A user's followers count indicates his or her trust level among other users. People connected in the OSNs usually also know each other in real world, with the exception of celebrities who are followed by numerous unknown people. It is quite natural for such known people to follow each other (both ways) on the social media network too. This results in the follow back rate of normal users being normally high. The FR of user " u " represents the fraction of his followers in his or her trust network. As real users will be connected to mutually known people, their FR will be high. But spam bots enter random groups, where nobody knows them. So, they may follow others but are not followed back. Hence, their FR is low.

$$FR(u) = |\vec{u}|/|\vec{u} \cup \vec{u}| \quad (9)$$

Reputation and Reciprocity (R)

Real-world reputation of people in society is related to the views and trusts of their own community, and same applies in virtual world also. This means if A follows

B on twitter, then B is also expected to follow A, thus leading to a high reciprocity rate for A. The reciprocity rate indicates follow back on being followed. Please note the difference between the FR defined above and the R defined here. This feature is defined using equation,

$$R(u) = |\overleftarrow{u} \cap \vec{u}| / |\vec{u}| \tag{10}$$

Spammers have a low R as the community group they follow will not follow them back. It is high for benign users.

Follower-Based Reputation (FBR)

A users reputation can be inherited from connected users. This feature utilizes the reputation of user’ followers. FBR is the average reputation of given users’ followers. Mathematically,

$$FBR(u) = \frac{\sum_{u \leftarrow v} R(u \leftarrow v)}{|\overleftarrow{u}|} \tag{11}$$

$R(u \leftarrow v)$ represents the reputation of v who is a follower of user u .

Once again, usually spammers follow each other to be able to project a false image that they are real persons, so FBR is low for spammers.

Clustering Coefficient (CC)

The clustering coefficient of a node represents the density of interconnectivity of adjacent nodes, excluding connections passing through that particular node. It indicates trust level among neighbors. For genuine users, there are real communities between people. IF A knows B, and B knows C, it is expected that A also knows C and all three belong to mutually known communities. Hence, genuine user may have a high CC (close to 1) as there is high trust among users connected in the real world. Their network is dense compared with a spammer network. Figure 1 shows the network of user A, the inter-connection among the neighbors of A and the communities among these neighbors, excluding the connections going through “A.”

$$CC(u) = E_u / (K_u \times (K_u - 1)) \tag{12}$$

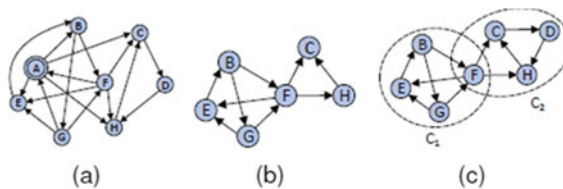


Fig. 1 a Network of A b inter-connection among neighbors c communities among neighbors excluding the connections going through “A”

3.5 Community-Based Features

Human beings have been living together in societies from ancient times. In real-world communities, users know one another and they have a trust level among themselves and high connection density among themselves compared to outsiders.

Community-Based Reputation (CBR)

Any given user's reputation is proportional to the reputation of their communities' and corresponding members. Communities associated with genuine users have good reputation, thus increasing the user's reputation. If u belongs to k communities, C_1, C_2, \dots, C_k , then the CBR value is

$$\text{CBR}(u) = (1/k) * \sum_{i=1}^k \left(\left(\sum_{j=1}^{|C_i|} R(C_i(j)) \right) / |C_i| \right) \quad (13)$$

$R(C_i(j))$ is the reputation for j th user of i th community. Spammers belong to their own interconnected communities, so the CBR is low for spammers.

Community-Based Clustering Coefficient (CBCC)

A user belongs to multiple communities. The CBCC indicates the clusters among the communities of a particular user. If CC_i is clustering coefficient of the i th community when u is a member of k communities, then CBCC for u is

$$\text{CBCC}(u) = (1/k) * \sum_{i=1}^k CC_i \quad (14)$$

CBCC will be high for real people and low for spammers, as independent users connected to spammers rarely have inter-communities.

4 Results

The performance was analyzed with three machine learning methods, viz., decision tree, random forest, and Bayesian network on the dataset. The metrics were.

- i. False positive rate (FPR),
- ii. Detection rate (DR), and
- iii. F-score.

The abbreviations used are:

TP → True positives; the number of spammers correctly classified as spammers.

FN → False negatives; the number of spammers classified wrongly as genuine users.

FP → False positives; the number of real people classified wrongly as spammers; and

TN → True negatives; the number of genuine users correctly classified as genuine.

Combining the FP and TN, we get FPR, or the false positive rate, which is the fraction of genuine users wrongly classified as spammers.

$$FPR = FP/(FP + TN) \quad (15)$$

A good classifier should have a low FPR, to be able to identify spammers correctly and not confuse genuine users to be spammers.

The DR (or recall) is the combination of TP and FN, the fraction of “detected” spammers to all the spammers in the network, which should be high in good classifiers.

$$DR = TP/(TP + FN) \quad (16)$$

Finally, F-score is mean precision, the ratio of the correctly identified spammers to all identified spammers. It shows discriminative power of the classifier. A high F-score classifier is desirable.

$$F - \text{Score} = 2 * \text{precision} * \text{Recall Ratio}/(\text{Precision} + \text{Recall}) \quad (17)$$

The proposed method is evaluated using three classifiers, viz., random forest, decision tree, and Bayesian network; using the metrics DR, FPR, and F-score as defined above. Table 2 shows the results of comparing the performance of various techniques discussed here. Table 3 indicates the performance with variation in benign user to spammer ratio.

Table 2 Performance comparison of metadata, content, interaction, and community-based features

Feature	Random forest			Decision tree			Bayesian network		
	DR	FPR	F-score	DR	FPR	F-score	DR	FPR	F-score
F	0.98	0.02	0.98	0.95	0.05	0.94	0.91	0.02	0.94
Metadata	0.97	0.03	0.97	0.93	0.06	0.95	0.92	0.03	0.95
Content-based	0.96	0.03	0.96	0.92	0.06	0.93	0.91	0.04	0.95
Interaction-based	0.93	0.03	0.95	0.94	0.05	0.94	0.85	0.05	0.9
Community-based	0.95	0.02	0.96	0.93	0.06	0.94	0.84	0.02	0.9

Table 3 Performance comparison with different benign to spammer ratio

Spammer to benign ratio	Random forest			Decision tree			Bayesian network		
	DR	FPR	F-score	DR	FPR	F-score	DR	FPR	F-score
1:1	0.98	0.02	0.98	0.95	0.05	0.94	0.91	0.02	0.94
1:5	0.92	0.01	0.93	0.89	0.02	0.88	0.88	0.02	0.90
1:10	0.87	0.00	0.92	0.86	0.01	0.86	0.90	0.01	0.94

5 Conclusion

Sighting spam has been a challenge from the beginning of electronic communication systems. Initially, the systems used to characterize spammers based on their own performance and profiles, but soon spammers started modifying their behavior to avoid being detected. The proposed approach beats this by working based on neighbor nodes (mainly the followers) not just on the users' own behavior, and also on their interaction network. Traditional metadata-based features do not perform too well as various algorithms have been devised by spammers to fool them. Interaction- and community-based features are found to be better performers.

References

1. Fazil M, Abulaish M (2018) A hybrid approach for detecting automated spammers in Twitter. *IEEE Trans Inform Forensics Secur* 1–1. <https://doi.org/10.1109/TIFS.2018.2825958>
2. Soundararajan K, Eranna U, Mehta S, A neural technique for classification of intercepted e-mail communications with multilayer perceptron using BPA with LMS learning. *Int J Adv Electr Electron Eng*. <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.639.3947&rep=rep1&type=pdf>
3. Mehta S, Eranna U, Soundararajan K, A fuzzy technique for classification of intercepted communication. *Int J Commun Eng Appl IJCEA* 3(1): 412–416. ISSN: 2230-8520; e-ISSN: 2230-8539, https://www.researchgate.net/profile/Shilpa_Mehta2/publication/267558855_A_Fuzzy_Technique_for_Classification_of_Intercepted_Communication/links/578f372b08ae9754b7ecc2e1.pdf
4. Tsikerdekis M (2017) Identity deception prevention using common contribution network data. *IEEE Trans Inf Forensics Secur* 12(1):188–199
5. Lee S, Kim J (2013) Warningbird: a near real-time detection system for suspicious urls in twitter stream. *IEEE Trans Dependable Sec Comput* 10(3):183–195
6. Wang AH (2010) Don't follow me: spam detection in twitter. In: *Proceeding SECUREPT*, Athens, pp 1–10
7. Boshmaf Y, Musluhkov I, Beznosov K, Ripeanu M (2013) Design and analysis of social botnet. *Comput Netw* 57(2):556–578
8. Stringhini G, Kruegel C, Vigna G (2010) Detecting spammers on social networks. In: *Proceedings ACSAC*, Austin, Texas, 2010, pp 1–9
9. Yu H, Kaminsky M, Gibbons PB, Flaxman A (2008) Sybilguard: defending against sybil attacks via social networks. *IEEE/ACM Trans Network* 16(3):576–589
10. Mehta S, Eranna U, Soundararajan K (2012) A neural technique for SMS classification using keywords search and identification of captured messages, using Hebbian learning. *IJESR* 2012 3(3):696–702

11. Fong S, Skillicorn DB, Roussinov D (2006) Measures to detect word substitution in intercepted communication. In: Proceedings of IEEE intelligence and security informatics conference (ISI 2006), May 23–24, 2006, San Diego, California
12. Fong SW, Roussinov D, Skillicorn DB (2008) Detecting word substitutions in text. IEEE Trans Knowl Data Eng 20(8)ss

Ensemble Model Discovery for Prognostication of Diabetes



Pranjal Bahore, Shreyansh Paliwal, Dipanshu Rautela,
and Rahul Chaurasiya

Abstract The global diabetes prevalence is estimated to be 10.8% by 2030 and is not expected to be plateauing in the future. Diagnosis of type 2 diabetes in the early stages is a very challenging task due to the complex interdependence of various factors. Various complex factor contributes regarding, the diagnostic measure of type 2 diabetes and hence predictive modelling in healthcare becomes quite a challenging task. In this paper, we compared standard ML algorithms (K-nearest neighbours (KNN), support vector machines (SVM), logistic regression, and decision tree) and ensemble-based algorithms (random forest and Xgboost) on the PIMA Indian women dataset sourced from the UCI repository. The performance was compared based on several metrics including precision, recall, and accuracy. Alongside, several data pre-processing techniques have been put in front to magnify the accuracy of predictions.

Keywords Automatic diabetes prediction · Machine learning · Ensemble learning · Supervised classification · Feature scaling

1 Introduction

The epochs of the twentieth and twenty-first centuries saw dramatic increases in life expectancy as diseases and mortality patterns shifted away from infectious diseases. Noncommunicable diseases (NCDs) are now the leading cause of premature morbidity and mortality accounting for 71% of global deaths [1], 77% of which are in low and middle-income countries (LMICs) [2]. The epidemiologic transition from infectious to chronic disease may have implications for perceived quality of life for the additional years of life lived. Diabetes mellitus (DM) stands among the major prevailing non-communicable disease warranting immediate attention. DM is a chronic condition in which blood sugar i.e. glucose, is not metabolized by the human body, and its level cannot be maintained because the body is not able to produce insulin or respond to it.

P. Bahore · S. Paliwal · D. Rautela · R. Chaurasiya (✉)
Maulana Azad National Institute of Technology Bhopal, Bhopal, MP, India
e-mail: rkchaurasiya.39@gmail.com; rkchaurasiya@manit.ac.in

Apart from culminating in death and disability due to heart attack and kidney failure, diabetes mellitus also affects the patients through its dreaded complications, the most prominent being cardiovascular disease, renal disease, and stroke-induced blindness, severely reducing the efficiency of a person [3]. Thus, complex approaches are required to get a complete picture of the burden of the disease. If the disease is diagnosed in time by prediction, then a person's health can be improved as levels can be controlled by changing the lifestyle, such as food habits, medications, exercises, etc. Therefore, if the healthcare system uses intelligent prediction mechanisms, then a person's life can be saved [4]. Thus, for LMICs, including India, a comprehensive robust national database is a necessity to understand disease incidence, prevalence, morbidity, and mortality.

Early diagnosis relies on the knowledge and experience of a doctor, but that can be inaccurate and susceptible to error. Although the healthcare sector collects vast bundles of data, those data cannot be used to discern undetected patterns for making effective decisions. Reliance on manual diagnosis can be dangerous as it relies on the healthcare personnel's observation and judgment that is not always accurate. Certain patterns may remain hidden from observation and can affect outcomes. Consequently, patients receive a low standard of care; hence, a sophisticated and advanced mechanism is required for early detection of disease with an automated diagnosis and higher accuracy. A multitude of hidden patterns and undetected errors give rise to an array of data mining and machine algorithms that can draw efficient results with improved accuracy. If not looked upon, a scale-up of 48% will result in 629 million people affected with DM by 2045, which is a severe condition. Various data mining algorithms have been developed to gather patterns buried within large healthcare data sets for the day-to-day growing impact of diabetes does not seem to be slowing down.

Scientific shreds of evidence had established the fact that sex and gender have a significant impact on the incidence, prevalence, symptoms, course, and response to many illnesses [5]. The role of sex and gender is fundamental in diabetes. Diabetes increases the risk of heart disease by about four times in women but only about two times in men, concluding women to be more prone, thus requiring an early detection. Other diabetes-related problems, such as blindness, kidney disease, and depression, are more common in women. Studies by researchers like [6–8] show that the induction of Diabetes can be through both environmental and genetic factors. Dataset on genetic diversity lacking thereof, we use the PIMA India women dataset containing several factors like blood pressure, BMI, diabetes pedigree functions, and others. From the studies, a pattern emerges between the dataset and a woman being diabetic, since the real-world dataset contains millions of elements, establishing a pattern within them is a cumbersome task. Hence, we have different statistical models and machine learning (ML) algorithms for pattern recognition. We propose the use of the Xgboost and random forest methods. Earlier models by [6, 9, 10] show random forest accuracy of about 75–78%. However, our study has achieved an accuracy of around 82%.

This study focuses on establishing a more efficient approach for the early detection of diabetes in women. The major contributions of the paper are as follows:

- Several unique data pre-processing techniques have been utilized to magnify the accuracy of predictions, such as log transformations and normal distributions.
- Performance of various standard ML algorithms (K-nearest neighbours (KNN), support vector machines (SVM), logistic regression, and decision tree) and ensemble-based algorithms (random forest and Xgboost) has been compared for predicting diabetes.
- The classification methods have been applied to the benchmark PIMA India women dataset. The performance is measured in terms of several metrics including precision, recall, and accuracy.

The rest of the paper is structured as follows: Sect. 2 presents a literature review on the existing methods of predicting diabetic conditions. The dataset and the features used for classification are described in Sect. 3. Section 4 presents the proposed methodology. Results along with the discussions are reported in Sect. 5. The work is concluded in Sect. 6.

2 Literature Review

With improved accuracy and precision in predictions, data mining has supplanted existing methodologies. Data mining and related ML methods are capable to detect the data, which remains hidden while using superior pioneering approaches. Earlier studies have discussed the concept of data mining methods that have been used successfully in driving prediction models, particularly for diabetes.

Application of ML techniques to identify diabetes dates back to 1988 when JW Smith and his collaborators published a paper [11] outlining a method to use an Adaptive algorithm to identify diabetes using ML techniques. With a decent accuracy of about 76% over the Indian PIMA dataset of diabetes onset of women, their study prompted many researchers to use ML algorithms to identify diseases such as diabetes. Reference [12] used general regression neural network (GRNN) to conclude the result on the prediction of diabetes on the PIMA Indian dataset with an accuracy of 80.1%. The study by [13] depicted the use of Weka software throughout all the phases of their study and combined it with the decision tree algorithm to facilitate a model with a resulting accuracy of 78.1768%. Reference [14] suggested a method of detecting diabetes using neural networks and SVMs in Pima Indian female populations.

Nowadays, data mining has become a crucial pillar of many sectors including health, banking, financial sector, education, etc.[15]. Reference [16] evidenced the superiority of AdaBoost technology with improved performance as it outperformed the bagging and J48 ensemble model in terms of accuracy. Through AdaBoost algorithm with decision stump algorithm, [17] modelled an algorithm to predict diabetes with 80.72% predictive accuracy. According to the results of [18], the naïve Bayes classifier provided the researchers with the highest accuracy of 73.558% in the

diabetes classification. A recent study by [19] applied SVM resulting in 70.4% accuracy and concluded that nearly 3% of global blindness can be attributed to diabetic retinopathy.

In the recent attempts to model the prediction of diabetes [9] application of random forest yielded an accuracy of 77.39%, [10] yielded an accuracy of 77.39% with random forest, whereas [20] managed to obtain an accuracy of 77.92% using Xgboost Ensemble Modelling.

3 Dataset and Features

The PIMA Indian women's diabetes dataset has been put in front by the National Institute of Diabetes and Digestive and Kidney Diseases [21], several restraints have been placed on the criteria of selection of these attributes from the larger database [22]. The Dataset consists of 768 rows of records and 8 columns (features). The target of our analysis is to predict the inception regarding diabetes established on these diagnostic measures. Specifically, patients present in the dataset are women of

Table 1 Feature description of the dataset

Attributes	DESCRIPTION	Count	Mean	Std	Min	Max
Pregnancies	Total pregnancy	768	3.845052	3.369578	0	17
Glucose	Glucose (plasma) level, after 2 h of an oral tolerance test	768	120.894531	31.972618	0	199
Blood-pressure	Pressure of blood (mm Hg) in diagnosis	768	69.105469	19.355807	0	122
Skin-thickness	Skin thickness in mm (triceps)	768	20.536458	15.952218	0	99
Insulin	Insulin (serum) level after 2 h (muU/ml)	768	79.799479	115.244002	0	846
BMI	Weight (kg)/(height (m) ²)	768	31.992578	7.88416	0	67.1
Diabetes-pedigree-function	Diabetes pedigree function	768	0.471876	0.331329	0.078	2.42
Age	Years of age	768	33.240885	11.760232	21	81
Outcome	Variable of class (0 or 1)					

PIMA Indian heritage with age over 21 years. Table 1 describes the feature vectors and their statistical value ranges. All the features are having numeric data types.

4 Proposed Methodology

The flow chart of the proposed method has been depicted in Fig. 1. For analysis, the data were pre-processed in such a way that ML algorithms can perform better on it. The classification algorithms were then applied to predict the diabetic conditions. Each step of the process is described in detail in this section.

4.1 Data Pre-processing

The unprocessed data consists of bogus or null values across several features [22], so it becomes very important to deal with those values for getting accurate results and predictions. Pre-processing has a many-fold advantage for analysis as it decreases the time and computational power required [23] and by scaling the given features our results don't get biased towards larger values. Pre-processing includes several methods such as data cleaning, standardizing values across features [24], removing outliers, and removing skewness from attributes.

In our dataset, we have zero values across some of the features that do not prove to be meaningful regarding the context of the information so we will replace these values with their median. Replacing odd values with either median or mode is a common technique in data processing as it nullifies the effect of outliers in the given features. After that, we have also removed the skewness from several features. The log transformation methods were used to remove the skewness in the distribution of the dataset. We have utilized Numpy (a python library) for the same. Other methods include square root transform and box-cox transform. Figure 2 depicts that all 8 features are normally distributed after preprocessing step. More reliable predictions are made as to the predictors and the target variable is normally distributed. The further data preprocessing techniques involve feature-scaling values across columns. We have used standardization as a feature scaling technique. Standardization is a

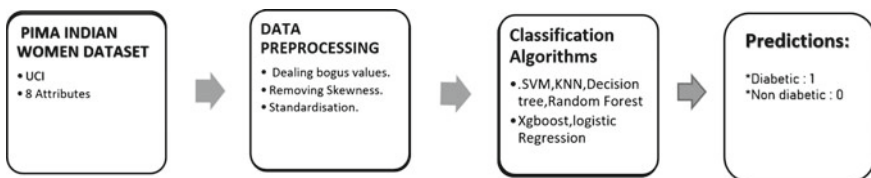


Fig. 1 Flow chart to represent the process for analysis of the dataset

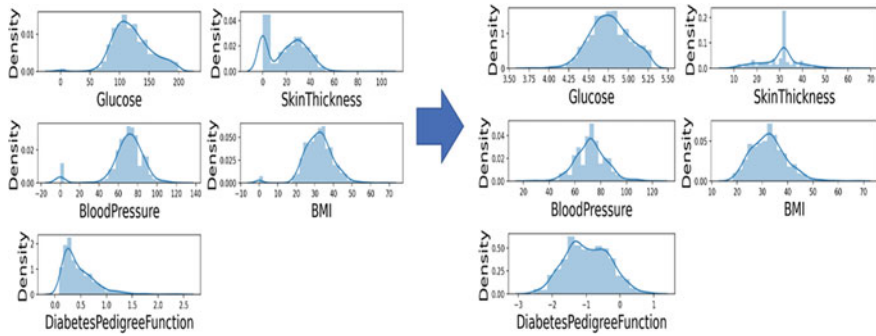


Fig. 2 Features are normally distributed after pre-processing

feature scaling technique where values are accumulated more around the mean, with a standard deviation of one unit.

4.2 Classification Algorithms

For the prediction of diabetes, we have used 6 different classification algorithms viz. random forest, KNN, SVM, logistic regression, Xgboost, and decision tree. Random forest and Xgboost algorithms are ensemble techniques. These ensemble methods combine various base models to generate one prime predictive model.

Random Forest

Comes under the Bagging subclass of ensemble techniques where multiple decision trees are aggregated to get results, hence also called Bootstrap Aggregation.

KNN

It uses a distance-based approach to classify data points, that attempts to determine, what group a data point is in, by looking at the data points around it, where 'k' in KNN is a parameter that tells us the number of nearest neighbours to include in the majority of the voting process.

SVM

It is a linear model which creates a hyperplane to separate data points into respective classes, moreover marginal distance is calculated from hyperplanes to support vectors (nearest points).

Logistic Regression

It predicts the probabilities regarding a data value based on prior observations of a data set, furthermore, the sigmoid activation function is used to convert values into discrete outcomes.

Table 2 Parameters for different classifiers selected by hyper-parameter tuning method

Classifier model	Parameters
KNN	k value = 29
SVM	DEGREE = 3, cache_size = 200, max_iter = -1
Logistic regression	C = 1.0, penalty = l2, tol = 0.0001
Random forest	n_estimator = 100, min_sample_leaf = 1, min_sample_split = 2
Xgboost	base_score = 0.5, learning_rate = 0.2, max_depth = 2, min_child_weight = 5, n_estimator = 50

Xgboost

Comes under the boosting subclass of the ensemble techniques, where multiple decision trees are used parallelly to produce optimal outcomes.

Decision Tree

It is a tree-based algorithm, which splits the values of the dataset into its complete depth and shows various outcomes over a series of decisions.

Various Parameters and Hyper-parameters are responsible for predicting the performance of any ML model. In our case, we have used GridSearchCV to perform hyper-parameter tuning and choose the best performing parameters as described in Table 2.

5 Results and Discussions

To forecast the onset of diabetes, various classification algorithms were used in this study. The classification results as the accuracy of predicting diabetes for different methods are presented in Table 2. From the table, it can be observed that the random forest and Xgboost classifier techniques, each with an accuracy of 82.00%, performed better than other methods on the PIMA India dataset. The ensemble methods (Random Forest and Xgboost) performed better than the traditional classification algorithms (KNN, logistic regression, SVM, and decision tree) because the ensemble techniques are meta-algorithms that join numerous ML techniques within a single prime model to lessen variance (bagging), bias (boosting), or enlarge prediction accuracy (stacking). The ensemble techniques are promising candidates for future data exploration and prediction of various multi-attribute diseases. Furthermore, it can assist the healthcare consultant and assist as a second assessment for better decision-making based on extracted attributes [9, 10].

Precision and recall metrics also help in figuring out the performance of different ML algorithms. Precision figures that, out of all the predictive positive results, how many of them are positive. It is also known as a positive prediction rate. Whereas, recall tells us that, out of total positive values, how many are predicted positive. It is

Table 3 Comparison of different classification techniques and their accuracies, precision, and recall values

Classification technique	Accuracy	Precision	Recall
Random forest	82	81	82
Xgboost	82	82	82
KNN	81.7	79	80
Logistic regression	79	79	79
SVM	78	77	78
Decision tree	77	76	77

also known as true positive rate or sensitivity. We have computed the classification performance based on these metrics as well. The results are presented in Table 3.

Numerous studies have been done before on the early prognosis of diabetes, these models also utilized data mining techniques such as ours. ML algorithms such as KNN have performed well, having accuracies ranging from 72 to 80% [4, 25, 26]. Logistic regression algorithm achieves results within the accuracies of 65–78% [9, 27, 28]. The Algorithms enlightened in this study vis-à-vis random forest and Xgboost achieved, as proposed, higher accuracies than others, falling within the range of 75–80% [9, 10, 29, 30] and 73–80% [31], respectively. Our results are in line with other results and validate that the ensemble method (Random Forest and Xgboost) are more suitable for early diabetes prognostication. The dataset pre-processing method we demonstrated in this study arrives at a comparatively higher accuracy in each of the models presented [4, 9, 10, 32]. However, our study utilizing Ensemble Model has achieved an accuracy of 82%.

6 Conclusion

This paper presented an automated method to predict diabetes using classical and ensemble-based ML algorithms. The methods were employed to predict the onset of diabetes in women of PIMA India heritage. The ML algorithms such as KNN, SVM, logistic Regression, Xgboost, decision tree, and random forest have been successfully geared on the training and testing dataset. Moreover, as a novel work, data pre-processing techniques such as feature scaling, feature engineering were applied the first time for this dataset, which lead to enhanced accuracy. Based on the experimental results, we conclude that the ensemble models provide the best substitute for traditional classification algorithms. The performance of the models was tested with only one data set. Various datasets could be utilized to test the performance of different models in the future. Deep learning methods and convolution neural networks will also be used as a classifier in future work.

References

1. Hanson MJAP (2019) The inheritance of cardiovascular disease risk, vol 108, no 10, pp 1747–1756
2. Wang Y, Wang JJBPH (2020) Modelling and prediction of global non-communicable diseases, vol 20, pp 1–13
3. Khazaei H, et al (2021) Medicinal plants for diabetes associated neurodegenerative diseases: a systematic review of preclinical studies, vol 35, no 4, pp 1697–1718
4. Bhoi SKJTJOCM (2021) Education, prediction of diabetes in females of Pima Indian heritage: a complete supervised learning approach, vol 12, no 10, pp 3074–3084
5. Mauvais-Jarvis F, et al (2020) Sex and gender: modifiers of health, disease, and medicine, vol 396, no 10250, pp 565–582
6. Time G (2016) On Diabetes
7. Ramachandran A, Ma RCW, Snehalatha CJTL (2010) Diabetes in Asia, vol 375, no 9712, pp 408–418
8. Zimmet PZ, et al (2014) Diabetes: a 21st century challenge, vol 2, no 1, pp 56–64
9. Tigga NP, Garg SJPCS (2020) Prediction of type 2 diabetes using machine learning classification methods, vol 167, pp 706–716
10. Arora A, Khan MO, Harwani H (2020) Prognostication of diabetes using random forest. *Int J Comp Appl* 175(29):40–43
11. Smith JW, et al (1988) Using the ADAP learning algorithm to forecast the onset of diabetes mellitus. In: Proceedings of the annual symposium on computer application in medical care, 1988. American Medical Informatics Association
12. Kayaer K, Yildirim T (2003) Medical diagnosis on Pima Indian diabetes using general regression neural networks. In: Proceedings of the international conference on artificial neural networks and neural information processing (ICANN/ICONIP)
13. Al Jarullah AA (2011) Decision tree discovery for the diagnosis of type II diabetes. In: 2011 International conference on innovations in information technology, 2011, IEEE
14. Zolfaghari RJIJCEM (2012) Diagnosis of diabetes in female population of Pima Indian heritage with ensemble of BP neural network and SVM, vol 15, pp 2230–27893
15. Woldemichael FG, Menaria S (2018) Prediction of diabetes using data mining techniques. In: 2018 2nd international conference on trends in electronics and informatics (ICOEI), 2018, IEEE
16. Perveen S, et al (2016) Performance analysis of data mining classification techniques to predict diabetes, vol 82, pp 115–121
17. Vijayan VV, Anjali C (2015) Prediction and diagnosis of diabetes mellitus—a machine learning approach. In: 2015 IEEE recent advances in intelligent computational systems (RAICS), 2015, IEEE
18. Deepika K, Seema S (2016) Predictive analytics to prevent and control chronic diseases. In: 2016 2nd international conference on applied and theoretical computing and communication technology (iCATccT), 2016, IEEE
19. You S, Kang MJKJOAI (2020) A study on methods to prevent Pima Indians diabetes using SVM, vol 8, no 2, pp 7–10
20. Bhulakshmi D, Gandhi G (2020) The prediction of diabetes in Pima Indian women mellitus based on xgboost ensemble modeling using data science. Technical report, EasyChair
21. Sarwar MA, et al (2018) Prediction of diabetes using machine learning algorithms in healthcare. In: 2018 24th international conference on automation and computing (ICAC). 2018, IEEE
22. Kadhms MS, Ghindawi IW, Mhawi DEJJOAER (2018) An accurate diabetes prediction system based on K-means clustering and proposed classification approach, vol 13, no 6, pp 4038–4041
23. Ayon SI, Islam MJJOIE, Business E (2019) Diabetes prediction: a deep learning approach, vol 11, no 2
24. Anand R, Kirar VPS, Burse KJIJSCE (2013) K-fold cross validation and classification accuracy of pima Indian diabetes data set using higher order neural network and PCA, vol 2, no 6, pp 436–438

25. Thirumal P, Nagarajan NJAJOE (2015) A science, utilization of data mining techniques for diagnosis of diabetes mellitus-a case study, vol 10, no 1, pp 8–13
26. Devi MR (2016) Analysis of various data mining techniques to predict diabetes mellitus
27. Wei S, Zhao X, Miao C (2018) A comprehensive exploration to the machine learning techniques for diabetes identification. In: 2018 IEEE 4th world forum on internet of things (WF-IoT), IEEE
28. Joshi RD, Dhakal CKJJOERP (2021) Health, predicting type 2 diabetes using logistic regression and machine learning approaches, vol 18, no 14, pp 7346
29. Khanam JJ, Foo SYJIE (2021) A comparison of machine learning algorithms for diabetes prediction
30. Mercaldo F, Nardone V, Santone AJPCS (2017) Diabetes mellitus affected patients classification and diagnosis through machine learning techniques, vol 112, pp 2519–2528
31. Li M, Fu X, Li D (2020) Diabetes prediction based on xgboost algorithm. In: IOP conference series: materials science and engineering. IOP Publishing
32. Bandyopadhyay S, Bose P, Goyal V (2021) Prediction of female diabetic patient in India using different learning algorithms

Classification of Epileptic Seizure Using Machine Learning and Deep Learning Based on Electroencephalography (EEG)



Mohammed Tawfik , Ezzaldden Mahyoub, Zeyad A. T. Ahmed, Nasser M. Al-Zidi , and Sunil Nimbhore

Abstract Epilepsy is a type of neurological brain disorder due to a temporary change in the brain's electrical function. If diagnosed and treated, there can be no seizures. Electroencephalogram (EEG) is the most common technique used in diagnosing epilepsy to avoid danger and take preventive precautions. This paper applied deep learning and machine learning techniques for detecting epileptic seizures and identifying whether machine learning or deep learning classifiers are more pertinent for the purpose and then trying to improve the present techniques for seizure detection. The best performance of the deep learning models has been achieved by implementing the convolutional neural network (CNN) algorithm on the EEG signal dataset in which the result appears as follows: accuracy 99.2%, specificity 99.3% and sensitivity 98.7%. For hybrid deep neural network CNN with long short-term memory (LSTM), the accuracy reached is 98.7%.

Keywords Convolutional neural network · Epilepsy · Seizures · Electroencephalography · Long short-term memory

1 Introduction

Epilepsy is a chronic neurological disorder that makes brain activity abnormal because of the brain's neurological electrical discharging. Epileptic seizures are considered as a positive sign of brain disorder [1]. It has many symptoms, such as sudden disruption, arm and leg spasms, fear and anxiety. According to WHO, about 50 million people in the world have epilepsy, 2.4 million annually [2]. However, approximately 70% of epileptic people, if correctly diagnosed and treated, can experience no seizure. Epilepsy has two types of seizures generalized primary and partial

M. Tawfik (✉) · E. Mahyoub · Z. A. T. Ahmed · S. Nimbhore
Department of Computer Science, Dr. Babasaheb Ambedkar Marathwada University Aurangabad,
Aurangabad, India
e-mail: kmkhol01@gmail.com

N. M. Al-Zidi
Faculty of Administrative and Computer Sciences, Albaydha University, Albaydha, Yemen

Table 1 Frequency of brain status

Frequency (Hz)	Type
13->	Beta
7.5-13	Alpha
3.5-7.5	Theta
3 Hz-<	Delta

[3] where the partial seizures paroxysmal discharge is in focal areas of cerebral cortex temporal lobe. The abnormal discharge originates from the diencephalic activating system and spread.

Seizure occurs suddenly and frequently. It may be impossible to locate epileptogenic areas due to the seizure's electric activity, and maybe it explodes and spreads over a wide range of cortical surfaces. Therefore, there are some technological tools that help epileptic people and give them hints to take precautions before it happens, enabling them to take safe actions and avoid falling or driving. The individuals should know about the indicators of causing epileptic disorders and indicators before it happens. The researchers are trying to develop instruments for predicting the most common EEG, MRI by recording the electrical activity that is called neurophysiological measurement from the surface of the head and skull by electrodes attached to an individual by wire. The brain activity for epileptic people can be preictal, interictal and ictal.

To locate the accurate epileptic seizure position is still a challenge. For example, the measurement applied to the postsynaptic neuron potential summation does not exceed an area of (1–6 cm) of the cortex. The acquired waveform reverts the cortical electrical action. The frequency of the wave of Delta starts at 8–13 Hz, delta <4 Hz, beta 13 Hz. Table 1 describes the frequency of brain status [4].

Delta is usually a normal activity in newborn children till one year and in stages 3 and 4 of sleep, which is above abnormal. The beta normal activity appears during paying attention, worries or eyes open or closed. Alpha is called normal relaxation. It occurs if the eyes are closing and relaxing and fades when opening eyes. Theta is considered normal in children up till 13-year-old, and above this age, it can be classified as abnormal. The researchers proposed a technique for distinguishing and predicting the epileptic seizure by extracting EEG signal features using fast Fourier transform (FFT), discrete wavelet transform (DWT), autoregressive (AR) and Mel-frequency cepstral coefficients (MFCCs). They applied ML classification algorithms to achieve high accuracy with the availability of data. We employed deep learning to achieve the highest accuracy to classify seizures.

2 Related Work

Deep learning and machine learning become one of the most effective disease detection and predictions techniques especially with a large dataset. A few studies incorporate deep learning, and the others use machine learning for epileptic seizure prediction; here, we report the most significant studies related to the study. There are various methods for feature extraction from EEG signals and classifiers for epileptic seizure detection. The most common feature extraction is permutation entropy (PE), FFT, DWT and AR model.

Ouyang et al. [5] developed an AR model to recognize EEG signals among controls and patients with epilepsy. The data have been collected from 23 patients. The model achieved 87.54% accuracy using the boost classifier. They used an Epilab software for feature extraction by a sliding window approach with non-overlapping windows and the duration 5 s, and EEG segment is divided into a set called EEG epochs. Every epoch contains 18 channel signals that apply finite impulse response (FIR), calculate and represent along with the channel signal and filter coefficients to get an 18-dimensional vector from each segment. Hindarto et al. [6] proposed a method for feature extraction from EEG using FFT and applied the root mean square RMS with average power spectral density to evaluate those acquired features. The data have used 200 signals, each signal has 1409 points, and then they apply ANN for classification; they achieved 92.5% accuracy with the number 60 hidden layers and 1000 Iteration.

Hassan et al. [7] have used the ensemble empirical mode decomposition with adaptive noise that has been designed for analyzing non-stationary and nonlinear signals. The study was conducted for each class using the AdaBoost algorithm. The accuracy for each class was as follows: class 1 97.6%, class 2 99.2%, class 3 100%, class 4 98.68% and class 5 100%. Chandaka et al. [8] used cross-correlation that is a mathematical operation to find similarity of two signals. They used a publicly available dataset from the University of Bonn, California. The result achieved was 95.96% accuracy by using the SVM classifier. Guo et al. [9] used a multi-wavelet transform based on approximate entropy which is a technique used to quantify the signal's complexity and regularity. They used public dataset from the University of Bonn and applied an ANN classifier. The overall accuracy result is 98.27%. Li et al. [10] proposed an automatic seizure detection system using deep learning on a dataset from Bonn University. Then, they applied nested long short-term memory (LSTM). The obtained accuracy was 98.44%.

Omar et al. [11] have applied DCNN to the EEG epilepsy dataset, which has been provided by Boston Children's Hospital for 23 patients. The overall accuracy is 90%. Minxing et al. [12] applied Stockwell transform which is a time–frequency representation to extract features from the EEG database provided by the Epilepsy Center of the University Hospital of Freiburg. The dataset consists of 21 records for patients who have epilepsy. They implemented bidirectional long short-term (BiLSTM). The achieved accuracy was 98.69%. Musa et al. [13] proposed a novel method for epilepsy detection by features extraction using dual-tree complex wavelet

transform and used complex-valued classifiers CVANN. The public dataset used was obtained from the University of Bonn. They achieved 99.5% accuracy using DTCWT with CVANN-2 and 100% by using DTCWT with CVAN.

Sari et al. [14] proposed FPGA-based real-time for the classification of an epileptic seizure. They used a dataset that consists of 822 signal recorders obtained from Temple University Hospital, Seizure Detection Corpus (TUH EEG Corpus). They implemented continuous wavelet transform (CWT) for feature extraction. The experiment was conducted on field programmable gate arrays (FPGAs) for real-time classification using ANN. They achieved 95.14% accuracy.

3 Materials and Methodology

Applications of machine learning and deep learning have a widespread technique recently in health and biological fields for better diagnosis and prediction. This paper aims to compare machine learning to deep learning models to get a high-accuracy model to diagnose epileptic seizures.

3.1 Dataset

For this study, the publicly available EEG dataset from Bonn University has been used from the UCI website. The dataset consists of 400 people; 200 with epilepsy and 200 normal. The EEG signal has been acquired by the “PCI-MIO 16E DAQ” card system [15]. The dataset consists of 5 folders that contain 100 files, each file refers to a single person, either patient or non, each file recorded brain activity and different status for 23.6 s, and then they were sampled to 4097 data points and a total of 500 individuals divided and modulated 4097 data points into 23 slices. Every slice consists of 178 data points per 1 s, so they become $23 \times 500 = 11,500$ rows of data variables X1-X178. Y represents the target that consists of 5 classes {1, 2, 3, 4, 5}. Class 1 represents seizure activity; class 2 represents the area where the seizure has been located; class 3 represents a positive sign if the area is identified; class 4 represents the signal recorded during eyes closed; and class 5 stands for the signal recorded during eyes open classes 2, 3, 4, 5 which refers to those who do not have a seizure, while class 1 refers to those who have an epileptic seizure.

3.2 Preprocessing

Data preprocessing techniques transform the raw data into a readable, understandable and appropriate miner for statistical methods and machine learning algorithms [16]. The data have passed through some standardized binary classification methods: data

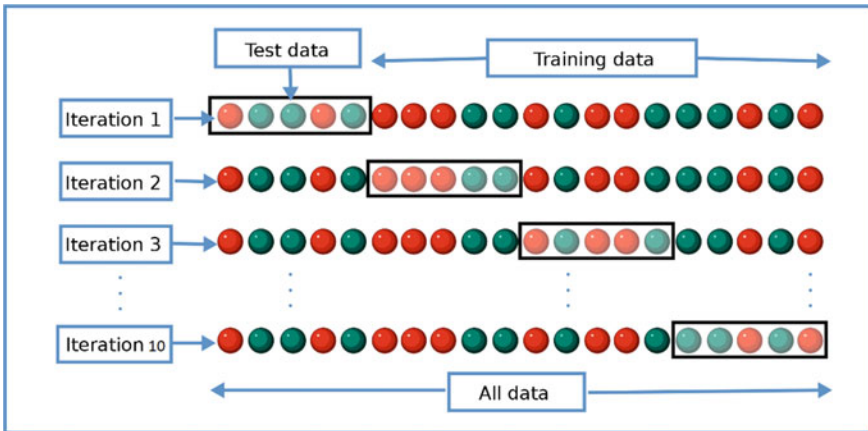


Fig. 1 Diagram of K-fold cross-validation (Modified from Ashfaq, 2018) [17]

cleaning, data quality assessment, data aggregations, data sampling, dimensionality reduction, feature encoding, data scaling and splitting process. The missing values have been processed by using the mean, Std., etc., excluding the first row and the remaining features were 178, 11,500 samples stored in the x variable. There are 5 classes (1–5). In this study, we have converted the value which is greater than 1–0 where class 0 refers to non-seizure and class 1 refers to epileptic seizure. Second, we applied a statistical method to split data called K-fold cross-validation. To avoid overfitting, data have been split into 10 groups as shown in Fig. 1.

In the feature scaling steps, we applied the standard scaler to minimize the value, which helps in speeding up execution that is performed using equation $z = (x - u)/s$ where u represents the mean and s is the standard deviation.

3.3 Machine Learning Models

The person with epilepsy can be detected using EEG signal and applying machine learning algorithms such as SVM, LR, KNN, decision tree, random forest, XGBoost and CatBoost.

SVM Classifier

SVM is a set of supervised learning techniques applied for classification or regression [18]. SVM classifier has been adopted in our model for seizure classification on EEG dataset. Our experiment has been conducted using python. A total of 2300 patients have seizures represented by class 1, and the non-seizure patients are 9200 represented by class 0. The evaluation of the model was done by using sensitivity and specificity. The hyperparameters were adjusted in the SVM model to achieve

Table 2 The results of machine learning models

	Model	Accuracy (%)	Sensitivity (%)	Specificity
1	SVM classifier	97.5	91.8	99.1
2	Decision tree classifier	93.2	81.7	96.3
3	Random forest classifier	97.7	95.1	98.6
4	Quadratic discriminant analysis	93.1	95.1	98.6
5	K-nearest neighbor	93.1	67.8	100
6	Extreme gradient boosting	98.1	93.4	99.3
7	CatBoost classifier	97.9	92.2	99.4

high accuracy like kernel = “poly,” gamma = “ scale,” and mixite = 15000. The evaluation of performance is shown in Table 2.

Decision Tree Classifier.

Decision tree (DT) is a type of supervised machine learning algorithm that is used to solve regression and classification problems [19]. The DT is used to identify a class or value according to which target it belongs using the decision rules from the training data itself. Decision tree classifier works like graph structure that consists of nodes. Root node represents the location where we take an attribute and ask some questions, the edges act as the answers to questions, and the leaves act as the class label or output. The decision tree provides a method for selecting the topmost attribute. The test condition evaluates each node for different attributes. It is the corresponding outcome by selecting the highest result. The accuracy result of decision tree is 93.2%. The performance metrics evaluates our classifier including accuracy, specificity and sensitivity as described in Table 2.

Random Forest Classifier

A random forest is an ensemble learning technique developed for regression and classification [20]. It is a construct of a large number of decision trees in which each decision tree in a random forest that produces one class, and the produced class that gets the highest number of votes will become our prediction models. The accuracy result of random forest is 97.7%. The performance metrics evaluates our classifier including accuracy, specificity and sensitivity as described in Table 2.

Quadratic Discriminant Analysis (QDA)

QDA is a type of supervised classification classic methods and probabilistic and statistical learning technique [21]. It is used for numerical attributes and nominal labels based on the discriminant analysis by determining which variables discriminate between two or more groups [22]. QDA is not different from LDA; only in the covariance matrix can it be different for each class Σ_k , for every class $k = 1, 2, \dots, K$. The quadratic discriminant can be found by using this equation:

$$k(x) = -12 \log|\Sigma_k| - 12(x - \mu_k)T \Sigma - 1k(x - \mu_k) + \log \pi_k \quad (1)$$

The classification can be found by using this equation:

$$G(x) = \arg \max_k \delta k(x) \quad (2)$$

The accuracy result of QDA is 93.1%. The evaluation of the performance of the QDA classifier is described in Table 2.

K-Nearest Neighbor

It is one of the topmost machine learning algorithms that is simple and easy to use for classification and regression. It is applied to various applications like healthcare, image and video recognition, finance, marketing [23], etc. K-NN is nonparametric that does not involve any assumptions as to the form or parameter [24]. It is considered of lazy learning types, which means the model does not need any training data points before predicting by storing the training data points until classification is performed. K-NN classifier work is based on closest training sample [25]. The corresponding data are classified into training and test sample points, and the evaluation is based on distance; the data point that has the lowest distance is called the nearest neighbor. The object is classified based on a vote of the nearest neighbor, and then the object is assigned based on the class most common among its K-nearest. In our classifier model, we are tuning the hyperparameters like n-neighbor = 3, weights = “distance” to achieve the high accuracy. The accuracy result of K-nearest neighbor is 93.1% as described in Table 2.

Extreme Gradient Boosting

Boosting is ensemble based on a learning technique that changes the powerless model to strong estimators that works sequentially as circular iteration [26]. Each one tries to correct an error that belongs to its predecessor model to minimize the loss function like learning in a neural network. Gradient Boosting is a boosting technique in which each iteration of the new predictor is created to fit on the pseudo-residuals (errors) of the predecessor predictor till getting final improvements, rather than adjusting the weights for each incorrect classified at each iteration. Ensemble machine learning includes several ML models to create one superior predictive model to get optimal accuracy [27]. XGBoost is a modern ensemble machine learning technique for classification, regression and ranking problems. It is an efficient and fast implementation of Gradient boosting structure to find the topmost tree model. Tianqi Chen and Carlos Guestrin created it on 2016-08-01 [28]. XGBoost is an application of the gradient boosting that is used to find the best and most accurate tree model and focused on computational speed. XGBoost supports parallelization, cache optimization, distributed computing, and processing large dataset. The accuracy result of extreme gradient boosting is 98.1%. The overall performance of XGBoost is shown in Table 2.

CatBoost Classifier

CatBoost classifier is a high-performance library for gradient boosting on the decision tree created by Yandex Russian company Technical in April 2017 [29].

It is the development of the matrix net algorithm. It is more accurate, extensible and robust. It supports textual, numerical and categorical features. CatBoost offers two features that make it more accurate in executing the ordered boosting [30]; it is a permutation-driven algorithm to process categorical features. It permutes the training data samples. Through the training phase, a group of decision trees is built. Consequently, a consecutive tree is built with reduced loss error compared to the preceding trees and the number of trees determined at the beginning using a parameter to avoid overfitting. In addition, there is an overfitting detector that helps to stop the training early. Then, the training parameters impose; it is operated automatically if the threshold's value is greater than the current p-value, current-P-value < threshold. Current p-value is calculated using metrics score, and ExpectedInc is calculated by the following formula.

$$\text{ExpectedInc} = \max_{i_1 \leq i_2 \leq i_0.99i - i_1} (\text{score}[i_2] - \text{score}[i_1]) \quad (3)$$

$$x = (\text{ExpectedInc}[i]) / (\max_{j \leq i} \{ \text{jscore}[j] - \text{score}[i] \}) \quad (4)$$

$$\text{CurrentPValue} = \exp \left\{ (0.5/x) \right\} \quad (5)$$

The CatBoost technique checks the iteration's number before building a new tree till the iteration is applied to the optimal loss function value. The building steps in CatBoost are:

1. Prior calculation of splits using quantization for the numerical features to find the best possible ways to split data into segments called buckets.
2. Transforming the categorical features into the numerical format using the ordered target encoding that is performed within two steps: permutation of the training data points to any random order and quantization to convert the targeted values from a floating to an integer point.
3. Selecting the right tree structure to select a structure following some rules. How "feature-split pair" is selected and assigned to leaf:
 - The list is formed of the candidates from ("feature-split pairs") then assign to the leaf as the split.
 - b-Several penalty functions are calculated for every object. All candidates obtained from the first step that has assigned to the leaf and finds the split with the smallest penalty that has been selected. The produced value is assigned to the leaf; the procedure continuously is repeated for all leaves till the number of leaves equals or matches the depth of the tree.

Score functions [31]:

$$L(f(x), y) = \sum w_i l(f(x_i), y_i) + J(f) \quad (6)$$

the purpose score type is used to identify the next split through the tree construction, and the goal of supervised learning minimizes the loss function. CatBoost supports several core functions, e.g., L2 Use first derivatives, Cosine, Newton L2 and Newton Cosine. The accuracy result of XGBoost is 97.9%. The evaluation for XGBoost classifier is described in Table 2.

3.4 Deep Learning Model

Deep learning is a subset of machine learning depending on the function of a neural network that consists of layers, weight, bias, and activation function. It provides solutions to so many complex problems like anomaly detection, image recognition, video recognition, etc. [32]. This section of this paper is the implementation of deep learning using neural networks such as MPL, ANN, RNN -LSTM and CNN-LSTM. Figure 2 shows the architecture of deep learning models.

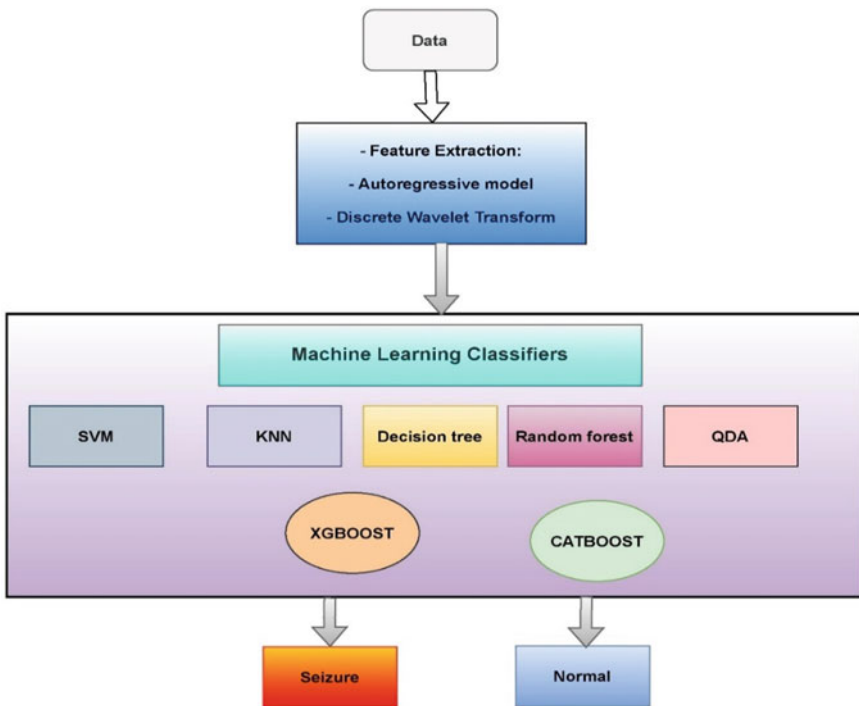


Fig. 2 The architecture of ML models

Multi-layer Perceptron (MLP)

Multi-layer perceptron (MLP) is a type of feedforward artificial neural network that consists of 3 layers: (i) an input layer, (ii) hidden layer and (iii) output layer of single hidden layer [34]. MPL classifier belongs to the Scikit-learn library. Figure 4 shows the MLP architecture, which uses a nonlinear activation function for neurons node like ReLU, tanh, etc.

The data training passes through input layer and then to the hidden layer for processing multiple inputs by weight, adding bias, etc., as shown in Fig. 3. The result is represented in the output layer. Here is the discussion for each layer. First, we feed the input data to the input layer (neurons), add weights, and calculate the summation. Second, we add the bias to the previous input multiplied by weight to shift every point in a particular direction that may be at left or right, etc., and then feed the summation to the activation layer for mapping the input values to output for classification or regression depending on the model function. MPL contains a variety of parameters that help to minimize the loss function based on the gradient descent technique [35]. This model is based on 150 hidden layers with Adam optimizer, and the training is done by 1000 iterations to get the highest accuracy of 98%, as described in Table 3.

Artificial Neural Network (ANN)

Artificial neural network (ANN) is a part of the computing system designed for information processing by mimicking the human brain through personifying the biological nervous system [36]. It consists of many interconnected neurons. It has

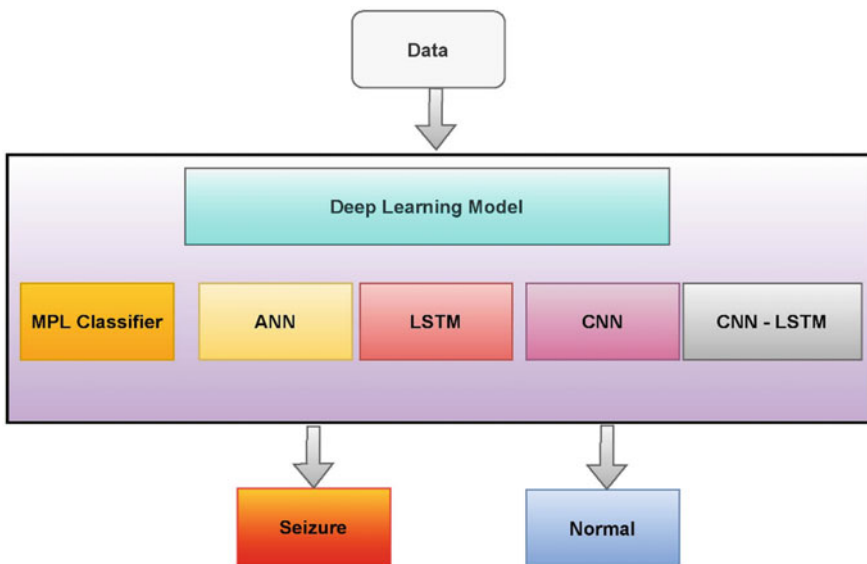


Fig. 3 The architecture of deep learning models

Table 3 The results of the deep learning models

	Model	Accuracy (%)	Sensitivity (%)	Specificity (%)	Optimizer	F-score Class 0	F-score Class 1
1	MPL classifier	98	93.9	99.1	–	0.99	0.95
2	ANN	98.6	97.15	99	RMSprop	0.99	0.97
3	LSTM	98.5	96.7	99.3	Adam	0.99	0.97
		98.3	94.7	99.3	Adagrad	0.99	0.96
4	CNN	99.2	98.7	99.3	RMSprop	1.00	0.98
		98.2	98.2	99.1	Adagrad	1.00	0.98
		98.6	95.12	99.6	Adam	0.99	0.97
5	CNN-LSTM	98.6	95.12	99.6	Adam	0.99	0.97

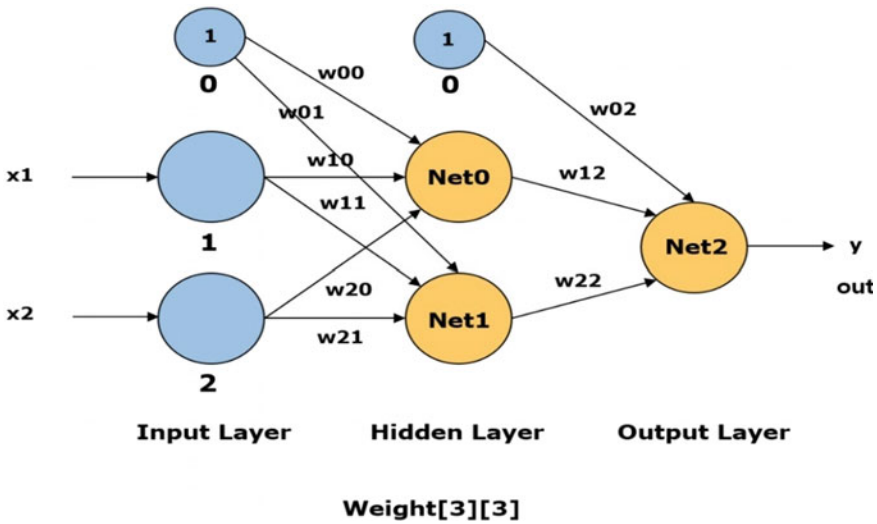


Fig. 4 The architecture of MLP, 2020 (Reproduced with permission from Taimoor) [33]

the same paradigm in the artificial neural network in which every cell is called node or neurons that receives the input. The components of ANN are the input layer/neurons, weights, bias, hidden layers, activation function, output layer that has two terminologies: feed-forward, back-propagation as shown in Fig. 5.

In feed-forward, we pass the input data to the node with its weight adding the bias and then we find summation for each node with weights and the bias adding to the sum. The function of bias is to move data point through the origin:

$$\sum F(X1.W1 + X2.W2 + b \dots) \tag{7}$$

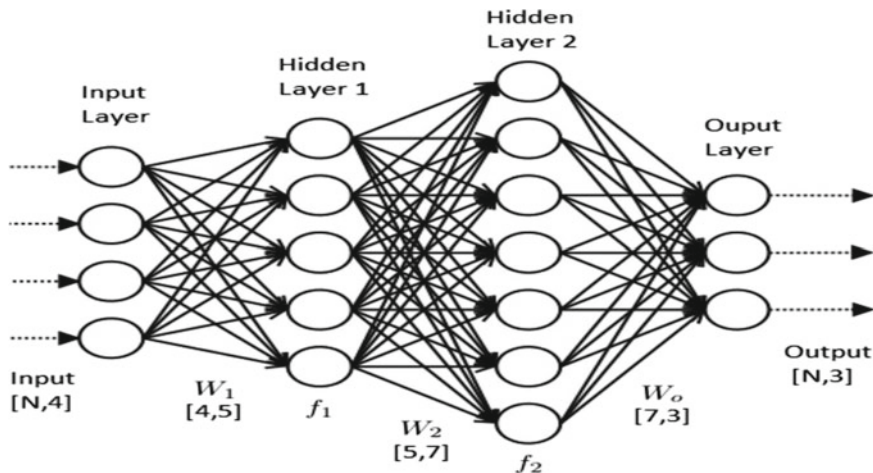


Fig. 5 The architecture of an ANN, (By Nadya Aditama, 2019) [37]

Then, the data pass through a nonlinear function called activation function for mapping input to output. There are types of activation functions such as sigmoid, tanh, ReLU. The backpropagation for minimizing the error (cost function $J(\Theta)$) equals predicted values—actual values till it becomes close to 0 by adjusting the weights. All that is performed using gradient descent or optimizer. The structure of our model for seizure epileptic classification is built using Keras API. The model consists of 100 neurons, activation function ReLU, sigmoid for mapping output for classification with Adam and Adagrad optimizers. The accuracy achieved is 98.6% on the validation of data, as described in Table 3.

Recurrent Neural Network

LSTM is a deep learning technique which is a special kind of RNN network created by Schmidhuber and Hoch Reiter in 1997. It is designed for memorizing the previous input data. It is used for classification, predicting time series [38].

LSTM provides an excellent solution for sequential data such as speech recognition, translation, image caption, and anything related by period time. It is a back-propagation terminology. LSTM has three gates: forget, input and output as shown in Fig. 6.

Input gate selects the appropriate value from input samples which is used to adjust the memory and input to the cell. The sigmoid function is used as the combining function for three gates: input, output and forget in which its output is between 0 and 1. Tanh or ReLU functions add weightage to the values that have passed calculated using formula:

$$it = \sigma(Wi \cdot [ht - 1, xt] + bi) \tag{8}$$

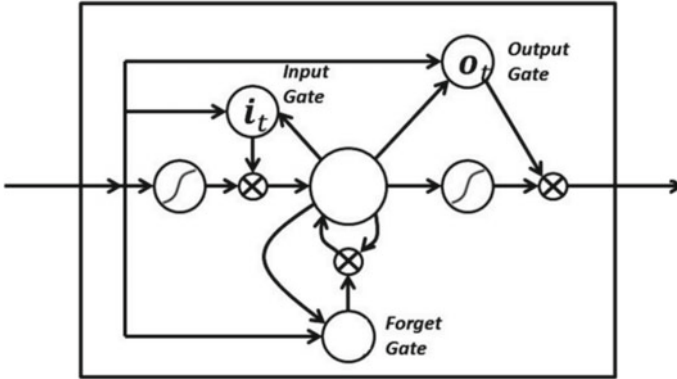


Fig. 6 LSTM gates (with permission from Greg Surma, 2018) [39]

$$C_t = \tanh(WC \cdot [h_{t-1}, x_t] + b_C) \tag{9}$$

Forget Gate

Forget gate is called sigmoid that selects which part of new information to store in the cell state or get rid of it out [40]. The input gate layer (sigmoid) selects the values that will update next tanh or ReLU. It creates an array for new candidates' values, $C_{\sim t}$. It will be added together to the state using formula:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{10}$$

Output Gate

The output gate selects the state of the next hidden state that should be remembered to be updated on the internal cell state. The hidden state decides what the information is to be kept to the next step [41]. The hidden state also is used for predictions.

$$o_t = \sigma(W_O [h_{t-1}, x_t] + b_o) \tag{11}$$

$$h_t = o_t * \tanh(C_t) \tag{12}$$

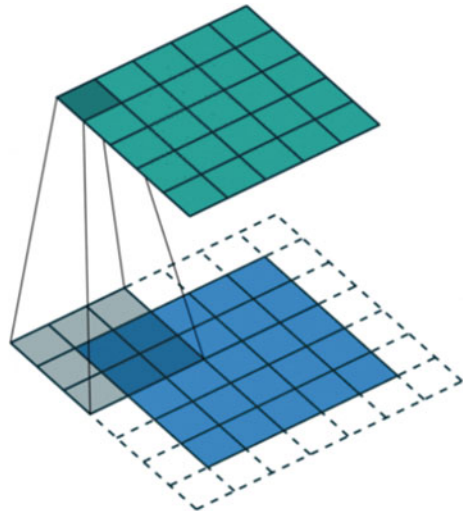
In our recurrent neural network architecture, LSTM, we are using 2 layers of LSTM with 200 neurons and many optimizers including Adam, Rmsprob, Nadam, SGD, loss sparse_categorical_crossentropy, Activation ReLU, early stopping with min mode for monitor loss, dropout = 0.2, 6 dense layers with different values, sigmoid for 2 classes, 100 epochs. The performance evaluation of LSTM with the Adam optimizer is described in Table 3. The accuracy achieves 98.5%. The performance evaluation of LSTM with Adagrad optimizer is described in Table 3. The accuracy achieves 98.3%.

Convolutional Neural Network (CNN)

CNN is a feed-forward deep learning network that is distinguished by extracting features automatically. It can be used in various fields such as image processing, pattern recognition, voice recognition, time series and natural language processing [42]. It is a special kind of neural networks that uses convolution that performs mathematical operations, linear transformation. Various types of convnets were first introduced in the 1980s by Yann LeCun [43]. Second, AlexNet was introduced by Alex Krizhevsky in 2012 that is based on multi-layered neural networks that reach 8 layers [44]. VGGNet 16 by Simony and Zisserman was introduced in 2014 to ILSVRC computer vision competition which consists of 19 layers [45]. Google Net was designed by Google in 2014 which consists of 22 layers, while residual neural network (ResNets) was introduced in 2015 with 152 layers [46].

- CONV layer captures some features from image or data and then execute the convolution operation by filters and scanning input data [46] that has width and height, conv layer has many parameters input and output channels as shown in Fig. 7 and filter with padding as shown in Fig. 8.
- Activation Layer is a nonlinear layer in the convolutional layer. Its function decides which neuron will be activated or not using calculation weighted sum and adding bias. There are many types of activation layers: sigmoid function, rectified linear unit (ReLU), tanh, leaky ReLU [48].
- Pooling layer is used to reduce the data dimensions from convolutional layer by function max pooling or average pooling. This stride collects outputs of the neuron at one layer into a single neuron. Stride function selects the number of pixels to which the window should be moved as shown in Fig. 9.

Fig. 7 Convolution layer



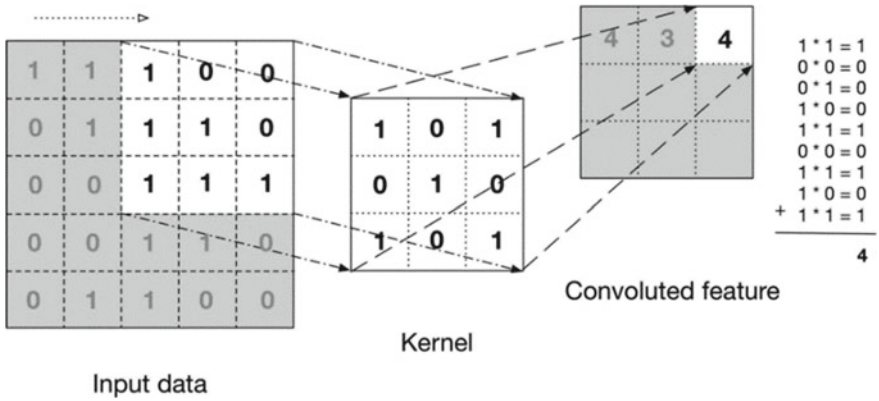
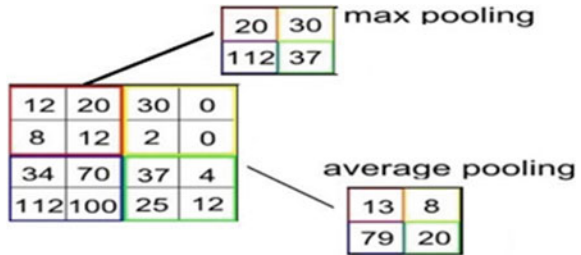


Fig. 8 Convolution layer (By Li Yin, Convolution layer, 2018) [47]

Fig. 9 Max pooling layer



- Fully connected (FC) layer consists of group MLP layers. FC is a function that makes full connections for all activations in the previous layers and converting them to a single vector. It applies the weight to predict the right label and find probabilities for every label separately [49].
- Softmax function is an activation function similar to tanh, ReLU. Sigmoid is applied on the output that depends on probabilities because it makes logistic function regression that is used for multi-classification [50].

Our CNN models consist of 5 layers of Conv1D as shown in Fig. 10 with 64 channels of filters, dropout layer, max-pooling, flatten, dense, sigmoid function for mapping the output. The model is trained on the cloud by using Google Colab, the batch size was 200, and the number of epochs was 100. The performance evaluation of CNN with RMSprop optimizer is shown in Table 3.

Figure 11 shows the confusion matrix. The accuracy of training and validation achieved is 99.2% as shown in Fig. 12. The loss of accuracy of training and validation is shown in Fig. 13.

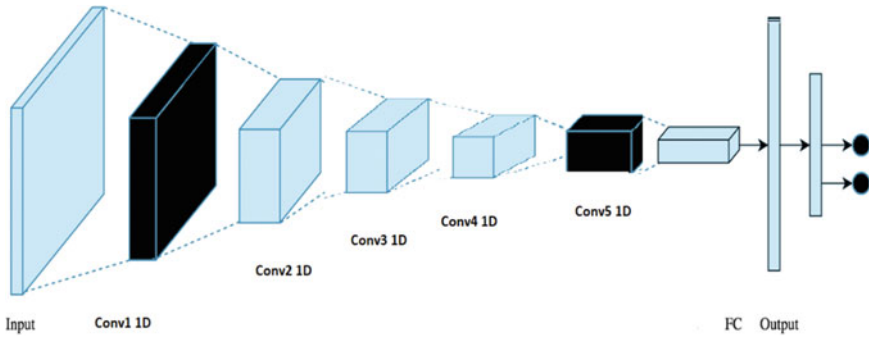


Fig. 10 The architecture of CNN model

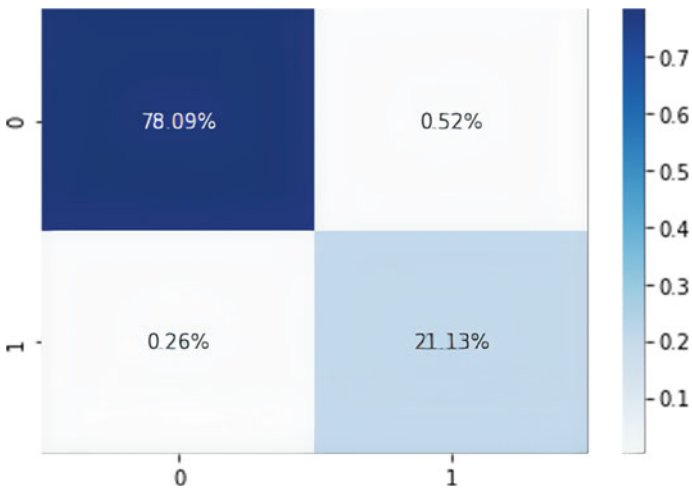


Fig. 11 Confusion matrix CNN model

CNN-LSTM Model

This model combines CNN Conv1D and LSTM [51]. The model structure has two layers of Conv 1D and two layers of LSTM as shown in Fig. 14. The accuracy achieved is 98.6% as described in Table 3.

4 Result and Discussion

Several studies have been conducted on an automatic seizure using EEG signals for prediction and detection. In this paper, we compared the performance of several different ML and DL techniques to find the highest accuracy model for seizure

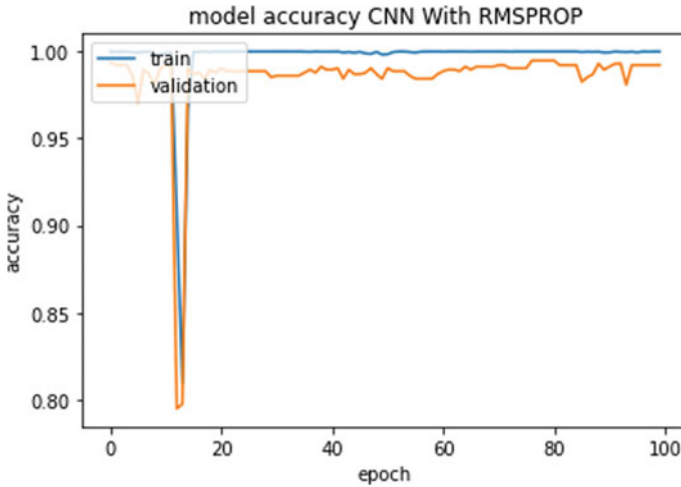


Fig. 12 The accuracy of CNN with RMSprop optimizer

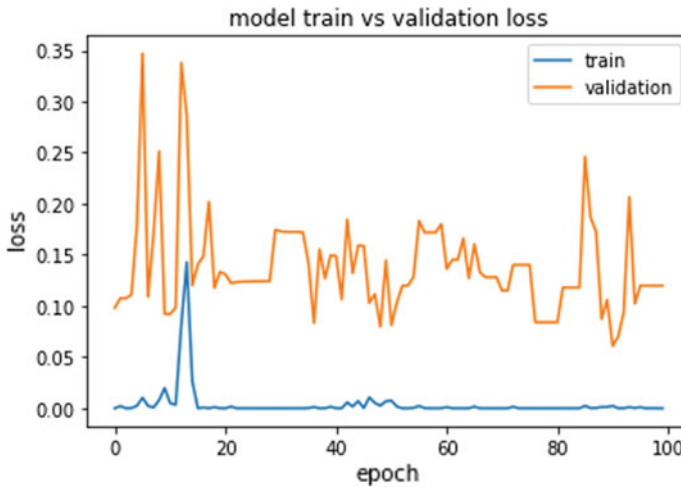


Fig. 13 The accuracy of training and validation of CNN with RMSprop optimizer

detection. In machine learning models, we modified the parameters to reach the highest accuracy. The evaluation of its performance included classifiers SVM, decision tree, RF, QDA, KNN, XGBoost and CatBoost. The highest accuracy performance achieved by using XGBoost classifier is 98%. Then, CatBoost archived 97% accuracy. Tables 2 shows the results: SVM classifier 97.5% and RF 97.4%.

This study applied various deep learning algorithms. The models are trained on the Google cloud using Google Colab and laptop core i7 8th, 6gb Graphics Cards using MPL Classifier, ANN, LSTM, CNN and CNN-LSTM. The fine-tuning was

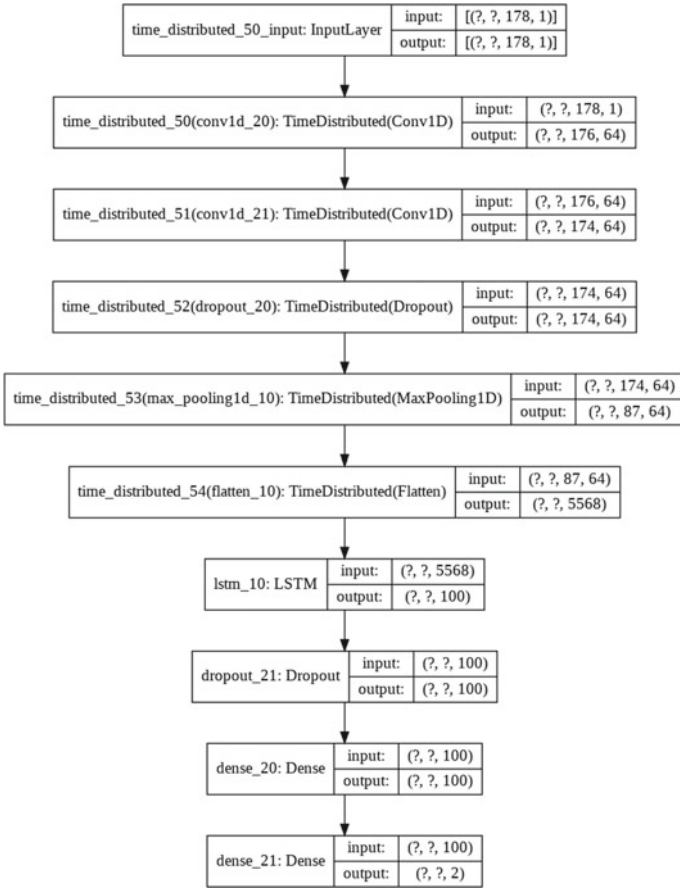


Fig. 14 The architecture of CNN-LSTM

done for each neural network of the models, and hyperparameters changed as well as several optimizers, like Adam, SGD, Adagrad, RMSprop, were applied in our models because of their effect on performance.

Here is a summary of the performance of DL models. 99.2–99.3% is the accuracy achieved in CNN with Conv1D and 64 filters with Adam optimizer and learning rate = 0.001 result as shown in Table 3, whereas as it is 98.5% with Adagrad. The best obtained accuracy using LSTM with RMSprop optimizer is 98.4% as shown in Table 3, which was equal to the results obtained using Adagrad optimizer. The accuracy obtained using ANN with RMSprop was 98.17%, while it is 98% when using MPL classifier as shown in Table 3.

5 Conclusion

Epilepsy is a neurological disorder in the form of sudden changes of behavior because of a temporary change in the brain's electrical function which makes locating the accurate epileptic seizure position challenging because the change spreads over a wide range of cortical surfaces. EEG is the most common technique used in diagnosing epilepsy. In this paper, we used an EEG dataset obtained from 400 people, 200 with epilepsy and 200 normal. The discrete wavelet transform and AR were used to extract the important features from the data obtained.

In our experiments, we used machine learning algorithms, deep neural network for the classification of epileptic seizure using two classes: class 1 for seizure and class 0 for non-seizure. The experiments were conducted to get the most accurate model by optimal tuning parameters, layers, etc. the best model we came up with achieved the accuracy of 99.2% in which CNN Conv1D and Adam optimizer were used. The results surpass those obtained using ML models including LSTM, ANN, MPL.

References

1. Hussein R, Palangi H, Ward R, Wang ZJ (2018) Epileptic seizure detection: a deep learning approach. arXiv Preprint. arXiv1803.09848
2. Santhosh NS, Sinha S, Satishchandra P (2014) Epilepsy: Indian perspective. *Ann Indian Acad Neurol* 17:3. <https://doi.org/10.4103/0972-2327.128643>
3. Mattson RH, Cramer JA, Collins JF (1992) A comparison of valproate with carbamazepine for the treatment of complex partial seizures and secondarily generalized tonic-clonic seizures in adults. *N Engl J Med* 327:765–771. <https://doi.org/10.1056/NEJM199209103271104>
4. Sowndhararajan K, Kim S (2016) Influence of fragrances on human psychophysiological activity: with special reference to human electroencephalographic response. *Sci Pharm* 84:724–751. <https://doi.org/10.3390/SCIPHARM84040724>
5. Ouyang CS, Yang RC, Chiang CT, Wu RC, Lin LC (2020) EEG autoregressive modeling analysis: a diagnostic tool for patients with epilepsy without epileptiform discharges. *Clin Neurophysiol* 131, 1902–1908. <https://doi.org/10.1016/J.CLINPH.2020.04.172>
6. Hindarto H, Sumarno S (2016) Feature extraction of electroencephalography signals using fast fourier transform. *CommIT Commun Inf Technol J* 10:49–52. <https://doi.org/10.21512/COMMIT.V10I2.1548>
7. Hassan AR, Bhuiyan MIH (2017) Automated identification of sleep states from EEG signals by means of ensemble empirical mode decomposition and random under sampling boosting. *Comput Methods Programs Biomed* 140:201–210. <https://doi.org/10.1016/J.CMPB.2016.12.015>
8. Chandaka S, Chatterjee A, Munshi S (2009) Cross-correlation aided support vector machine classifier for classification of EEG signals. *Expert Syst Appl* 36:1329–1336. <https://doi.org/10.1016/J.ESWA.2007.11.017>
9. Guo L, Rivero D, Pazos A (2010) Epileptic seizure detection using multiwavelet transform based approximate entropy and artificial neural networks. *J Neurosci Methods* 193:156–163. <https://doi.org/10.1016/J.JNEUMETH.2010.08.030>
10. Li Y, Yu Z, Chen Y, Yang C, Li Y, Allen Li X, Li B (2020) Automatic seizure detection using fully convolutional nested LSTM. *Int J Neural Syst* 30:2050019. <https://doi.org/10.1142/S0129065720500197>

11. Shamim Hossain M, Amin SU, Alsulaiman M, Muhammad G (2019) Applying deep learning for epilepsy seizure detection and brain mapping visualization. *ACM Trans Multimed Comput Commun Appl* 15:1–17. <https://doi.org/10.1145/3241056>
12. Geng M, Zhou W, Liu G, Li C, Zhang Y (2020) Epileptic seizure detection based on stockwell transform and bidirectional long short-term memory. *IEEE Trans Neural Syst Rehabil Eng* 28:573–580. <https://doi.org/10.1109/TNSRE.2020.2966290>
13. Peker M, Sen B, Delen D (2016) A novel method for automated diagnosis of epilepsy using complex-valued classifiers. *IEEE J Biomed Heal Informat* 20:108–118. <https://doi.org/10.1109/JBHI.2014.2387795>
14. Sarić R, Jokić D, Beganović N, Pokvić LG, Badnjević A (2020) FPGA-based real-time epileptic seizure classification using artificial neural network. *Biomed Signal Process Control* 62:102106. <https://doi.org/10.1016/J.BSPC.2020.102106>
15. Andrzejak RG, Lehnertz K, Mormann F, Rieke C, David P, Elger CE (2001) Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: dependence on recording region and brain state. *Phys Rev E* 64:061907. <https://doi.org/10.1103/PhysRevE.64.061907>
16. García S, Luengo J, Herrera F (2015) Instance selection. In: *Intelligent systems reference library*. pp 195–243. Springer, Cham
17. Ashfaqe JM, Iqbal A (2019) Introduction to support vector machines and kernel methods. Publ. <https://www.Res.net/publication/332370436>
18. Rüping S (2010) SVM classifier estimation from group probabilities. In: *ICML*, pp 911–918
19. Gianey HK, Choudhary R (2017) Comprehensive review on supervised machine learning algorithms. In: *Proceedings—2017 international conference on machine learning and data science, MLDS 2017*, pp 38–43
20. Jiang N, Riley ML (2015) Exploring the utility of the random forest method for forecasting ozone pollution in Sydney. *J Environ Prot Sustain Develop* 1:245–254
21. Murphy KP (2012) *Machine learning: a probabilistic perspective*. MIT press
22. Wu W, Mallet Y, Walczak B, Penninckx W, Massart DL, Heuerding S, Erni F (1996) Comparison of regularized discriminant analysis linear discriminant analysis and quadratic discriminant analysis applied to NIR data. *Anal Chim Acta* 329:257–265
23. Peterson LE (2009) K-nearest neighbor. *Scholarpedia* 4:1883
24. Chirici G, Barabati A, Corona P, Marchetti M, Travaglini D, Maselli F, Bertini R (2008) Non-parametric and parametric methods using satellite images for estimating growing stock volume in alpine and mediterranean forest ecosystems. *Remote Sens Environ* 112:2686–2700
25. Palaniappan R, Sundaraj K, Sundaraj S (2014) A comparative study of the svm and k-nn machine learning algorithms for the diagnosis of respiratory pathologies using pulmonary acoustic signals. *BMC Bioinformatics* 15:223
26. Bühlmann P (2012) Bagging, boosting and ensemble methods BT—handbook of computational statistics: concepts and methods
27. Friedman JH (2002) Stochastic gradient boosting. *Comput Stat Data Anal* 38:367–378
28. Chen T, He T, Benesty M, Khotilovich V, Tang Y, Cho H (2015) Xgboost: extreme gradient boosting. *R Packag Vers* 0.4-2. 1:1–4
29. Postnikov EB, Esmedljaeva DA, Lavrova AI (2020) A CatBoost machine learning for prognosis of pathogen's drug resistance in pulmonary tuberculosis. In: *2020 IEEE 2nd global conference on life sciences and technologies*, pp 86–87, IEEE
30. Dev VA, Eden MR (2019) Gradient boosted decision trees for lithology classification. In: Muñoz SG, Laird CD, Realf MJBTCACE (eds) *Proceedings of the 9 international conference on foundations of computer-aided process design*, pp 113–118. Elsevier
31. Prokhorenkova L, Gusev G, Vorobev A, Dorogush AV, Gulin A (2018) CatBoost: unbiased boosting with categorical features. In: *Proceedings of the 32nd international conference on neural information processing systems*, pp 6639–6649. Curran Associates Inc., Red Hook
32. Buduma N, Locascio N (217) *Fundamentals of deep learning: designing next-generation machine intelligence algorithms*. O'Reilly Media, Inc.

33. Taimoor M, Aijun L (2020) Adaptive strategy for fault detection, isolation and reconstruction of aircraft actuators and sensors. *J Intell Fuzzy Syst* 38:4993–5012. <https://doi.org/10.3233/JIFS-191627>
34. Windeatt T (2006) Accuracy/diversity and ensemble MLP classifier design. *IEEE Trans Neural Networks* 17:1194–1211. <https://doi.org/10.1109/TNN.2006.875979>
35. Basheer IA, Hajmeer M (2000) Artificial neural networks: fundamentals, computing, design, and application. *J Microbiol Methods* 43:3–31
36. Smith S (2013) *Digital signal processing: a practical guide for engineers and scientists*. Elsevier
37. Computer Vision: Introduction of Convolutional Neural Network and VGGNet, <https://medium.datadriveninvestor.com/computer-vision-introduction-of-convolutional-neural-network-and-vggnet-fe4cc0f10b48>. Accessed 21 Feb 2020
38. Huang Z, Xu W, Yu K (2015) Bidirectional LSTM-CRF models for sequence tagging, arXiv preprint [arXiv:1508.01991](https://arxiv.org/abs/1508.01991)
39. Language Modeling with Recurrent Neural Networks (LSTMs), <https://towardsdatascience.com/text-predictor-generating-rap-lyrics-with-recurrent-neural-networks-lstms-c3a1acbbda79>. Accessed 2 Mar 2020
40. Gers FA, Schmidhuber J, Cummins F (2000) Learning to forget: continual prediction with LSTM. *Neural Comput* 12:2451–2471
41. Yao K, Cohn T, Vylomova K, Duh K, Dyer C (2015) Depth-gated LSTM. arXiv Prepr. [arXiv1508.03790](https://arxiv.org/abs/1508.03790)
42. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. *Nature* 521:436–444. <https://doi.org/10.1038/nature14539>
43. LeCun Y, Bengio Y (1998) Convolutional networks for images, speech, and time series. In: *The handbook of brain theory and neural networks*. pp 255–258. MIT Press, Cambridge, MA
44. Krizhevsky A, Sutskever I, Hinton GE (2012) ImageNet classification with deep convolutional neural networks. In: Pereira F, Burges CJC, Bottou L, Weinberger KQ (eds) *Advances in neural information processing systems*. Curran Associates, Inc.
45. Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. In: *International conference on learning representations*
46. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: *2016 IEEE conference on computer vision and pattern recognition (CVPR)*, pp 770–778
47. A Summary of Neural Network Layers, <https://medium.com/machine-learning-for-li/different-convolutional-layers-43dc146f4d0e>. Accessed 06 Mar 2020
48. Zhang H, Weng T-W, Chen P-Y, Hsieh C-J, Daniel L (2018) Efficient neural network robustness certification with general activation functions. In: *Proceedings of the 32nd international conference on neural information processing systems*, pp 4944–4953. Curran Associates Inc., Red Hook, NY
49. Garipov T, Podoprikin D, Novikov A, Vetrov D (2016) Ultimate tensorization: compressing convolutional and fc layers alike. arXiv Prepr. [arXiv1611.03214](https://arxiv.org/abs/1611.03214)
50. Wang B, Li Z, Shi Z, Luo X, Zhu W, Osher SJ (2018) Deep neural nets with interpolating function as output activation. In: *Proceedings of the 32nd international conference on neural information processing systems*, pp 751–761. Curran Associates Inc., Red Hook, NY
51. Wang J, Yu L-C, Lai KR, Zhang X (2016) Dimensional sentiment analysis using a regional CNN-LSTM model. In: *Proceedings of the 54th annual meeting of the association for computational linguistics, vol 2*, pp 225–230. Association for Computational Linguistics, Berlin

An Analytical Approach for Extracting Entities and Their Emotional Tones in Narrative Scenarios



V. Ashwanth and Sneha Sreedevi

Abstract Analyzing novels or stories helps us to review or evaluate certain aspects related to the plot, characters or the theme of the story content. But story analysis done manually is an extremely time-consuming task. The introduction of machine learning algorithms has made analysis of text relatively easier. Relation extraction is a Natural Language Processing (NLP) task that helps us to extract relations of people, places and other entities in a text. Identifying the characters and their intent in stories is an important part for understanding the meaning of the story. The characters are firstly identified, and sentences related to them are selected. Relation extraction is performed by extracting the Subject-Verb-Object (SVO) triples in the selected sentences, and the relations of the extracted entities are represented using a knowledge graph. Performing sentiment analysis on the extracted information of the entities will enable us to understand the emotions regarding the characters or plots involved in the story. Finally, a summary of the story is presented using the latent semantic analysis (LSA) method. These would constitute a good story analyzer that could help us understand the essence of the story represented through a shorter format.

Keywords Natural Language Processing · Story analysis · Latent semantic analysis · Relation extraction · Sentiment analysis · SVO triples

1 Introduction

Literary analysis is the process where we critically examine a work of narrative text in order to understand how the individual parts contribute to the whole. Elements such as the characters, themes, plot, etc. will have to be considered when analyzing a

V. Ashwanth (✉) · S. Sreedevi
Muthoot Institute of Technology and Science, Varikoli 682308, Kerala, India
e-mail: ashwanthv.27@gmail.com

S. Sreedevi
e-mail: snehas@mgits.ac.in

novel or short story. Presently, knowledge representation is mostly hand-authored, a notoriously time-consuming task, which requires expertise in both English literature and storytelling.

Natural Language Processing (NLP) is a branch of artificial intelligence that helps in creating a seamless interaction between computers and humans using natural language. Some of the everyday uses of NLP include spellcheck, autocomplete, spam filters, etc. Multiple NLP tools are available that would help us perform tasks like relation extraction, sentiment analysis and text summarization. NLP can be applied for any text, but in this paper, focus is given on narrative texts such as short- and medium-level stories.

Relation extraction involves the extraction of semantic relationships between the entities in a text. The relations could occur between multiple entities like Person, Organization, Location. Since stories could come under a vast field of themes and relations, classifying the relationships based on any particular types like father-son, brother-sister, would be out of the question. Thus, an unsupervised-semantic based approach would be followed that would extract the relations that are already present in the text.

Sentiment analysis is the process of detecting whether a text is positive, negative or neutral. It helps in understanding the emotional tone involved behind a body of text. In the context of story analysis, a more aspect-based or entity-based sentiment would be required. The sentiments of the main characters in the literature would help us gather an insight on their emotional involvement in the setting of the story. A rule-based approach would help us identify and classify the sentiments of the characters in the story.

Text summarization is the technique of shortening or condensing a long piece of text to a shorter version, reducing the size of the initial text while maintaining the key elements and meaning of the content. There are two different approaches for automatic text summarization: an extractive and an abstractive approach. Extractive summarization will pick up sentences directly from the document based on a scoring mechanism to form a comprehensible summary. Latent semantic analysis (LSA) is an extractive approach that identifies semantically important sentences. It learns latent topics by performing a matrix decomposition on the document-term matrix using singular value decomposition (SVD). The matrices we obtain are word assignment to topics matrix, topic importance matrix and topic distribution across documents matrix. These matrices can be used to further perform computations and choose the most relevant and important sentences for creating a summary.

Story analysis when done manually is a tedious work. Hence, there is a need to automate this task. Using NLP-based techniques, a review of the story can be made such that a user can understand the essence of the story from the shorter version of it.

2 Related Works

Previously, researches have been performed on story analysis and the individual components that could make up an automated story analyzer, but no paper exists that performs all the necessary tasks to make a fully developed analyzer. In the story analyzer developed by Mitri et al. [1], Stanford's CoreNLP is used along with D3 visualization functionality to produce dashboard of interrelated and user-responsive visualizations. They have used Stanford's CoreNLP's NLP services for the preprocessing tasks including tokenizing, parts-of-speech identification, dependency parsing, named entity recognition, coreference resolution and temporal tagging. Subject and object in a sentence are being extracted if a verb is present, and a word-cloud of the result is presented. Google charts and maps are also used for visualizations of the time, date and locations identified in the story. Relation extraction is one of the key components required for an analyzer. A verb-based algorithm developed by Rodrigues et al. [3] that would extract multiple relations in a single sentence from biomedical publications is a good approach to relation extraction. Named Entity Recognition (NER) is applied for extracting entities, and verb-detection algorithm is applied for extracting relevant verbs. The identification is based on two conditions, if there is more than one recognized bio-entity, and if there is at least one verb that is semantically similar to one of the Unified Medical Language System (UMLS) verb list. The algorithm then determines the main verb in the sentence and extracts it. Also, complicated sentence structures, sentence-level conjunctive structures, phrase-level conjunctive structures are considered, thus enhancing the relation triples. Atapattu et al. [2] have extracted triples from lecture notes using the concept of Subject-Verb-Object extraction. When it comes to sentiment analysis, the most representative way to execute sentiment analysis is the lexicon-based method. These methods rely on a pre-defined sentiment lexicon to identify the sentiment polarity of a given document. Valence Aware Dictionary and sEntiment Reasoner (VADER) [6], AFINN, Sentiword, are some of the lexicon-based sentiment analysis tools [4]. A comparative study done on these tools have shown that VADER and AFINN showed good sentiment classification. AFINN scores words unlike VADER that scores an entire text. Thus, for entity-based sentiment analysis, AFINN might be a better choice. A character to character sentiment analysis is being performed in [9]. They were able to use AFINN to capture detailed emotional dynamics on the characters using simple techniques. A popular semantic-based extractive summarization technique is LSA based on SVD [8]. LSA shows that single-value decomposition collapses multiple terms with same semantic and can identify terms with multiple meaning and represent documents in lower-dimensional conceptual space. The main part of text summarization using LSA is in the sentence selection process. Different term matrices along with other sentence selection techniques have to be used in order to extract the main sentences to create the summary. One way is to use topic method, to extract concepts and subconcepts from SVD calculations and are called topics of the input document [7].

3 Problems Identified in Existing Methods

Some problems identify in the existing system during the literature survey are:

1. There is no story analyzer that performs all the necessary tasks such that a user can understand the story from the shorter representations.
2. The SVO triple extraction doesn't consider the existence of multiple subject/object/verb. Also, all the sentences are extracted rather than considering the main sentences with entities.
3. For the LSA-based summarization, the system can be improved by giving importance to sentences that has important entities in it.

4 Methodology

The process of literature analysis using NLP is multi-stage. Initially, text pre-processing has to be performed on the input text. This processed text will be provided as input to a Named Entity Recognizer. The identified entities along with sentence split text will be provided as input to the relation extractor which would provide a knowledge graph representation of the relations identified. Next, sentiment analysis will be performed to identified the entity sentiments. Finally a short summary of the story will be produced.

4.1 Data Collection

Short- to medium-level stories are collected, about 1000–3000 words. These include fictional and non-fictional stories.

4.2 System Architecture

The diagram (Fig. 1) is the system architecture that represents the overall working of the methods involved this approach.

4.3 Components

The individual components are:

- (i) **Pre-processor:** The input story is passed to a pre-processor before performing other tasks. Initially, the contraction words like can't, aren't, etc. are replaced with

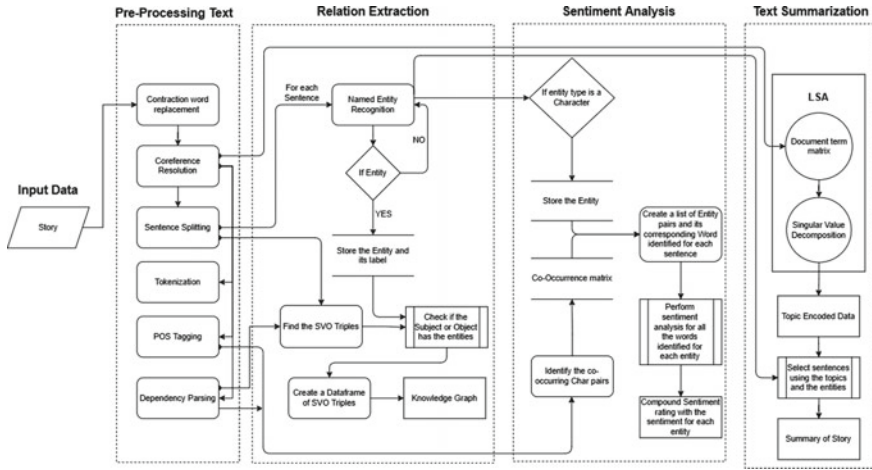


Fig. 1 System architecture

its expansion. This will help us identify the sentiment better. Then coreference resolution is performed to resolve pronouns that are substituted for previously mentioned named entities. This text is passed to sentence splitting, tokenization, POS tagging and dependency parsing processes. Sentence splitting is the process where we convert every sentences into individual tokens, tokenization is the process of converting every word into tokens, POS tagging helps us understand the part of speech of every word in the text and finally the dependency parsing will give us the output of the dependencies of the words in the sentences. Outputs from each of these processes will be used in the upcoming modules. It is accomplished using SpaCy toolkit. The pipeline of the toolkit is shown below (Fig. 2). SpaCy toolkit was specifically used because of its ability to provide fast and accurate pre-processing results. Also the tool helps us easily train and configure its NLP pipeline according to our need.

- (ii) **Named Entity Recognition:** SpaCy tool’s NER is used for entity recognition. The entities recognized by default are PERSON, NORP, FAC, GPE, LOC, PRODUCT, etc. Since stories can include other entities such as animals, the NER is custom-trained with new dataset and labels.
- (iii) **Relation Extraction:** Textacy tool’s SVO functionality is used to identify the Subject-Verb-Object triples from a sentence. It structures triples from a document or sentence through rule-based pattern matching of the annotated tokens. Initially, entities and its types will be recognized and stored in a dictionary. The SVO triples will be identified from each sentence that contains an entity. If the subject or object is an entity that sentence and its relation will be chosen. Existence of multiple objects/subjects/verbs within a sentence is also consider when extracting the triples. Finally, a knowledge graph will be created using the information obtained of the SVO triples.

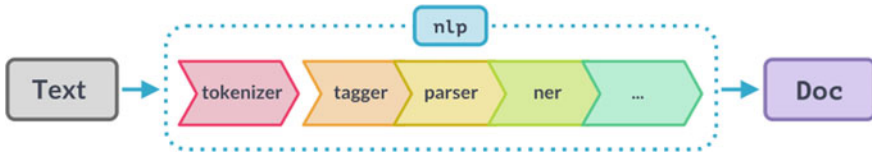


Fig. 2 SpaCy pipeline

- (iv) **Sentiment Analyzer:** The task is to analyze the sentiment of each entity and how they impact other entities in the story. The most important characters, i.e., named entities such as person or animal character, will be identified from the text. And a cooccurrence matrix will be created to identify which of those two important entities occur in a sentence. Simultaneously, the sentiment of the sentences where the characters occur is calculated. AFINN [5], a popular lexicon-based sentiment analyzer tool is used to calculate the sentiment score. The score of that character pair will be used to display how positive or negative that character is affecting the story, and between themselves. This will be represented using a colored knowledge graph, where the brighter the color, the more negative that character bond is.
- (v) **Text Summarizer:** Extractive summarization will be created using the method of LSA and SVD. The input text will be converted into a matrix (M) where the cells are used to represent the importance of the words in sentences. SVD will be used to decompose the original matrix into three individual matrices that consists of (U) document x topics matrix, (Σ) topic x topic matrix and (V^T) terms x topic matrix (Eq. 1)

$$M = U \Sigma V^T \quad (1)$$

Using the information of how important the entities are in the text, and topic method to extract concepts and sub concepts from the SVD calculation, i.e., how important the terms and topics are to the document, sentences will be selected to create the summary.

5 Result Analysis and Discussion

Tests were performed on 30 stories that had 25 fictional stories and 5 non-fictional stories. A sample output obtained on the stories “The rabbit and the Tortoise” and “Akbar and Birbal’s Kingdome” is given below.



Fig. 3 NER output

5.1 Named Entity Recognition

All the named entities were recognized properly by the custom-trained entity recognition model (Fig. 3). The model was trained for 100 iterations with training dataset consisting of 300 sentences.

5.2 Relation Extraction

The relation extraction produced the output as shown (Fig. 4). The initial node represents the entity class of subject, the end node represents the object and the edge label represents the relationship between the two nodes. For example, tortoise is the subject and race is the end object and finished is the relation between both. The algorithm was able to clearly identify the SVO triples and was able to create a relationship knowledge graph.

5.3 Sentiment Analysis

The representation of the negative or positive sentiment between characters is based on color intensity of the line between the nodes. Here, darker the color (e.g.,

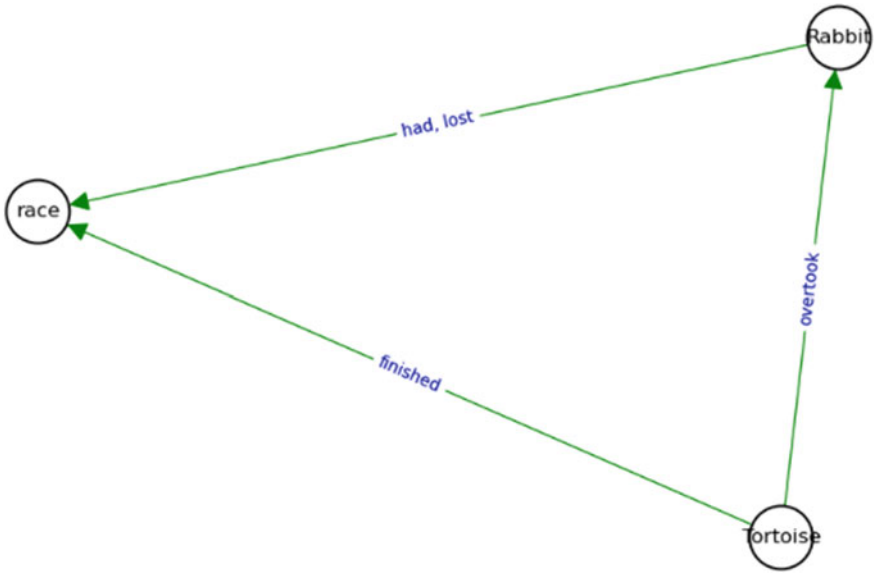
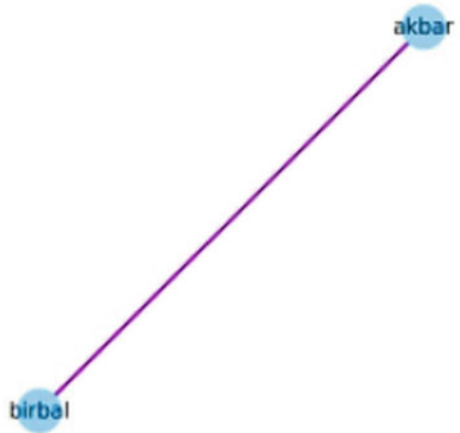


Fig. 4 Relation extraction output

Fig. 5 Sentiment analysis output



red) represents the negative sentiment between the two characters Akbar and Birbal (Fig. 5.). According to the story, Akbar had banished Birbal from his kingdom and there is a feud occurring between the two. Hence, the sentiment analyzer was able to capture the sentiment involved between the characters properly.

5.4 Text Summarization

Using the concept of LSA, the most important and semantically correct sentences were extracted from the text to produce the short summary. Out of 30 sentences of the original text, only 8 important sentences were selected.

Even though the output for the fictional stories were convincing, the non-fictional stories had issues in recognizing the entities since they didn't have enough character entities to perform other NLP tasks. Also, since some stories didn't have enough sentimental words in them, the sentiments displayed were null. Even though the LSA was able to capture most of the important sentences, some major sentences were left out because of its lack of occurrences in the document, as LSA works on the basis of topic importance. The LSA was able to capture most of the important sentences with a rouge score: 83% .

6 Conclusion and Future Work

In this research work, a story analyzer has been implemented that could help a user understand a story through a shorter and condensed version of the same. This is achieved through NLP tasks such as relation extraction, sentiment analysis and text summarization. The methods of SVO triple extraction for relation extraction, lexicon-based entity sentiment analysis and text summarization using LSA are performed to create the short review version of the story. The system of story analyzer presented is uncommon and provides better text summary score than the previous attempts.

Other relation extraction techniques like LSTM, tree kernel-based relation extractions can be used to improve the output of the system. Also, abstractive text summarization can be used instead of the extractive method to provide a human like summary output.

References

1. Michel M (2020) Story analysis using natural language processing and interactive dashboards. *J Comput Inf Syst.* <https://doi.org/10.1080/08874417.2020.1774442>
2. Atapattu T, Falkner, Falkner K, Falkner N (2014) Acquisition of triples of knowledge from lecture notes: a natural language processing approach
3. Rodrigues O, Keppens J, Hao Q (2017) A Verb-based algorithm for multiple-relation extraction from single sentences
4. D'Andrea AF, Grifoni F, Patrizia Guzzo T (2015) Approaches, tools and applications for sentiment analysis implementation. *Int J Comput Appl* 125:26–33. <https://doi.org/10.5120/ijca2015905866>
5. Nielsen F (2011) A new ANEW: evaluation of a word list for sentiment analysis in microblogs. In: *Proceedings of the ESWC2011 workshop on 'making sense of microposts': big things come in small packages.* Volume 718 in CEUR workshop proceedings, 2011 May. Matthew Rowe, Milan Stankovic, Aba-Sah Dadzie, Mariann Hardey, pp 93–98

6. Hutto CJ, Gilbert E (2014) VADER: a parsimonious rule-based model for sentiment analysis of social media text. ICWSM
7. Kherwa P, Bansal P (2017) Latent semantic analysis: an approach to understand semantic of text. In: 2017 international conference on current trends in computer, electrical, electronics and communication (CTCEEC), pp 870–874. <https://doi.org/10.1109/CTCEEC.2017.8455018>
8. Rahimi SR, Mozhdehi AT, Abdolahi M (2017) An overview on extractive text summarization. In: 2017 IEEE 4th international conference on knowledge-based engineering and innovation (KBEI), pp 0054–0062. <https://doi.org/10.1109/KBEI.2017.8324874>
9. Nalisnick ET, Baird HS (2013) Character-to-character sentiment analysis in Shakespeare’s plays. In: ACL 2013—51st annual meeting of the association for computational linguistics, proceedings of the conference, vol 2, pp 479–483

A Simple Divide-and-Conquer Algorithm for Solving an Instance of Planar Convex Hull Problems



Sariah López-Fierro 

Abstract This work presents, analyzes and justifies an alternative of divide-and-conquer algorithm for solving an instance of planar convex hull problem of a set S with a *MergeSort* algorithm. The presented technique allows to improve the execution time for the solution of the problem in $O(n \log n + n)$. It is also proven that k -subproblems would not improve the execution time, unless its implementation in a parallel computation model is considered.

Keywords Convex hull · Algorithm · Divide-and-conquer · QuickSort · Computational geometry

1 Introduction

More than a century ago, in 1893, J. J. Sylvester posed the first documented mathematical question about expected probability in a two-dimensional plane of three random points [1]. He wondered if, for example, each set of three or more points of a set S , must at least determine an ordinary line in the plane (line that passes exactly two points of S).¹

In 1896, H. Minkowski introduced the idea of “extreme points” or vertices of a tetrahedron. Thus, as these points are not within a convex set, if any of these are eliminated, the set would remain as a closed convex bounded in \mathbb{R}^3 being the smallest closed convex of the set from their extreme points [3].

The notions contributed by these authors have been alleged to be the origin of what was later universally known as convex hull (Konvexe Hülle) [1, 3].

¹Almost fifty years later (1944), Gallai T. presented the first solution to this problem. Gallai proposed a point $s_1 \in S$ and a set of parallel lines containing s_1 , with the projection of a new connecting line C . This structure would form angles with all parallel lines, which can allow containing three (or more) points of S , where C is an ordinary line [2].

S. López-Fierro (✉)

Technological Research Department, Soluciones Wandarina S. A., Guayaquil, Ecuador
e-mail: slopezf@wandarina.com

2 Background

The convex hull can be originated by cutting part of a closed convex curve with a straight line, which would generate two points in an n -dimensional Euclidean space. Through this repeated/completed process, it is possible to obtain the largest subset of all possible sets [4].

But to do so, those sets of points must meet five suggested properties:

- First, the set has to be *bounded*, which means that a space is defined within all points.
- Second, the set must be *closed*, containing all the limit points.
- Third, the set must have *internal points*.
- Forth, the set must be *linearly connected*.
- And, fifth, the set must be *supported at its boundary points* [4].

Therefore, the convex hull (CH) of a set S of n points is defined as the smallest intersection of all convex sets, containing the points [5].

To calculate this problem, it can be visualized as a set of independent and random points $S = \{s_1, s_2, \dots, s_n\}$ distributed in a plane or in a space, which requires generating a list of those points that are the vertices or the points that create the hull (h_n) [1, 5]. For a graphical representation, see Fig. 1.

The calculation of the planar convex hull is one of the first problems studied in computational geometry [5]. This work includes an approach to the divide-and-conquer algorithm technique to solve this geometric problem.

It needs to be added that even though the convex hull problem is also addressed in a three-dimensional plane version, this document will only focus on the planar or two-dimensional approach.

3 State of the Art

The complexity of the algorithms is based on the total number of points (n) and the number of points that create the hull (h). Among these algorithms, in computational geometry, the most popular algorithms for calculating the convex hull problem are “Graham scan” and “divide-and-conquer” [6].

This section includes some literature with applications and variations of the latest algorithm on the convex hull problem.

In 1975, M. Shamos and D. Hoey related six problems, such as calculating the proximity of n points in a plane, with a geometric structure called a Voronoi diagram. To determine their hull, they proposed working on polygons in the diagram, dividing the lowest vertex l and the highest vertex h on each. Therefore, since all the polygons are connected, moving “round robin mode” through the lowest vertex l would produce h [7].

Two years later, an adaptive algorithm called CONVEX was introduced to calculate the hull. Its authors proposed dividing the input into two matrices, which will contain the “top” and “bottom sides”. This process continues until its subarrays

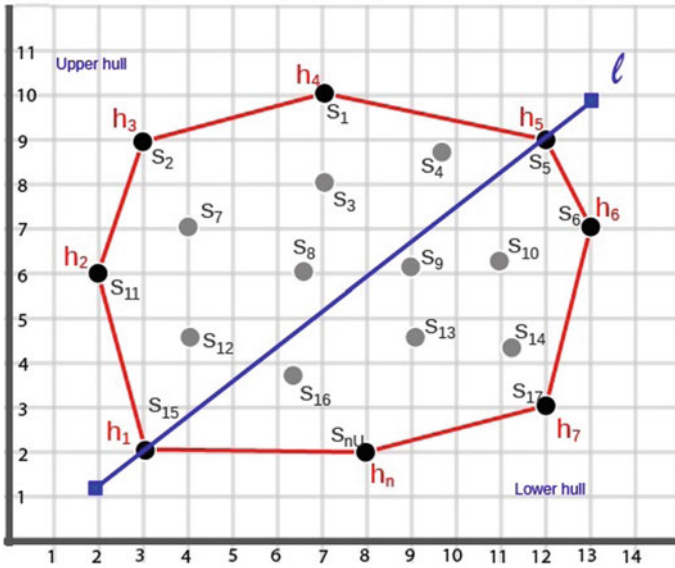


Fig. 1 Representation of a convex hull (h) of a set S , $h \in S$

reach 1 in length. As it is not necessary to partition all the subarrays, this algorithm achieved better speed [8].

J. Bentley and M. Shamos worked on a divide-and-conquer algorithm that received data with the probability distribution of the vertices or h , this “weak assumptions about the underlying probability” allowed to obtain a linear average-case time complexity [9].

D. Rappaport proposed a different approach that includes angles. He worked with a set of disks of different sizes. The set was divided into two equal subsets named P and Q , in which a parallel pair of support lines L_p and L_q were rotated for each set. In each iteration it was determined if the current arc is extreme for both hulls and belongs to $CH(P \cup Q)$ [10].

T. M. Chan contributed with a “simple output sensitive convex hull algorithm”, which managed to run in $O(n \log h)$ time. This was achieved because it preprocessed a certain hull calculated in a previous iteration and kept combining them as the size increased [11].

For image processing applications, X. Zhang and Z. Tang recently proposed to calculate the extreme points of an object in a binary image, dividing it into several regions, then processing the monotonous segments and merging their results. This process showed a low complexity space, after requiring storage only for the vertices or h [12].

Modern divide-and-conquer algorithm approaches involve parallelization alternatives.

S. Näher and D. Schmitt worked with *Quickhull* to independently process sub-problems of the same structure, or *jobs*, that were distributed in parallel processes or threads, managing to assign them to each kernel of CPU [13]. T. Jurkiewicz and P. Danilewski introduced a CUDA algorithm to utilize the processing power of the GPU. This algorithm applies *QuickSort* at points n and solves the problem of convex hull in $O(k \log g)$ [14]. Meanwhile, S. Masnadi and J. LaViola introduced *ConcurrentHull*, a novel alternative based on pruning, which combined pruning, divide-and-conquer and parallel computing techniques. Making the algorithm flexible to be deployed in distributed computing environment. The algorithm allowed to achieve a better performance gain as the size of the input data increases [6].

At the end of November 2020, V. Tereshchenko and S. Chudakov published the notions of a unified algorithmic environment model (*UAEM*) based on divide-and-conquer methods. The procedure involves preprocessing and dividing the initial dataset into right and left *subhulls* that formed a modified balanced binary tree and used a concatenable tail to keep the hull convex, and then merged them into *resolved subtasks*. This approach represented a reduction in time and less use of computational resources to solve the problem of the convex hull [15].

This 2021, A. Sapucaia et al., presented a novel polygon-based integer-programming formulation for improving instances of minimum convex partition problems, which are geometric instances that work with subdivisions or regions of convex polygons [16].

Filtering or preprocessing techniques that consisted of reducing the input set S , thus reducing computational time and memory space, have also been included in modern publications.

In 2019, Ferrada, H. et al., proposed the reduction of interior points in a two-dimensional space, obtaining an algorithm that ran on $O(n \log n)$ [17]. Another approach that achieved a speedup of 12 times faster than the standard Graham scan by grouping and filtering the extreme points into priority queues was exposed by R. Alshamrani et al., in 2020 [18].

Along with the literature presented above, this article includes the presentation of the divide-and-conquer algorithm with the *MergeSort* strategy.

4 Proposed Algorithm

In 1970, Chand et al. proposed an $O(n^2)$ time algorithm that used the geometric properties of S to construct the convex hull [12]. A few years later, Ronald Graham proposed a better solution to calculate CH of a plane with operations at most $\frac{n \log n}{\log 2} + cn$ and a stack for intermediate storage of $O(n)$ [19, 20]. Following his contribution, different proposals have been published that sought to improve Graham's time or space. This section brings together some of these publications.

4.1 Lower Bound Analysis

A comparison-based sorting algorithm operates on the input array by comparing its elements by two, and moving or exchanging them according to the results of

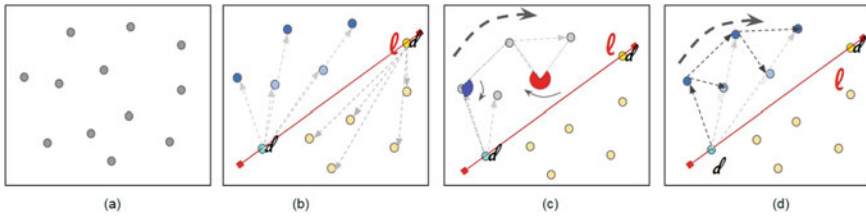


Fig. 2 Graphical representation of running Graham InPlace scan for calculating the upper hull (h) of a set S , $h \in S$

the comparison. MergeSort, QuickSort, HeapSort are popular examples of these algorithms [9].

The lower limit of comparison-based sorting algorithms can be demonstrated using a decision tree model. Thus, there exists a convex hull problem of a set $S = \{s_1, s_2, \dots, s_n\}$ of n points. Through which, when comparing pairs of its points (a.e. is $s_i > s_j$?), it returns its results (“YES” or “NO”), in 1 time step. Therefore, there is an input S of size n such that the algorithm makes $\log_2(n!) = \Omega(n \log n)$ comparisons to sort S [21].

Also, following the notes of A. Blum,² worst case, ordering S means $n!$ different permutations possible. If S is divided into two groups to separate the answers of “YES” on one side and “NO” on the other, each comparison will reduce the size of S at most by a factor of 2, until S is reduce to 1. Therefore, the algorithm must do at least $\log_2(n!)$ comparisons before stopping. Meaning:

$$\log_2(n!) = \log_2(n) + \log_2(n - 1) + \dots + \log_2(2) \tag{1}$$

$$= \Omega(n \log n) \tag{2}$$

4.2 Divide-and-Conquer Algorithm

Divide-and-conquer techniques are widely used because it is easier to work on a problem proportionally. MergeSort is one of their *in-place* algorithms that divides the n size problem into k sizes. And once these subproblems are solved, merge the partial results obtained into a single solution for the whole problem [9].

Based on [19, 20, 22], the following solution is presented for the convex hull problem.

To determine its vertex h , the following procedure is proposed:

1. Sorting the convex hull. By dividing the set into two $\frac{n}{2}$ subsets of approximately the same size. So, let l be an imaginary line that divides the set S from the lower leftmost point to the rightmost point, forming the upper and lower hull, see Fig.

² Avrim Blum is a professor at CMU, part of his notes were used to indicate this reduction <https://www.cs.cmu.edu/~avrim/451f11/lectures/lect0913.pdf>.

- 2, part *a* and *b* for a graphical representation. Hence, to reduce the constants in the algorithm, the classification will be handled through MergeSort [19].
2. Running Graham InPlace Scan [19] for computing the upper hull of the set of S . A parameter d is used in this part for signaling if the algorithm is working on the upper ($d = 1$) or the lower ($d = -1$) hull. If $d = 1$, the points are working in increasing order (left-to-right); in this way calculating the upper hull. Graham's scan uses a stack to allocate all vertices h , see Fig. 2, part *c* and *d* for a graphical representation.
 3. Running Graham InPlace-Hull [19] to calculate the bottom hull of the S set. When $d = -1$, the sort should be done in descending order (from right to left). The vertices h are also allocated on the same stack.

The Algorithm 1 includes the pseudocode used to order and preprocess the points of the set S , thus forming a symbolic closed path by dividing the set into two $\frac{n}{2}$ approximately equal subsets size.

Algorithm 1 InPlace MergeSort (S, n)

```

if  $n \leq 2$  then
  return 0
else
   $m = n/2$ 
   $P = s_1, s_2, \dots, s_m$ 
   $Q = s_{m+1}, s_{m+2}, \dots, s_n$ 
   $S_P = \text{MergeSort}(P, m)$ 
   $S_Q = \text{MergeSort}(Q, n - m)$ 
   $S = \text{Merge}(S_P, S_Q)$ 
end if
return  $\text{Merge}(P, Q)$ 

```

Therefore, starting with the top layer of S to store the vertices obtained clockwise in $S[0], \dots, S[h - 1]$, the pseudocode is provided with the name of Algorithm 2.

Algorithm 2 Graham InPlace Scan (S, n, d) [19, 20]

```

InPlaceMergeSort( $S, n, d$ )
 $h \leftarrow 1$ 
for  $i \leftarrow 1 \dots n - 1$  do
  while  $h \geq 2$  and not right-turn ( $S[h - 2], S[h - 1], S[i]$ ) do
     $h \leftarrow h - 1$ 
  end while
  swap  $S[i] \iff S[h]$ 
   $h \leftarrow h + 1$ 
end for
return  $h$ 

```

Also, to calculate the lower hull of $S[h - 2], \dots, S[n - 1]$, and store the vertices obtained clockwise in $S[0], \dots, S[h + h' - 2]$, Algorithm 3 is presented. Figure 3

Algorithm 3 Graham InPlace-Hull (S, n) [19, 20]

```

h ← GrahamInPlaceScan(S, n, 1)
for i ← 0...h - 2 do
    swap S[i] ⇔ S[i + 1]
end for
h ← GrahamInPlaceScan(S + h - 2, n - h + 2, -1)
return h+h'-2
    
```

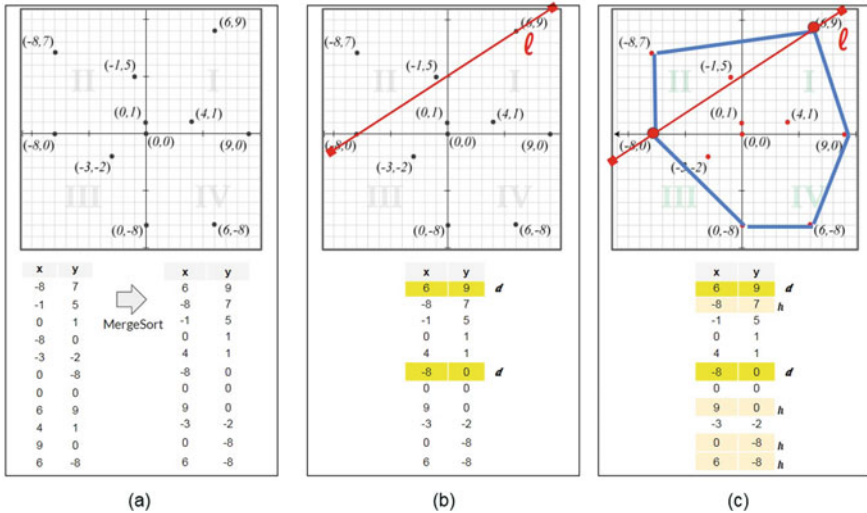


Fig. 3 Graphical representation of an example process after running Graham InPlace scan to calculate the upper hull (h) of a set $S, h \in S$

provides an example of the process presented above. Part *a* portrays how *Mergesort algorithm* arranges a set of random points. Part *b* shows the imaginary line l that divides the set S on the graph; despite the graphical representation, it identifies the points d , which divides the ordering set into almost two equal groups. Part *c* represents the points h , after running Graham InPlace Scan and InPlace-Hull.

Another way to improve runtime is to use a and b as two pointers that can split the array into the vertices of the top and the possible vertices of the lower hull, see Fig. 4. Thus, after executing “Graham InPlace Scan (S, n, d)”, which has assigned all vertices to the beginning of the array, adding as pointers a in $S[h + 1]$ and b in $S[n]$, obtaining the vertices of the lower hull would represent more constant improvements in the execution time [19].

Execution time analysis The complexity of the described convex hull construction algorithm for a static set of points is $O(n \log n)$ with sequential execution. Detailed in the following procedures:

1. Preprocessing: To sort n items through MergeSort, represent $O(n \log n)$ time. And to divide the matrix into $\frac{n}{2}$ requires $O(1)$.

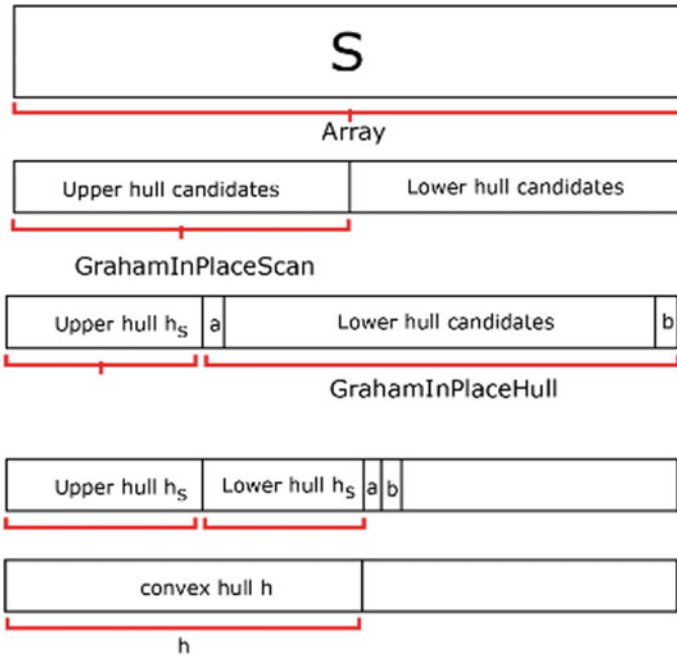


Fig. 4 Representation of how to improve the running time in the convex hull algorithms

2. To determine the upper and lower hulls, through Graham InPlace Scan and Graham InPlace Hull requires $n \log_2 n + O(n)$ comparisons, plus $\frac{3}{2}n \log_2 2 + O(n)$ swaps; and $O(1)$ on additional memory. And at most it requires to pass 2 times the array, except to the founded vertices ($h = p \cup q$), hence taking at most $2n - h$ tests [20].

Thus the complexity of the convex hull algorithm is [10]:

$$T(n) = O(n \log(n))n \log_2 n + O(n) \frac{3}{2}n \log_2 2 + O(n) \tag{3}$$

$$= O(n \log n) + O(n) \tag{4}$$

$$= O(n \log n) \tag{5}$$

k-subproblems analysis What if, instead of partitioning the convex hull problem in $\frac{n}{2}$, it is divided into 3, 4, ..., $\frac{n}{k}$ subproblems? Since the execution is done in sequence, when $k > 3$, it would require an extra step, which means going through all the h , to determine if any vertex is contained in other returned points. Therefore, the time would be the same as stated before, plus h . But for a large n , if calculated in parallel computational models, determining the convex hull in different chains ($\frac{n}{k}$

subproblems) would allow to reduce to $\Theta(k \log k) + O(h)$ in the preprocessing step. Also, as stated above, if k is a constant, each comparison will reduce the size of S at most by a factor of 2, until S is reduced to 1, thus obtaining a

$$\Theta(1).$$

Figure 5 shows an example of how the algorithm works when $k > 2$. Note in part *a*, an imaginary line l would not be possible; instead there would be $k - 1$ lines. In this particular example, there are 4-subproblems. Part *b* lists the points. Part *c* shows them after running the algorithms, which sort the points and divide them into four groups. And part *d* shows the result of the final execution of the algorithms to determine the hull.

4.2.1 Recurrence Equation

The recurrence relationship is seen in Algorithms 2 and 3, or the combined version exposed, thus adapted from [20], the following equation is proposed:

$$h[\pi(i)] = \begin{cases} n \leq 2 & 0 \\ n > 2 & \text{upper hull} \leftarrow \text{GInPlaceScan}(h + 1) \\ n > 2 & \text{lower hull} \leftarrow \text{GInPlaceHull}(h + 1) \end{cases}$$

5 Limitations and Recommendations

Like any other divide-and-conquer algorithm, one of the main advantages of this technique is its focus on the “division” of tasks, thus also allowing its availability to share resources. Consequently, it is argued that the presented procedure for calculating the convex hull can also be run in a parallel environment. However, this approach only considers effectively two main comparisons: the upper hull and the lower hull; therefore, this algorithm might not be efficient for a problem with a large number of points, even though the set S could be organized in k -subproblems. Note that in Fig. 5, part *d*, the point with the coordinates $(14, -4)$ is discarded, although, it is part of the hull.

6 Conclusions

This article has summarized a version to calculate the convex hull problem of a set S with a *MergeSort* algorithm. The application of this technique allows to improve the execution time for the solution of the problem in $O(n \log n + n)$. It has also been argued that the k subproblems would not improve execution time, unless their implementation in a parallel computational model is considered. Since the aforementioned problem has been widely studied in the context of geometric applications, the importance of continuing to investigate in this area is maintained.

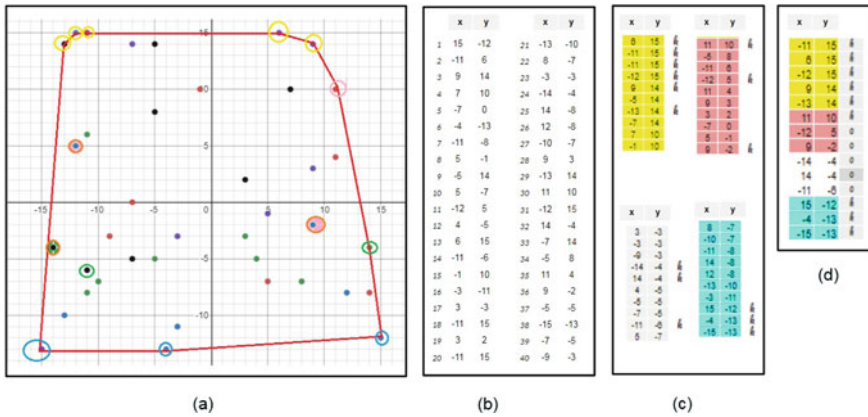


Fig. 5 Graphical representation of an example process for k-subproblems

Acknowledgements Special recognition to the DGIIIP of the Universidad Técnica Federico Santa María, of Valparaíso-Chile for the motivation to work on this research. This work has been supported and funded by the Technological Research Department of the company [Soluciones Wandarina S. A.](#) of Ecuador.

References

1. Efron B (1965) The convex hull of a random set of points. *Biometrika* 52(3–4):331–343
2. Borwein P, Moser William OJ (1990) A survey of sylvester’s problem and its generalizations. *Aequationes Math* 40(1):111–135
3. Fenchel W. Convexity through the ages. In: *Convexity and its applications*. Springer, pp 120–130
4. Dines LL (1938) On convexity. *Ame Math. Month* 45(4):199–209
5. de Berg M, Otfried C, van Kreveld M, Mark O (2008) Delaunay triangulations. In: *Computational geometry*, pp 191–218
6. Masnadi S, LaViola JL (2020) ConcurrentHull: a fast parallel computing approach to the convex hull problem. In: *International symposium on visual computing*. Springer, pp 593–605
7. Shamos MI, Hoey D (1975) Closest-point problems, pp 151–162
8. Eddy WF (1977) Algorithm 523: convex, a new convex hull algorithm for planar sets [z]. *ACM Trans Math Softw (TOMS)* 3(4):411–412
9. Bentley JL, Shamos MI (1977) Divide and conquer for linear expected time. Technical report, Carnegie-Mellon University
10. Rappaport D (1992) A convex hull algorithm for discs, and applications. *Comput Geom* 1(3):171–187
11. Chan TM (1996) Optimal output-sensitive convex hull algorithms in two and three dimensions. *Discr Comput Geometry* 16(4):361–368
12. Zhang X, Tang Z, Yu J, Guo M (2010) A fast convex hull algorithm for binary image. *Informatica* 34(3)
13. Näher S, Schmitt D (2008) A framework for multi-core implementations of divide and conquer algorithms and its application to the convex hull problem. In: *20th Canadian conference on computational geometry*, pp 203–206
14. Jurkiewicz T, Danilewski P (2011) Efficient quicksort and 2d convex hull for cuda, and msimd as a realistic model of massively parallel computations

15. Tereshchenko V, Chudakov S (2020) Application of concatenable queue for parallel computational geometry algorithms, pp 56–62
16. Sapucaia A, de Rezende PJ (2021) de Souza CC (2021) Solving the minimum convex partition of point sets with integer programming. *Comput Geom* 99:101794
17. Ferrada H, Navarro CA, Hitschfeld N (2020) A filtering technique for fast convex hull construction in r2. *J Comput Appl Math* 364:112298
18. Alshamrani R, Alshehri F, Kurdi H (2020) A preprocessing technique for fast convex hull computation. *Proc Comput Sci* 170:317–324
19. Brönnimann H, Iacono J, Katajainen J, Morin P, Morrison J, Toussaint G (2002) In-place planar convex hull algorithms. In: *Latin American symposium on theoretical informatics*. Springer, pp 494–507
20. Brönnimann H, Iacono J, Katajainen J, Morin P, Morrison J, Toussaint G (2004) Space-efficient planar convex hull algorithms. *Theor Comput Sci* 321(1):25–40
21. Lee D-T (1983) On finding the convex hull of a simple polygon. *Int J Comput Inf Sci* 12(2):87–98
22. Graham RL (1972) An efficient algorithm for determining the convex hull of a finite planar set. *Inf Pro Lett* 1:132–133

COVID-19 Pandemic: Review on Emerging Technology Involvement with Cloud Computing



K. Anushka Xavier, S. L. Chetradavee, and N. Jayapandian

Abstract Cloud computing is the latest technology that has a significant influence on everyone's life. During the COVID-19 crisis, cloud computing aids cooperation, communication, and vital Internet services. The pandemic situation made the people switch to online mode. The technology helped to bridge the gap between the work space and personal space. A quick evaluation of cloud computing services to health care is conducted through this study in COVID situation. A short overview on how cloud computing technologies are critical for addressing the current predicament has been held. The paper also discusses distant working of cloud computing in health care. Moreover, cloud infrastructure provides a way to connect with different aid personnel. The patient data can be transferred to the cloud for monitoring, surveillance, and diagnosis. Thus, health care is provided instantaneously to all the individuals. Additionally, the study addresses the privacy and security-related issues with appropriate solutions. The paper also briefs on the different kind of services are provided by different CSPs that are cloud service providers to confront this epidemic. This article primarily focuses on cloud computing technology involvement in COVID, and secondary focus is on other technology like blockchain, drones, machine learning and Internet of things in COVID-19.

Keywords Artificial intelligence · Blockchain · Cloud computing · COVID · Drones · Internet of things · Robots

1 Introduction

COVID-19 is a rapidly spreading contagious disease that has engulfed several countries, resulting in a very seldom global health catastrophe. As reported by World

K. Anushka Xavier · S. L. Chetradavee · N. Jayapandian (✉)
Department of Computer Science and Engineering, Christ University, Kengeri Campus,
Bangalore, India
e-mail: jayapandian.n@christuniversity.in

K. Anushka Xavier
e-mail: anushka.xavier@mtech.christuniversity.in

Health Organization (WHO), the infectious disease has infected 215 countries around the world as well as is spreading rapidly across other territories. As the cases are outnumbered, the authorities and hospitals do not have enough time to adapt with the new circumstance. This calls for the deployment of several advanced technology to tackle the situation. The coronavirus pandemic of 2019 (COVID-19) has demonstrated that the major hurdles and hazards to mankind require coordinated judgment and unified solutions. Shifting the healthcare sector to a cloud-based computing system will certainly be a smart approach to combat this menace [1]. There are several parameters to deal with this cloud computing platform that is named as flexible, reliable, and scalable infrastructure [2]. The COVID-19 scenario, the appealing common purpose, and informational demand have created a significant impetus for cloud-based solutions to be adopted and scaled up quickly. The healthcare sector is constantly changing, and the forthcoming healthcare system is expected to be data-driven. The enterprise can leverage to greater heights using cloud computing. This innovative technology has the potential to help healthcare providers communicate, cooperate, and organize more effectively. With this technology, it is possible to access computing infrastructure and capabilities at any time from any location. It facilitates on-demand network access to computer resources that are frequently offered by a third party and require only minor administration [3]. Those resources include storage space, servers, software, analytics, applications, and intelligence over the Internet [4]. This framework allows the customers to take advantage of efficient, diverse, and efficient services and also frees them from upkeep. Altogether, patients will receive quality treatment as a result of up-to-the-minute health records and ongoing interactions between medical providers. Cloud computing technology has helped humankind during COVID-19 from a different perspective. The Australian government has introduced an app to provide real-time insights and recommendations on the COVID-19 pandemic. Also, the cloud helps to retain COVID-19-related datasets, which allows the health researchers to carry out the experiments [5]. This article is discussing insight of health sector with cloud computing involvement in the COVID-19 pandemic. This article provides a succinct but comprehensive overview of current cloud research, as well as new research avenues and outstanding challenges. This article is mainly focused on cloud computing technology involvement in pandemic situation. Second section of this article is given higher importance of literature review in different technology aspects in cloud computing. Cloud computing technology to tackle the COVID-19 pandemic is explained in Sect. 3. Cloud computing service providers are discussed in Sect. 4. Section 5 details the recent technology involvement in COVID-19. The research challenges of cloud computing are discussed in Sect. 6. The overall conclusion part is deliberated in Sect. 7.

2 Literature Review

Leveraging the use of cloud computing and new technologies to combat COVID-19 could be extremely beneficial. This research proves that the cloud computing has had

a substantial impact on major sector like health care in our life. In the current circumstances, hospital staff is working tirelessly to assist the afflicted patients. Our research aids to bring out the emerging technologies to this scenario. The study significance the importance of using cloud technology in the healthcare sector [6]. The paper stated that cloud computing model offers a transparent, economical, and interactive framework which utilizes cutting-edge technology like smartphones, smart cards, robots, and sensors [7]. The study also provides a clear-cut on their inherent limitations and implementation hurdles, such as security, legal and ethical issue, along with workforce and organizational issues. The cloud-based healthcare practices are used in various fields. A cloud-based model is constructed using IoT to keep track of the patients [8]. Jin and Chen in their study focused on the necessity to adopt cloud-based telemedicine have also brought a multitude of challenges [9]. The research also emphasizes on the development and deployment of different telemedicine applications. Javaid et al. have discussed various technological tools and their application in addressing COVID-19 pandemic [10]. The paper has discussed on Industry 4.0's innovation strategies to monitor and control the pandemic. Singh et al. in his paper examined how to use cloud computing to combat COVID-19 pandemic [11]. The paper also focuses on the latest technologies and the advanced infrastructure utilized to withstand the catastrophe. The author's study briefs on the remote working, and vivid cloud applications are used in this scenario.

As the COVID-19 pandemic progresses, technological innovations and resources are expanding to halt the spread of the disease. Cloud computing technology emerges to provide complete scalable and flexible services to reduce the impact. It offers online access to treat patients anytime at anywhere. Newly infected patients can be tracked, detected, and monitored using this technique. Applications such as artificial intelligence, drones, 3D printing, and machine learning use cloud computing technology to provide a productive and creative environment.

3 Cloud Computing in COVID

As the world moves toward digitization, one of the most pressing concerns is to manoeuvre the healthcare data more systematically and efficiently. Due to the advancement in IT technology, it is now possible to retain and monitor personal data in digital format. Cloud computing has risen to prominence as a key component of healthcare IT solutions. The concept of remote monitoring was originated back in 1950s when a few healthcare organizations and medical institutions began to experiment with phone-based information sharing. New technologies like video conferencing and telemedicine software have evolved providing better options in telemedicine health care. Telemedicine is among the most comprehensive information and communication technologies (ICT) facilities to provide healthcare services readily. Its services include distant vital sign surveillance, patient consultations through videoconferencing, nurse contact centers, and clinical data transmission and storage. Recent developments and the widespread adoption of wireless mobile technology have

fueled a surge in interest in telemedicine as a means of delivering health care. Traditional telemedicine systems are built with a fixed set of specs and criteria for a single medical provider or organization. As a result, we need new and inventive approaches to make telemedicine more accessible and cheap, utilizing cutting-edge technology like cloud computing. Cloud computing is causing a paradigm transformation of business structures. Its goal is to give users with more modular and reliable services. Cloud computing, in particular, offers on-demand solutions, flexibility, extensive network connectivity, and information exchange [12]. The Web enables cloud storage, which the physician and the patients may easily reach from anywhere in the globe owing to cloud computing. The fundamental goal of cloud computing is to enter information of the patient into the system, making it accessible to the patient or the doctor whenever they need it. Data may be put into the cloud database over the Internet from medical centers or authorized data centers. For example, for consultation on a distant platform, the doctor and patient can use a cloud service to communicate with one another [13]. The doctor establishes a connection with the patient, enabling audio and video transmission in real time. The cloud services such as computing, networking, storage could be used to provide a greater help to facilitate the consulting process.

Table 1 deliberates different cloud platform service tools working in cloud database storage system. There are three main models of cloud computing, namely IaaS, SaaS, and PaaS. IaaS stands for “infrastructure as a service,” which charges customers for their storage use, virtualization services, and for networking capabilities. PaaS stands for “platform as a service,” and it is both the combination of hardware and software. The combination of hardware and software could be used to access the cloud network. Software as a service (SaaS) is a part of software application; in general, it is used to access remote software with the help of Internet. These three models contribute equally to the cloud infrastructure. The cloud computing notion can be designed and deployed to address and combat the current COVID-19 pandemic concerns. Cloud resources may be used in a variety of ways, including data keeping and exchange, maintaining medical records and so on. Information storage and clouding provides service to medical staff and COVID-19 patients throughout the first servicing phase. These cloud data can be accessed by doctors, who can then employ them for diagnosis, medication, and counseling. The software as a service acts as an electronically deployed cloud in which the required records could be retrieved for their stated function. The IaaS service could be utilized to provide compute and

Table 1 Significant elements of cloud computing

Cloud storage	Cloud platform	Cloud infrastructure	Cloud application
<ul style="list-style-type: none"> – Network attached storage-Niravanix, cloudNAS – Database-Amaozon DynamoDB, HANA, EnterpriseDB, Rackspace 	<ul style="list-style-type: none"> – Web apps framework-ruby on Rails, Django, ASP.net – Web hosting 	<ul style="list-style-type: none"> – Compute-Amaozon, IBM, Google – Grid computing-sungrid 	<ul style="list-style-type: none"> – Web application-FB, Instagram, Twitter – SaaS-Salesforce.com

larger storage for aid personnel. Moreover, cloud provides additional process-based services in context of health and medical care. COVID-19 pandemic has taken a big leap in every individual's life. It has changed the whole scenario of built-in world. The lockdown and mask incorporated phases pushed the people to reside in their own homes to prevent the catastrophe. eHealth became the need of the hour. Cloud comes to this picture to help the patients to keep in touch with the aid personnel for remote monitoring and diagnosis. In any difficult scenario, an individual in a faraway location can ease and benefit from the theories and concepts proposed by cloud services. In such circumstances, medical personnel can exchange instructions, details, assessment outlooks, and so on, and patients can benefit by simply accessing those shared content. The patients can access the EHRs via cloud service portals available in Internet. In the continuing COVID-19 pandemic, this will eventually make enhanced patient care service feasible.

4 Service Provides in Cloud Computing

Cloud service providers (CSPs) are providers who lend cloud solutions to their customers that are continuously deployed based on the users demand [14]. There are different service providers such as Microsoft, Amazon, and Google. The CSPs are working with physicians, global health groups, regulatory agencies, and health science companies all across the world to help them cope with COVID-19's impacts. A brief review on their services during the COVID-19 crises is listed in Table 2. Cloud computing has been widely used in a variety of industrial services and infrastructure. It also provides a plethora of services in the medical domain by providing flexible, enhanced features to patients, medical workers, and others. There are a wide variety of applications developed and deployed to provide better assistance through cloud computing. As a result, the suggested techniques could be very useful in combating the persisting COVID-19 pandemic. The applications in this scenario include online storage, spreadsheets, online diagnosis and counseling facilities, spreadsheets, patient's data safety, and so on [15]. Different e-commerce software is employed to provide solutions to confront the pandemic. These applications examine different possibilities to be explored without a great deal of money to upfront. The guiding software helps the users to reach anew and remote locations faster. The cloud-enabled word processors and spreadsheets allow to maintain, edit, and analyze data to procure accurate results. COVID-19 patients can use cloud solutions to view their electronic health records (EHRs) on their smartphones over the Internet [16]. This technology provides remote access service to patients.

Table 2 Services provided by cloud service providers (CSPs)

Amazon Web Service (AWS)	Google Cloud	Microsoft Cloud
<ul style="list-style-type: none"> - COVID-19 HPC consortium-offers advance support and technical expertise to carry out studies on detection, management and development on vaccination projects using AWS service - AWS diagnostic development initiative-supports speedy and precise patient testing for COVID-19, as well as other diagnostic technologies, in order to prevent future breakouts - COVID-19 search-COVID-19 search is a revolutionary machine learning-powered search engine that allows academics to search multiples of research articles and materials 	<ul style="list-style-type: none"> - Virtual care and telehealth-Remote assistance is provided to patients by analyzing, diagnosing, monitoring patient’s health, and scheduling virtual visits - Healthcare data interoperability-maintain patient confidentiality while concealing and interoperating data for COVID-19-related research and population health - Remote work solutions-integrating medicine and health science personnel with the patients through Chrome OS, Google Workspace, and chrome devices 	<ul style="list-style-type: none"> - Enhance patient engagement-providing secure platforms and mobile solutions for patients to engage directly with healthcare providers and streamline mundane chores, thus achieving better quality of treatment and care efficiency - Empower health team collaboration-offers advanced technologies to succor patients to identify their impairments and to monitor their chronic conditions - Increase scientific and operational data insights-Microsoft Cloud for healthcare enables individuals, healthcare professionals, and health plans to electronically access, communicate, and use electronic health information

5 Recent Technology Involvement in COVID

Digitalization has transformed many spheres of life. This advanced technology delivers a wide range of integrated services to health care to combat the COVID-19 outbreak. As the highly contagious (COVID-19) epidemic grows, advancements and initiatives are proliferating in an attempt to keep the situation under control. In an attempt to curb the spread of the disease, treat patients, and relieve pressure, technological applications and initiatives are expanding. Along with cloud computing, other emerging technologies stood forward to combat the impact of pandemic. They formulate innovative and efficient way to tackle the crisis. Technology and data help in rapid action to monitor the pandemic’s spread, promoting healthcare advancements and aids in the diagnosis and management of the disease. This review highlights the essential features and significance of the technologies now in use in the fight against the coronavirus pandemic. It sets out a framework for using digital technologies in pandemic preparation, monitoring, quarantine, testing, contact tracing, and health care.

5.1 Artificial Intelligence

Artificial intelligence has primarily been used to identify whether individuals are affected by the virus and monitor differences in physical temperature in real-time system. Other AI applications can use drones to transport medical supplies, sterilize hospital rooms, and search approved prescription databases for drugs [17]. The AI programmes are efficient to mine social media to identify the false news about the disease. The performance of these applications will be determined by both their technical capabilities and how human controllers, such as AI developers, interact with them. The WHO has worked with Google, Facebook, TikTok, and Twitter to examine and disclose fake facts on COVID-19 [18]. Artificial intelligence along with machine learning is used in digital epidemiological surveillance and rapid case identification.

5.2 Drones

Drones are being used to oversee containment procedures, enable air transmission, undertake aerial thermal sensing, spray disinfectant, data analysis, and carry medical resources in affected areas during the Covid-19 pandemic. Along with the technical ability, the key advantage of using drones is to help human to reduce direct exposure to the virus. The Italian Civil Aviation Authority (ENAC) has revealed that drones have been approved for use by local authorities in Italy to monitor civilian movements during the coronavirus outbreak. DJI, a Chinese company, created drones with the ability to sanitize 100 m in an hour. These drones are ideal for reducing infection and preventing diseases among healthcare professionals. The University of South Australia Researchers is working on more innovative and improved drones named as “pandemic drone”. It conducts more advanced tasks like monitoring heart and respiratory rate, temperature count and recognizing persons who are sneezing and coughing in crowds [19].

5.3 Robots

Robots for COVID-19 cover a broad array of possibilities in medical settings. In most part of the world, robots are being sent to assist in the care of patients and healthcare workers [20]. They are capable of disinfecting the virus affected places such as surgical centers, isolation wards, medical centers, and intensive units by distributing UV light that may quickly wipe away microorganisms. The robots also send out messages and inform people about social segregation and other steps that can be done to prevent the spread of the virus. UBTECH Robotics, in China, has developed a squad of robots named Droid Team to fight against COVID-19. The temperature rate will be

taken by the equipment with thermal cameras mounted atop their heads. CloudMinds supplied six different types of robots to Chinese hospitals that can help with security, surveillance, sanitation, and distribution. Invento Robotics, located in Bangalore, has developed three robots that can handle everything from sanitizing surfaces to addressing patient inquiries and facilitating video consultations with doctors. Police in China have started deploying robots to guard and oversee toll gates, monitor mask usage, and record body temperatures with infrared thermometers. Same system is introduced in Shanghai to spread awareness among the public.

5.4 *Internet of Things (IoT)*

Many projects have been developed and initiated to use IoT-enabled devices to track impacted individuals, maintain surveillance, and monitor temperature. Pulse oximeter and smart thermometers were introduced to monitor health status of the patients and alert medical unit or family members if anything happens. Smart sensors and gadgets are developed for surveillance and contact tracing. The Brazilian government established a COVID-19 hotline, which allows individuals who are experiencing symptoms to get a COVID-19 test. MIT researchers have developed a method for incorporating electrical sensors into stretchable fabrics, allowing them to create shirts or other articles of clothing that can monitor vital signals such as temperature, respiration, and pulse. IoT plays a crucial role in remote patient monitoring, telemedicine, and other important healthcare domains.

5.5 *Blockchain*

The use of blockchain technology can increase diagnostic performance and clinical outcomes, as well as monitor drug supply chains and medical supplies. The development of blockchain has added extra value to the medical emergency when compared to standard surveillance systems. It enhances regulatory compliance, resilience, consistency, and reliability of the existing system. The large amounts of real-time incoming data are published. This application helps to securely manage health data, ensuring interoperability without jeopardizing integrity or patient confidentiality. Blockchain helps in data tracking by providing a tracker which has a dedicated dashboard to track illnesses. The Algorand Foundation, a Singapore-based blockchain corporation, recently released an app called IReport-Covid that allows symptomatic and non-symptomatic users to anonymously submit any information they want about the virus by filling out a survey. Blockchain technology makes sure that data integrity, transparency, quality, and performance are achieved by the system. Table 3 summarizes each digital technology, description, and functionality to tackle the issue.

Table 3 Digital technology, description, and functionality

Digital technology/tool	Description	Functions
Machine learning	Machine learning deals with algorithms and models that is capable of prediction. It learns from the patterns present in the data and draws conclusion with minimum human intervention	Real-time monitoring of disease aids in clinical decision making, testing, and risk assessment, virtual syndromic surveillance and Web-based epidemic advanced analytics, rapid case detection
Big data	This system can hold a large number of data related to virus-infected patients. This technique lays the groundwork for a more rapid and near-real-time examination of decision making	Real-time data collection and tracking provide the most up-to-date information to scientists, hospitals, epidemiologists, and governments
Biosensors	The translation of a biological signal into an electrical signal is done using biosensors. Thermal, optical, piezoelectric, and electrochemical biosensors are some of the most common types of biosensors	Early detection and surveillance of COVID-19. Real-time recording of respiration rate, ECG, and temperature

6 Research Challenges

There is a huge potential for the healthcare sector to grow with the help of cloud computing but it also possess many challenges. Cloud computing applications must meet a number of security requirements in order to increase trust in this novel system. The privacy and security problems in the eHealth system are not limited to adhering to the confidentiality, integrity, and availability (CIA) security framework. It must look into different aspects such as access control, authenticity, anonymity, audit, and so on. The limitations of confidentiality provided while using a cloud-based solution are a pressing concern. In a healthcare application, before accessing the patient data, the system must have an integrity and verification feature check using a checksum or hash. The program must indicate an exception and exit without processing the data if the integrity check fails. The availability of data is a vital and frequently overlooked feature of the eHealth system. It refers to the capacity to function adequately even when some entities disobey and the ability to maintain services even if a data breach occurs. Some standards and regulations must be followed by cloud service providers in order to alleviate security and privacy issues. Cloud service providers should keep track of all data processing activities to remain compliant. They should take the necessary individual and organizational precautions. Privacy protection indicates that service providers should design their operations to respect client privacy, follow applicable laws, and keep track of what personal data they have.

7 Conclusion

Cloud computing services are flexible, scalable, and cost-effective and could be extended across a multitude of sectors. These studies help to find the better computing environment during pandemic COVID situation, so that a large number of people can be benefited. A detailed study on how cloud computing technology can be utilized to combat the pandemic is conducted. Virtual computing solutions are presented, in which the cloud resources astoundingly benefited COVID-19 sufferers for reaching healthcare professionals, seeking guidance from remote locations, treatment recommendations, and so on. The secondary aim of this study is to discuss various cloud service provider details with comparisons. Furthermore, the cloud-based application provides a significant advantage in defending and combating the current COVID-19 epidemic by improving healthcare personnel's efficiency and performance. Moreover, the challenges faced by this technology are also discussed with the appropriate solutions. Thus, the paper proves that the introduction of advanced technologies like cloud computing can mitigate the catastrophe level, and it is the need of the hour.

References

1. Manavi SY, Nekkanti V, Choudhary RS, Jayapandian N (2020) Review on emerging internet of things technologies to fight the COVID-19. In: 2020 fifth international conference on research in computational intelligence and communication networks (ICRCICN), pp 202–208. IEEE
2. Darwish A, Hassanien AE, Elhoseny M, Sangaiah AK, Muhammad K (2019) The impact of the hybrid platform of internet of things and cloud computing on healthcare systems: opportunities, challenges, and open problems. *J Ambient Intell Humanized Comput* 10(10):4151–4166
3. Sanabria-Russo L, Serra J, Pubill D, Verikoukis C (2020) CURATE: on-demand orchestration of services for health emergencies prediction and mitigation. *IEEE J Sel Areas Commun* 39(2):438–445
4. Adi E, Anwar A, Baig Z, Zeadally S (2020) Machine learning and data analytics for the IoT. *Neural Comput Appl* 32(20):16205–16233
5. Ziyad RA, Anbar M, Mahinderjit Singh M, Leau Y-B, Al-Sai ZA, Abu Alhayja'a S (2021) Impact of coronavirus pandemic crisis on technologies and cloud computing applications. *J Electron Sci Technol* 19(1). 100059, ISSN 1674-862X, <https://doi.org/10.1016/j.jnlest.2020.100059>
6. Mohammed F, Alzahrani AI, Alfarraj O, Ibrahim O (2017) Cloud computing fitness for e-government implementation: importance-performance analysis. *IEEE Access* 6:1236–1248
7. Budd J, Miller BS, Manning EM, Lampos V, Zhuang M, Edelstein M, McKendry RA, et al (2020) Digital technologies in the public-health response to COVID-19. *Nature Med* 26(8):1183–1192
8. Krishna K, Karumuri N, Christopher C, Jayapandian N (2021) Research challenges in self-driving vehicle by using internet of things (IoT). In: 2021 5th international conference on intelligent computing and control systems (ICICCS), pp 423–427. IEEE
9. Jin Z, Chen Y (2015) telemedicine in the cloud era: prospects and challenges. *IEEE Pervasive Comput* 14(1):54–61
10. Javaid M, Haleem A, Vaishya R, Bahl S, Suman R, Vaish A (2020) Industry 4.0 technologies and their applications in fighting COVID-19 pandemic. *Diab Metab Syndr Clin Res Rev* 14(4):419–422

11. Singh RP, Haleem A, Javaid M, Kataria R, Singhal S (2021) Cloud computing in solving problems of COVID-19 pandemic. *J Indus Integr Manage*
12. Kasyap VB, Jayapandian N (2021) The world of communication and computing platform in research perspective: opportunities and challenges. In: 2021 3rd international conference on signal processing and communication (ICPSC), pp 289–293. IEEE
13. Yang CT, Liu JC, Chen ST, Lu HW (2017) Implementation of a big data accessing and processing platform for medical records in cloud. *J Med Syst* 41(10):1–28
14. Sagar GR, Jayapandian N (2020) Internet of things: service-oriented architecture opportunities and challenges. In: *Inventive communication and computational technologies*, pp 71–78. Springer
15. Higgins JP (2016) Smartphone applications for patients' health and fitness. *Am J Med* 129(1):11–19
16. Kruse CS, Smith B, Vanderlinden H, Nealand A (2017) Security techniques for the electronic health records. *J Med Syst* 41(8):1–9
17. Natarajan J (2020) Cyber secure man-in-the-middle attack intrusion detection using machine learning algorithms. In: *AI and big data's potential for disruptive innovation*, pp 291–316. IGI global
18. Ittefaq M, Hussain SA, Fatima M (2020) COVID-19 and social-politics of medical misinformation on social media in Pakistan. *Media Asia* 47(1–2):75–80
19. Jayapandian N (2019) Cloud enabled smart firefighting drone using internet of things. In: 2019 international conference on smart systems and inventive technology (ICSSIT), pp 1079–1083. IEEE
20. Di Lallo A, Murphy R, Krieger A, Zhu J, Taylor RH, Su H (2021) Medical robots for infectious diseases: lessons and challenges from the COVID-19 pandemic. *IEEE Robot Autom Mag* 28(1):18–27

An Empirical Examination on Forecasting VN30 Short-Term Uptrend Stocks Using LSTM along with the Ichimoku Cloud Trading Strategy



Pham Ngoc Hai, Hoang Trung Hieu, and Phan Duy Hung

Abstract Stock market forecasting is a highly difficult time-series problem due to its extreme volatility and dynamic. This paper proposes a long short-term memory (LSTM) model that predicts the probability to outperform the market of all of the VN30-Index constituents by using historical price changes along with features calculated from the Ichimoku Cloud trading strategy. After acquiring the proposed model's outputs, we buy three stocks with the best probability to sell them ten days later. We then reinvest the money on the next day using the same strategy. The yearly returns of the above trading scheme are used as the empirical results. This study is conducted in a period of 9 years—from the VN30-Index's establishment in 2012 to the end of 2020. On average, the adoption of the Ichimoku Cloud features in the LSTM model makes our trading strategy go from an annual loss of 2.86% to a profit of 14.29%.

Keywords VN index · Stock market · Short-term uptrend forecasting · Ichimoku Cloud

1 Introduction

Precise prediction of the stock market may generate an enormous amount of profit. As a result, this topic piques a great deal of interest from both financial and scientific researchers. At first, traditional statistical approaches, such as moving averages and relative strength index, were struggling to correctly capture the nonstationary, nonlinear and high-noise stock market because of their linear nature. Nowadays, researchers have started to explore more sophisticated approaches that might be

P. N. Hai · H. T. Hieu · P. D. Hung (✉)
Computer Science Department, FPT University, Hanoi, Vietnam
e-mail: hungpd2@fe.edu.vn

P. N. Hai
e-mail: haipnhe141371@fpt.edu.vn

H. T. Hieu
e-mail: hieuhthe141638@fpt.edu.vn

more compatible with the chaotic stock market, namely machine and deep-learning architectures, and they have found promising results.

There are many different applications of various models and architectures in order to predict the stock market. One of the most common approaches is using historical prices as inputs and then generating either the actual price or price trends (up or down) as outputs. Huynh et al. [1] achieve nearly 60% and over 65% accuracy for the price trends of Standard and Poor's (S&P) 500 index and individual stock, respectively. Moreover, Krauss et al. [2] create a profitable trading strategy by combining deep neural networks, gradient-boosted trees and random forests to predict stocks' trends. They attain an average daily return of 0.45% prior to transaction costs. Ghosh et al. [3] employ Krauss et al. [2]'s framework with improvisation in terms of features, and they are able to reach an average daily return of 0.64% using LSTM networks. On the other hand, some researchers try to tackle this topic from unorthodox angles such as Makrehchi et al. [4]. They use labeled social media text as inputs, and the trading strategies based on their model are able to outperform S&P 500 index by 20%. In addition, Oncharoen et al. [5] introduce a risk and reward function in their loss function, and their proposed model is more effective than traditional trading strategies. Recently, researchers have introduced more novel, complex and comprehensive approaches in regard to stock price forecasting. Deepika and Bhat use the Kalman filter to smoothen out the noise and filter out abnormal incidents in the data [6]. The Kalman filter improves the accuracy of the accelerated gradient long short-term memory from 57.53% to 90.42%. Additionally, Senapati, Das, and Mishra propose a highly advanced hybridization of Adaline neural network and modified particle swarm optimization [7]. Sharaf and colleagues conduct extensive empirical analysis on many learning models [8]. They find that the CNN model outperformed the other models based on evaluation metrics. Their proposed scheme edges out all the juxtaposed schemes in terms of results. Chen et al. are the first to extract stock intrinsic properties from mutual funds' portfolios [9]. They then use the extracted properties to generate the dynamic correlation between the stock and the market. In Vietnam, researchers have also started to show interest in using deep learning to predict the local stock market [10–13].

The Ichimoku Cloud is a relevant signal indicator in the world of trading, and its potency has been investigated by researchers. Gurrib and Elshareif [14] construct an automated trading model based on the Ichimoku Cloud in a narrow sector (S&P 1500 Composite Energy Index). Their empirical results indicate that the Ichimoku Cloud is able to yield profit even in a declining market. Compare to the previous study, Lim et al. [15] tap into a more expansive, including stocks from both Japan and the USA. They conclude that the Ichimoku Cloud can generate profitable signals across both markets. However, not all studies find that the Ichimoku Cloud is a reliably profitable indicator, as [16] is not able to create a consistently profitable model.

Although it is evident that the studies above contribute greatly to this field, there are certain limitations. For example, the Ichimoku Cloud, for the most part, is only used as a trading strategy instead of inputs in deep learning. At the same time, most studies conducted in Vietnam focus solely on prediction accuracy without practical application. Our paper's contribution lies in three aspects. Firstly, we demonstrate

how the Ichimoku Cloud trading strategy can be implemented effectively in a deep neural network. Secondly, we propose a combination of an LSTM model and a practical, profitable trading strategy specialized for a niche stock market (Vietnam). Finally, we follow a traditional scientific framework while evaluating the performance by the standards of the modern financial world. Hence, our study may prove to be valuable to both scientific researchers and financial experts.

In this study, we propose an LSTM network that uses historical price changes in percentages and features extracted from the Ichimoku Cloud strategy to forecast the probability to outperform the market of all stocks listed in the VN30-Index. Next, we select the top three stocks with the highest probability as stocks in a short-term uptrend. We then employ the following short-term trading strategy: We buy the shares of selected short-term uptrend stocks and then sell them ten days afterward. After getting the money from the closure of our positions, we reinvest the money right away with the same trading strategy. The yearly returns of this trading scheme are used as the empirical results.

The rest of this paper is organized as follows: In Sect. 2, we clarify the software and hardware used in this study, how we determine the baselines and why we choose the VN30-Index as the data sample. Section 3 covers our methodology. Section 4 indicates our empirical results. Lastly, we discuss the result and shed some light on what may come in our future works.

2 Data Sample, Baselines, and Technology

We specifically choose the VN30-index constituents as the targets of our study because we want to avoid stocks with potential price manipulation. In order to be qualified for VN30, a stock must pass strict requirements in terms of liquidity, market cap, free float ratio, and reputation. Therefore, stocks listed in the VN30-index are more likely to reflect the true demand and supply of the market.

There are two categories in our list of baselines. The first one is the market indexes (VN30 and VN-index) and some common, safe investments such as gold, saving accounts (we used the saving interest rate of AgriBank, one of the biggest banks in Vietnam), and government treasury bonds. We want to see how our trading strategy fares against the actual market and common, proven investments. We collect gold historical performance from Kitco, Vietnam market indexes from FireAnt, Agribank's saving interest rates from the bank's official documents, and government treasury bond's historical rates from Investing.com. In the second category, we include the performance of our LSTM model using the same trading strategy but without the features from the Ichimoku Cloud to highlight its effectiveness in improving LSTM's prediction power.

We first create a timeline of all stocks listed in the VN30-Index since its establishment in 2012 to the end of 2020 by collecting news from CafeF, Vietnam's latest

economic, financial, and securities news channel. We then collect the dividend-adjusted open and closing prices of stocks from the timeline of Cophieu68.vn, a website specializing in stock analysis, daily stock market commentary.

The experiments are carried out in Google Colab. We implement our source codes with the assistance of the following libraries: Pandas, Numpy, Tensorflow, and Warnings.

3 Methodology

This study follows the general procedure from Krauss et al. [2]. There are four major phases. In the first phase, we split our data samples into study periods and then divide our study periods into training and test sets. In the second phase, we select the inputs and outputs necessary for our model. The third step is establishing the setup for our model. Ultimately, we apply our trading strategy based on the predictions.

3.1 *Creating Training and Testing Sets*

Firstly, we divide our original 9-year (2012–2020) period into “study periods” according to Krauss et al. [2]. The total length of a study period is four years. Each study period is furtherly divided into a training and test set. Since our LSTM model has a time sequence of approximately 1 year (240 days), the first year in a study period is only used to generate features. The remaining three years are split into a 2-year training set and a 1-year testing set. Figure 1 illustrates how we split our dataset.

3.2 *Features and Target Variable Selection*

Features Selection

Firstly, we introduce the calculations used in the Ichimoku Cloud trading strategy in order to avoid confusion in our upcoming explanation. For a more detailed description, we refer to [10].

- Conversion line

$$(9 - PH + 9 - PL)/2 \tag{1}$$

- Base line

$$(26 - PH + 26 - PL)/2 \tag{2}$$

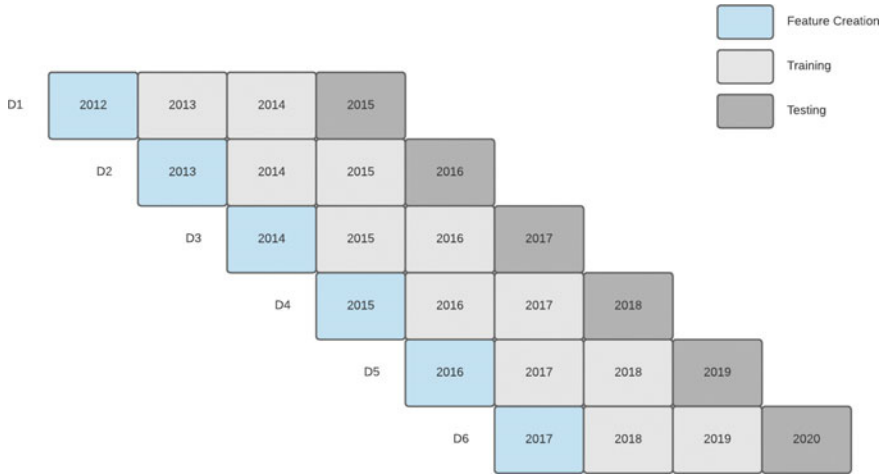


Fig. 1 Dataset division

- Leading Span A

$$(\text{Conversion line} + \text{Base line})/2 \tag{3}$$

- Leading Span B

$$(52 - \text{PH} + 52 - \text{PL})/2 \tag{4}$$

where n-PH stands for the highest closing price in the most recent n days and n-PL is the lowest closing price in the most recent n days.

Our input consists of 240 timesteps, and there are 5 features accompanied with each timestep:

- Returns in terms of 10th last closing price:

$$\text{Current Closing Price} / \text{10th Last Day Closing Price} - 1 \tag{5}$$

- Four ratios derived from the Ichimoku Cloud trading strategy:

$$\text{Conversion line} / \text{Base line} \tag{6}$$

$$\text{Conversion line} / \text{Closing price} \tag{7}$$

$$\text{Leading Span A} / \text{Leading Span B} \tag{8}$$

Leading Span A/Closing Price (9)

We choose to use (6)–(9) as input features instead of (1)–(4) because buying and selling signals in the Ichimoku trading strategy focuses on the relationship between the closing price, the lines, and spans rather than the actual numbers. Finally, we use robust scaler to standardize our features.

Target variable selection

First, we define the cross-sectional median at time $t + 10$ trading return. $\text{Perc} = [0.5, 0.5]$ is used for cumulative summing to get thresholds. Then, we use the command `pandas.qcut` (quantile-based discretization function) with the three parameters:

- `x.rank`: rank of all percentage change after 10 days
- `perc`: thresholds for creating label range
- `labels = False`: Return label 0 and 1.

To split the dataset into two discrete bins. The result is a column with values of 1 or 0, representing class 1 (if the corresponding stock return after 10 days is bigger than the cross-sectional median value of all stocks at time t) and class 0 (if the corresponding stock return after 10 days is smaller than the cross-sectional median value of all stocks at time t). Using these labels as target variables, our model predicts the probability for each stock in VN30 to outperform the cross-sectional median return in period $t + 10$.

3.3 Model Specification

LSTM is an extension of recurrent neural network (RNN) architecture that was specialized for modeling time series and their long-range dependencies [17]. Hochreiter and Schmidhuber introduce its concept in order to overcome the vanishing gradients issue [18]. Our model has 25 LSTM cells that precede a dropout layer of 0.1 and a dense layer with 2 output nodes.

- Loss function: we use categorical cross-entropy
- Optimization: we use RMSprop
- Batch size: 64 with epochs = 200
- Early stop: patience of 10 epochs, monitoring the validation loss
- Train-validation ratio: 0.2.

3.4 Trading Strategy

Our model predicts the probability for each stock in VN30 to outperform the median return after ten days. Then, we buy the shares of three stocks that have the highest chance then sell them ten days later. We then reinvest the recouped money on the

following day using the same strategy. In this trading scheme, we spend the approximately same amount of money on each stock with the original principle being 100.000.000 VND. In order to generate more samples, we split the initial principle into three smaller investments. Each investment was spent on a different day from the others. For the ease of calculations, we assume that we can spend approximately the same amount of money on each stock and we can use up all of the available money to purchase new stock. Every purchase was deducted by 0.1% to compensate for the brokerage fee, and the sale money was deducted by 0.2% (Both brokerage fee and tax account for 0.1% each.).

4 Results and Discussion

Overall, the implementation of the Ichimoku Cloud features greatly enhances the performance of LSTM. As we can observe from Table 1, our proposed model with the Ichimoku Cloud features outperforms its non-Ichimoku counterpart in 5 out of 6 years (2017 being the exception) by a fair margin. Moreover, the non-Ichimoku suffers losses in 4 out of 6 years, while our proposed model only sees a loss of 13.88% in 2018. On average, the annual gains of our proposed model with the Ichimoku Cloud (14.29%) dwarfs the non-Ichimoku Cloud version (-2.86%). We can observe that features from the Ichimoku Cloud even turn what is originally a notable loss into a significant profit. The empirical evidence taken from this table is capable of proving that features extracted from the Ichimoku Cloud indeed improve the prediction power of LSTM.

On the one hand, it can be inferred from Table 2 that our trading scheme's performance is very competitive with the performance of the local stock market indexes. Overall, the annual gain of our methodologies slightly outperforms the growths of the VN30-index and VN-index by approximately 19% and 4.2%, respectively. It is also worth mentioning that market indexes are not subjected to brokerage fees and taxes like our strategy. Therefore, being able to outperform the market indexes with

Table 1 Comparisons of the performance with and without the Ichimoku Cloud

Year	Annual gain	
	With Ichimoku (%)	Without Ichimoku (%)
2015	+31.85	-21.24
2016	+15.42	-11.71
2017	+21.87	+48.44
2018	-13.88	-20.05
2019	+12.33	+4.53
2020	+18.13	-17.17
Average	+14.29	-2.86

Table 2 Comparisons with the annual performances of stock market indexes and safe investments

	This paper's strategy (%)	VN30 (%)	VN-index (%)	Vietnam 1-year savings (%)	Gold (%)	Vietnam 10-year treasury bond (%)
2015	+31.85	-1.01	+6.12	+6.2	-11.59	+6.43
2016	+15.42	+5.38	+14.82	+6.5	+8.63	+7.03
2017	+21.87	+55.29	+48.03	+6.5	+12.57	+6.01
2018	-13.88	-2.36	-9.32	+6.3	-1.15	+4.09
2019	+12.33	+2.82	+7.67	+6.8	+18.83	+4.88
2020	+18.13	+21.81	+14.87	+3.9	+24.43	+3.15
Average	+14.29	+12.01	+13.7	+6.2	+8.62	+5.31

such handicaps is a considerable achievement. In terms of consistency, our strategy outperforms the market growths in 3 years (2015, 2016, and 2019) out of 5 years.

On the other hand, on average, our trading strategy outperforms all other safer and more common investments during the 9-year study timeframe.

5 Conclusion and Future Works

Applying deep learning in the Vietnam stock market is a virtually unexplored domain since the 15-year-old market is still at the infantile stage compared to 100-year-old giants such as Dow Jones. To contribute to this topic, this paper proposes an LSTM model that predicts the probability of a VN30 stock outperforming the market and an accompanying trading strategy with an annual return of +14.287%. We also aim to showcase the Ichimoku Cloud as a viable technical indicator that can be implemented as a feature in deep learning models. Besides, our study follows the general procedures of a scientific study while using modern financial standards as criteria for empirical analysis. Thus, experts from both the academic and financial worlds may find this research valuable.

There are several prospects in our future works. Firstly, the trading strategy employed in this paper is somewhat pristine. In the future, we may consult with professional financial advisors to construct more sophisticated trading principles in our scheme. Lastly, the architecture used in this LSTM model is fairly basic. The first step toward improving our LSTM model might be adding more layers or cells. Another possible direction is employing a more advanced version of LSTM such as stacked LSTM, bidirectional LSTM, and hybrids. In addition, attention mechanism, as indicated in the work of Qiu et al. [19], can improve the performance of LSTM and we may look to incorporate this in the next framework. We can also make a more comprehensive framework by using an ensemble of models instead of just a single

model. This approach will reduce generalization error since the components in an ensemble are diverse and work independently from each other.

References

1. Huynh, H.D., Dang, L.M., Duong, D.: A New Model for Stock Price Movements Prediction Using Deep Neural Network. In: Proceedings of the Eighth International Symposium on Information and Communication Technology (SoICT 2017), pp. 57–62. Association for Computing Machinery, New York, NY, USA (2017).
2. Krauss C, Do XA, Huck N (2017) Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500. *Eur J Oper Res* 259(2):689–702
3. Ghosh P, Neufeld A, Sahoo JK (2021) Forecasting directional movements of stock prices for intraday trading using LSTM and random forests. *Financ Res Lett* 41:102280
4. Makrehchi, M., Shah, S., Liao, W.: Stock prediction using event-based sentiment analysis. In: 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), vol. 1, pp. 337–342 (2013).
5. Oncharoen, P., Vateekul, P.: Deep learning using risk-reward function for stock market prediction. In: CSAI '18: Proceedings of the 2018 2nd International Conference on Computer Science and Artificial Intelligence, pp. 556–561, Association for Computing Machinery, New York, NY, USA (2018).
6. Deepika, N., Bhat, M.N.: An Efficient Stock Market Prediction Method Based on Kalman Filter. *Journal of The Institution of Engineers (India): Series B* 102(4), 629–644 (2021).
7. Das, S., Mishra, S., Senapati, M.: A Novel Model for Stock Price Prediction Using Hybrid Neural Network. *Journal of The Institution of Engineers (India): Series B* 100(4), 387 (2019).
8. Sharaf, M., Hemdan, E.ED., El-Sayed, A., El-Bahnasawy, N.A.: StockPred: a framework for stock Price prediction. *Multimedia Tools and Applications* 80(12), 17923–17954 (2021).
9. Chen, C., Zhao, L., Bian, J., Xing, C., Liu, TY.: Investment Behaviors Can Tell What Inside: Exploring Stock Intrinsic Properties for Stock Trend Prediction. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 2376–2384, Association for Computing Machinery, Anchorage, AK, USA.
10. Tra, N., Tien, H., Dat, N., Vu, N.: Vn-index trend prediction using long-short term memory neural networks. *Journal of Science and Technology: Issue on Information and Communications Technology* 17(12.2), 61 (2019).
11. Do QH, Tran T (2020) Forecasting Vietnamese stock index: A comparison of hierarchical ANFIS and LSTM. *Decision Science Letters* 9(2):193–206
12. Lien MD, Sadeghi-Niaraki A, Huy HD, Min K, Moon H (2018) Deep Learning Approach for Short-Term Stock Trends Prediction Based on Two-Stream Gated Recurrent Unit Network. *IEEE Access* 6:55392–55404
13. Hai, P.N., Tien, N.M., Hieu, H.T., Son, N.T., Son, N.T., Chung, P.Q., Ha, P.N.: An Empirical Research on the Effectiveness of Different LSTM Architectures on Vietnamese Stock Market. In: 2020 International Conference on Control, Robotics and Intelligent System (CCRIS 2020), pp. 144–149, Association for Computing Machinery, New York, NY, USA (2020).
14. Gurrib, I., Kamalov, F., Elshareif, E.: Can the leading US energy stock prices be predicted using the Ichimoku Cloud? *International Journal of Energy Economics and Policy* 11(1), 41–51 (2020) <https://doi.org/10.32479/ijeeep.10260>
15. Lim, K.J.S., Yanyali, S., Savidge, J.: Do Ichimoku Cloud Charts Work and Do They Work Better in Japan? *International Federation of Technical Analysts Journal* (2016 edition), 18–24 (2016).
16. Deng S, Yu H, Wei C, Yang T, Tatsuro S (2020) The profitability of Ichimoku Kinkohyo based trading rules in stock markets and FX markets. *Int J Financ Econ* 26(4):5321–5336

17. Sak, H., Senior, A., Beaufays, F.: Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In: Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH), pp. 338–342 (2014).
18. Hochreiter S, Schmidhuber J (1997) Long short-term memory. *Neural Comput* 9(8):1735–1780
19. Qiu J, Wang B, Zhou C (2020) Forecasting stock prices with long-short term memory neural network based on attention mechanism. *PLoS ONE* 15(1):e0227222

Applications of IoT in Industrial Transformation and Green Manufacturing



Arshi Naim , Mohammad Rashid Hussain, Salem Alelyani, and Mohammed Saleh Alsaqer

Abstract Internet of Things (IoT) is the most growing technological branch of computer science. In the current scenario, IoT and green manufacturing (GM) have impacted all the sectors including materials and manufacturing sectors. GM and IoT are two important applications applied in major business domains for the positive results. This study shows the role of IoT as an example for emerging technology in industrial transformation (ITN) and in GM in general context. This extended research article shows the relevance of IoT for achieving optimum growth, development, and safe working for ITN and GM. This research paper is an instance-based qualitative analysis that explains how IoT has contributed in the important fields of ITN and GM. The paper covers a general impact and advantages of green manufacturing in business process modeling (BPM) and product life cycle (PLC). The results show that GM and IoT are well integrated for ITN. The working of ITN has become effective and has contributed to the social benefits with the applications of GM and IoT.

Keywords Internet of things · Industrial transformation · Green manufacturing · Business process management · Product life cycle

A. Naim (✉)

Department of Information Systems, King Khalid University, Abha, Kingdom of Saudi Arabia
e-mail: arshi@kku.edu.sa

M. Rashid Hussain · S. Alelyani · M. Saleh Alsaqer

Center for Artificial Intelligence, College of Computer Science, King Khalid University, Abha, Kingdom of Saudi Arabia
e-mail: humohammad@kku.edu.sa

S. Alelyani

e-mail: s.alelyani@kku.edu.sa

M. Saleh Alsaqer

e-mail: msalsaqr@kku.edu.sa

1 Introduction

For past one decade, the climatic concerns have aggravated, and researchers have been trying hard to bring solutions to reduce the pollution and wastes that are increasing due to industrialization and technological advancements. Green manufacturing (GM) can be seen as a major and important solution to this issue which aids in reducing wastes and pollution in production processes [1]. This change covers many domains of GM that result in better production design, introducing new BPM and using eco-friendly devices and materials as inputs in production processes. Climatic change across the world has brought new measures of GM and imposed competitive notion at all the business services to adopt GM in the BPM [1]. However, not all sectors could utilize the GM at all level, and its application is still at developing stage due to unavailability of good replacements [1, 2]. Many researchers have explained the concepts and experiential learning related to green productivity (GP) and its advantages in profit maximization and achieving customer satisfaction. As mentioned earlier, many firms are still unaware of valid application of GM, therefore, the factors related to successful implementation of GM have to be known to improve production processes and increase competitiveness. GM has significant advantages in ITN as this has changed the entire concept of BPM and contributed in environment friendly production and distribution [2].

World's industrial companies are stressed by the competitive and divided markets, continuing economic uncertainties and growing operational costs (OC) [3]. All these issues have to be solved through reducing the OC, increasing productivity along with the achievement of competitive advantage [3]. The solution is not easy to implement because that also requires some investment like increasing cost in technology. Cost in operation technology (OT) may help in reducing OC but increasing other technological-related expenditures. Traditional industries have focused on OT for competitive advantages, but with the introduction of information systems (IS), better decision-making options came into existence [3]. IS gave rise to new digital technological developments in ITN and offered solutions that could integrate OT and information technology (IT) [3]. This was the time of application of IoT in ITN. This combination could mean exploring more options for business processes and adding values to the products. IoT creates a liaison between business and all the human resources and other tangible assets and resources that give rise to obtaining valuable business vision for industrial competitors [4].

ITN is a practical and synchronized principle to facilitate IoT to develop chronological changes and developments in operations. ITN is assessed by changes in the context of industrial structure, defined as the ratio of the variety of imitation—to innovation-based intermediate goods. IoT creates integration of digital technology into all areas of an ITN and aids in the methods of operations, CRM, BPM, and other areas needed for transformations in industries. Apart from these changes, IoT also brings changes in organizational culture [4].

The industrial internet of things (IIoT) suggests to interrelated sensors, instruments, and other devices networked together with computers' industrial applications,

including manufacturing and energy management [4, 5]. This connectivity allocates for data collection, trade, and investigation, beneficially directing to improvements in productivity and efficiency and economies of scale. The IIoT is an evolution of a distributed control system (DCS) that allows for a higher degree of automation by using cloud computing to refine and optimize the process controls [5]. There are three essential components of a digital transformation, namely the repair of processes, the repair of operations, and the rebuilding of relationships with customers. IoT provides three types of IT, for example, process transformation (PTN), business model transformation (BMT), domain transformation (DT), and cultural/organizational transformation (COT) [6]. This era is termed as digital era, and Internet of Things (IoT) has helped in creating new and innovative products that can know, sense, and share data as well as information to all the users such as companies, suppliers, and consumers. There are many retail products' examples, such as Sunsilk shampoo, Reebok sneakers, or even beverages such as Pepsico or fruit juices like Tropicana, all these products have extended the basic functionalities of regular products by offering the ability to collect and share data via the Internet. This new category of data collecting and sharing products is called as IoT-ready products or simply IoT products (IoTP) [6, 7].

Such IoTP may be seen as a provisional stage in the development of smart products (SP), which are able to analyze and, potentially, "interpret" usage data in a goal-oriented way. SP can make decisions that would require human learning and critical thinking. For example, DSS or machine learning methods can analyze the stage of SP and use data to evaluate customer behavior, preferences, and complaints too that may contribute in the process of new product development considering the achievement of customer satisfaction and eventually depending on the product type and application purpose, decisions made by a SP can be used to provide users and the company with recommendations or rating and reviews for further research and development [8]. Figure 1 shows the development stages from traditional products (TP) to (IoTP) to (SP) where TP has only basic functionalities, IoTP includes function of autonomous data collection and product analysis through virtual interfaces, and finally, SP encompasses data analysis, artificial applications, machine learning DSS applications for decision-making processes [8].

This research paper is a descriptive study segmented into three parts. First part covers the literature review giving the details on historical and growth of IoT in ITN and GM. Second discusses the general features of IoT in ITN and their integration along with GM. This paper defines the roles, advantages, challenges of IoT in ITN, also gives a brief account on the firms using IoT in their ITN processes. Paper also covers the general scenario of GM in BPM and PLC. Results in the third part outline the advantages and relevance of IoT in ITN and GM in BPM and PLC along with

Fig. 1 Transformation from TP to SP [8, 9]



their integration. The paper is concluded with the clear advantages of IoT and GM in the current environment [9].

The contribution of this study is to integrate three concepts together, which are IoT, ITN, and GM and their applications in BPM and PLC. Also, this paper will unify the advantages of IoT and explains the relevance of GM for competitive advantages; therefore, firms will realize that investing in IoT and GM will bring cost advantages and profit maximization.

2 Literature Review

Firstly, in this part, we have explained the definition of GM, advantages, and its applications, and we have given a brief account on GM's role in PLC and BPM. In the second part of the same section, we highlight some of the related concepts in the applications of IoT techniques in industrial transformation (ITN) [11].

GM means as green and sustainable manufacturing process that identifies manufacturing as the BPM for producing outputs from inputs for achieving customer satisfaction and earn good return on investments. In this process, the focus should be of creating less pollution and reducing wastes from both materials and finished goods. Researchers have studied the past work to know the importance and results of sustainable manufacturing and significantly concluded many several benefits of GM at all levels of BPM and phases of PLC. The analytical results suggest a positive correlation between environmentally sustainable manufacturing practices and competitive outcomes and establish a relationship between GM and BPM as well as for PLC [10].

Green manufacturing covers all factors which are important for conserving and integrating green inputs and outputs in the process of ITN, BPM, and achieving CRM. GM has many advantages in applying it in BPM such as it can moderate and reduce the pollution level in air, water, and land. This advantage is for all living things, flora, and fauna, because GM can reduce waste at the source [11]. When GM is used in materials and processes of production focus to use the eco-friendly inputs, reuse of materials, recycling, eliminating toxic release in BPM, and in all the phases of PLC. In past one decade, many researchers have confirmed that GM is the most efficient method of improving BPM and reducing toxicities. Firms who have adopted GM in their BPM have minimum adverse impact on environment and have optimum consumption during the manufacturing process, not only this also GM has resulted in increasing competitive advantages. Firms applied GM have achieved their organizational and strategic advantages through process design (PD), where the PD in GM aids in reducing the wastes, pollutions, and achieve CRM and profit maximization. Studies have also shown that firms using GM have consideration in measuring performance by dealing with two notions, GM dimension and GM factors. Dimension explains various qualitative aspects, viewpoints for achieving optimum GM, and factors elaborate the application part that may include product design,

synthesis, processing, packaging, transportation, and recycling in the BPM, and all the phases of PLC [11, 12].

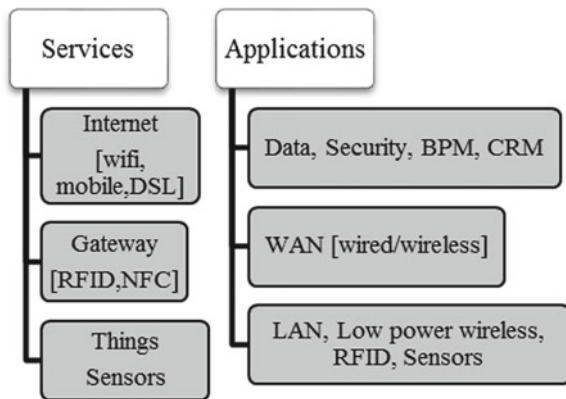
Over the last decade, the term Internet of Things (IoT) has attracted attention by projecting the vision of a global infrastructure of networked physical objects, enabling anytime, anyplace connectivity for anything and not only for any one. According to IoT, it is an open and comprehensive network of intelligent objects that have the capacity to auto-organize, share information, data and resources, reacting and acting in face of situations, and changes in the environment.

This process of ITN started in the UK in the eighteenth century and from there moved to other parts of the world. French writers mentioned the term ITN in 1852 during the industrial revolution, but Britishers were the pioneers in making it known to the world. Economic historian from the UK in 1852 defined ITN as a way to UK's economic development [12]. One of the most famous examples of ITN is a textile industry which was the first to use modern production methods. Connectivity and data collection as well as analysis are major roles of IIoT for achieving competitive advantages and explained the fundamental principle of growth and development in ITN. IoT has brought major ITN by decreasing unpredicted economic depressions and increasing productivity, which is subjected to over scheduled repairs, reduce complete preservation costs and eradicate crashes. It is assumed that IoT devices are integrated into various forms of energy consuming devices that can communicate with the different requirements of firms to balance power generation and energy usage which will help in ITN eventually.

In general, IoT system consists of three main parts, viz., sensors, network connectivity, and data storage applications (Fig. 2).

IoT is a global network which allows the communication between human-to-human, human-to-things, and things-to-things, which is anything in the world by providing unique identity to each and every object and anything can be connected and communicates in an intelligent fashion that ever before. It supports coding and networking of everyday objects and things to render them individually machine-readable and traceable on the Internet, like the content created through coded RFID

Fig. 2 IoT system [13, 14]



tags and IP addresses linked into an electronic product code (EPC) network. The world changes day-by-day, with IoT along with it. Now, it goes beyond M2M interaction [14]. IoT services can collect requested data from different sources and transfer it to the other devices and systems automatically; thus, it became possible to solve daily personal issues and business tasks with ease and flexibility. Many researchers have already conducted in this area justifying the importance and relevance of IoT, and GM in various sectors our research focuses only on ITN and GM for BPM and PLC and their integration [14].

3 Research Methodology

This is a qualitative study entirely based on data collected from firm's website studying on how they applied IoT and what advantages and challenges they faced. We studied various web sources to understand how GM and its features are used in the current scenario. This is a descriptive analysis conducted on various firms to know the applications of IoT applications. Real examples will justify the usefulness of IoT in ITN, and in the final part of the study, we presented the impact and advantages of green manufacturing in PLC and BPM.

4 Discussion

IoT plays a major role in ITN where asset-intensive industries, such as manufacturing, where the combination of OT and IT aids in improving new product developments, improving the existing product and changing the mind-set of old methods of performing business practices [15]. IoT gives rise to opening of new channels, developing new customers, identifying new products' usages, in simple terms a complete change over in industry's way of processing. IoT gave rise to ITN, where firms are able to enhance their value BPM by using various IT-based business models and developing the products as per the customers' requirement to meet customer satisfaction and good return on investment. Application of IoT-based solutions at any part of the firm, from the operation level to strategic or production level, the objective is to cut OC and use better OT and IT so that the firm can reach to its breakeven point in less time and can increase profit [15]. This is the most important reason for ITN to implement IoT in most of the industrial processes. In the current scenario, many companies use IoT solutions for receiving benefits from their investments and decreasing OC and other related expenditures [15, 16]. This is evident that IoT provides real-time vision and advantages to the BPM for any companies from collecting the data to analyzing and using it for decision-making processes for different levels of management. Apart from these monetary gains, IoT also aids ITN in effective SCM, building relationships between customers and firms, discovering new channels for the flow of products, encouraging firms to apply personalization approaches. IoT advances

Table 1 Threats of IoT in ITN [18]

Data security and privacy concerns are seen as the most challenging issues in IoT initiatives and strategies
Cultural change in the plants caused by implementing IoT solutions and smart machines
Data security and privacy concerns
Cost of purchase of IoT solutions
Lack of data scientists and analytics
Slow Internet connections
Cost of implementation and management of IoT solutions
Employees lack of technical development
Employee’s internal organization challenges
Building the business case for IoT investment

internal performances at industrial levels by empowering the human resource and improving their performance, enhancing skills, and encouraging better results. As mentioned before, IoT is used for ITN for the purpose of resolving users-based problems such as problems related to services, usages, quality, durability, availability[16]. Considering the importance of IoT in ITN, there could be two major roles of IoT; firstly, focusing on over all industrial improvement and specifically targeting at the internal level of related firms; secondly, aiming to facilitate for external factors such as creating needs, achieving CRM, contributing in economy of the nation and development of new as well as improved business models [17]. When IoT has several advantages, there are also noticeable threats that industry in general has to face. Table 1 shows some of the threats of application of IoT in ITN is given below:

The academic literature on IoT-related topics can be traced back to early publications on ubiquitous computing that correspond to the idea of information technologies penetrating “the fabric of everyday life until they are indistinguishable from it.” IoT development has affected many areas, and some of these areas are shown as examples in Table 2.

Firms have been using IoT in ITN for various benefits, and firms in material manufacturing are found to use IoT for the advantages shown in Fig. 3.

IoT in ITN applied at an industrial level has a broader scope and definitions. Firms try to observe the IoT applications at industrial level before applying at organizational level. Figure 4 shows the application of IoT at industrial level and explains how these areas are benefitted by the IoT.

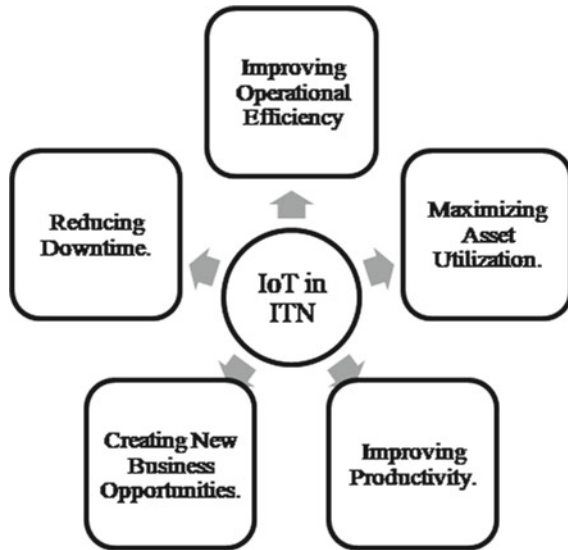
IoT is used by ITN for innovation, where new value, new pricing methods could be developed; competition has changed by IoT for new competition rules and enhances choices for the customer. IoT assists in data analysis for profit optimization in ITN processes. Other scopes of IoT are in business, customer, and resources too.

There are many firms which are using IoT for ITN and have experienced several benefits. Table 3 gives the list of those firms.

Table 2 Development of sectors with IoT applications [19]

Internet of things
Sensor technologies
Wireless communication, as well as supply
Layered architectures of digital technology
Energy consumption
Harvesting
Strategic management
Transportation
Supply chain management (SCM)
Market competition and new business models
Customer relationship management (CRM)
Privacy and security applications
Wearable devices

Fig. 3 IoT used by firms for benefitting their ITN processes [19]



Some of the examples for using the IoT in ITN are, IoT ABB Robotics that has applied IoT for achieving higher efficiency, for innovation purposes, and for developing new communication substrates. Another example is Dundee mining that applied IoT for data analytics for ITN transformation.

Few mobile communication firms have also utilized IoT for increasing profit and control costs, and Fig. 5 presents their main objective of using IoT in ITN.

GM is one of the important features of green supply chain. To achieve eco-friendly and cost-effective green supply chain, successful applications of other factors are also relevant along with GM. Figure 6 shows the list of green factors of green supply

Fig. 4 IoT scope and utility in ITN [20]

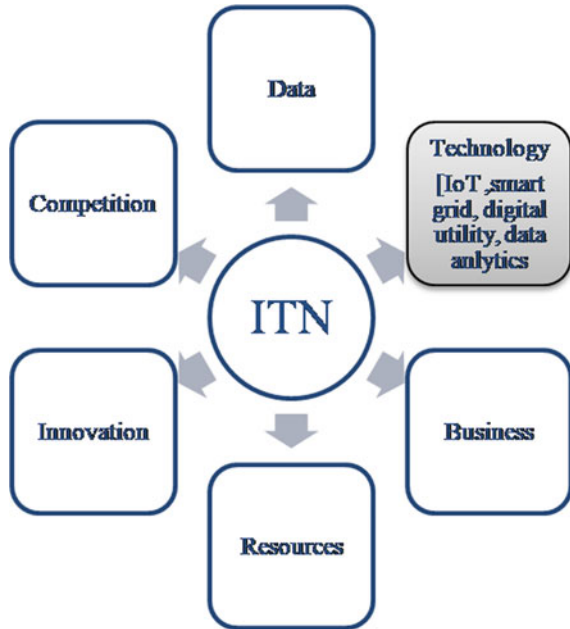


Table 3 List of the successful firms using IoT for ITN [21]

ScienceSoft (USA and Europe)
iTechArt (New York, US)
Oxagile (New York, US)
Indium Software (USA, UK, Singapore)
Softeq (Houston, Texas, USA)
Style Lab IoT Software Company (San Francisco, CA)
HQ Software Industrial IoT Company (USA and Europe)
PTC (Boston, Massachusetts)

chain. It is important to mention that all factors are interdependent and GM is the key factor. For the purpose, the study, we discussed only Green Designing (GD), Green Processing (GP), and Green Packaging (GPY).

GD focuses on producing eco-friendly outputs that minimize waste and maximize materials utilization. As a part of ITN, GD is used by many firms these days for the purpose of minimizing waste. The process needs to reduce the toxics inputs used in the BPM in such a way that productivity and cost effective both can be achieved. Firms which have beliefs in environment preservation use GD in all the phases of PLC, and the purpose is to maximize materials utilization, product recycling, reuse, ease of re-production, and ensuing energy recycling. Firms using GD in GM are focused on providing an environmentally aware that measures the product green quality, best use of product design, coordinate different product development phases,

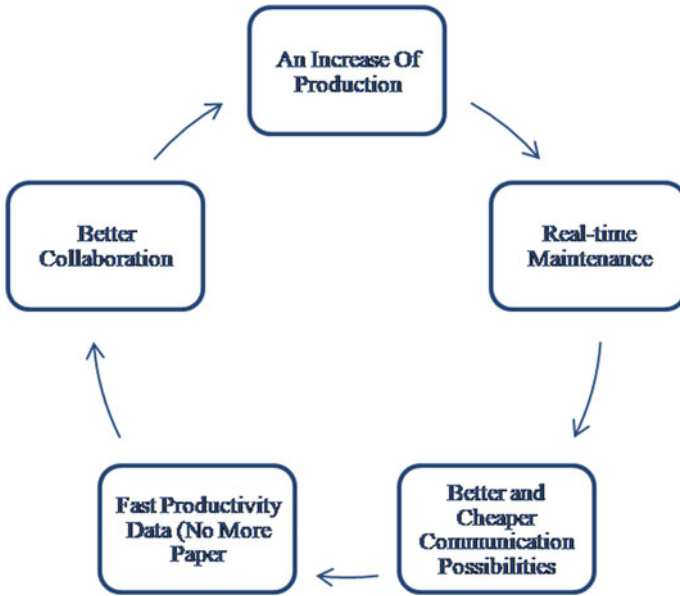


Fig. 5 Objectives of IoT in ITN [21]



Fig. 6 List of green factors for green supply chain [22]

and satisfy the green needs of consumers, companies, the environment, and society. Eventually, GD in the GM aids in developing good product and achieves company's competitive advantage. When the firms are more focused in using GM in their BPM, GP helps the firms to achieve their image of eco-friendly because GP tries to reduce all manufacturing processes the environmental burden in such areas as input resources, chemical substances used, and energy consumption. Figure 7 shows the role of GM in BPM in providing environmental advantageous and that can aid in achieving competitive advantages and enhance the firm's image [22].

GP focuses on the applications of environmental technologies that minimize the environmental hazards through sustainable product and process design to conserve energy and reduce reliance on non-replaceable raw materials; therefore, if firms use GP as one of the factors for GM, firm will be able to use GM in BPM control pollution at all the phases of PLC and reduce the waste through recycling (refer Fig. 8). GM and technologies are applied in ITN at all the levels such as in manufacturing, operational processes, BPM, production technologies, and management-oriented factors like CRM. GPY has a major role in ITN and in green marketing because by adopting the concept of GPY, firms can minimize the environmental pollution and encourage using

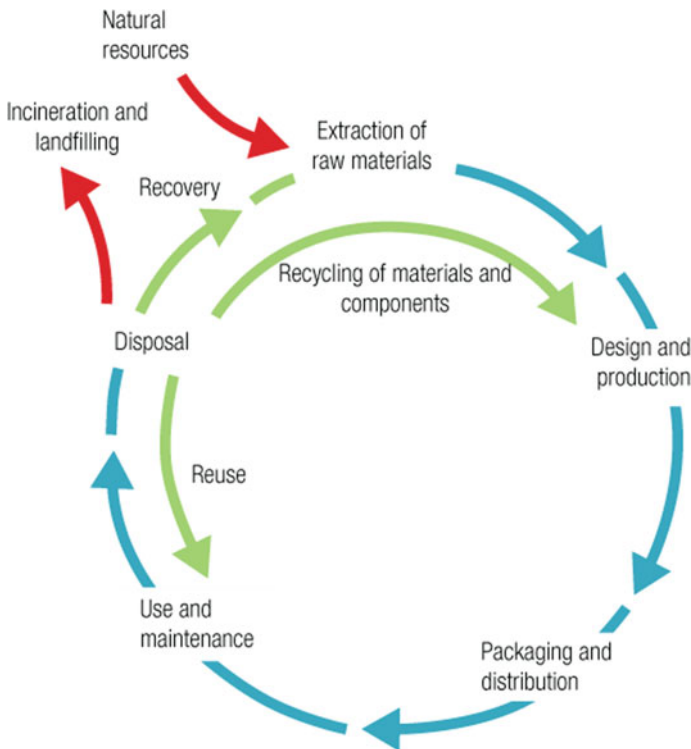


Fig. 7 Role of GM in BPM [23]

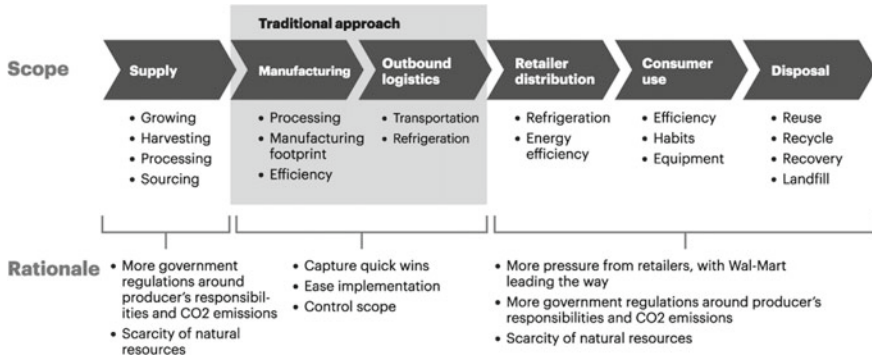


Fig. 8 GM in PLC in the production process for sustainability [3]

Table 4 List of firms using GM [3]

Nike
Johnson and Johnson
Philips Electronics
Earth Tech
Schott
Dell Inc
Tupperware

the material which is GCYL [24]. Many firms conduct their marketing campaign on use of GPY to achieve competitive advantages and also to build CRM.

Many firms are motivated for following eco-friendly concepts and applied GM in their production processes; some of the famous companies using GM are given in Table 4.

Research found that firms with different nature and products have used GM in developing their products and offering the services and all achieved success in using eco-friendly methods and materials in the processes. These firms have applied GM in all the stages of PLC and in carrying out BPM and were able to maximize profit and achieve competitive advantages. Also, such examples motivate other firms to follow the GM in their BPM. This descriptive study explains the importance of IoT in ITN and GM in BPM and developing PLC models by adopting the concepts of GM that includes the subcomponents such as GD, GPY, and GCYL.

5 Result

Innovation has become an imperative for the entire world’s industrial companies. Dealing with cost pressure is the most crucial issue for most of the companies. IoT

has provided solutions to such critical issues and resolved OC and IT-related problems at internal as well as external levels. IoT has provided major advantages such as increasing productivity, reducing OC, improving operational efficiency by introducing new and advanced business models, effective data analysis, and achieving CRM. The majorities of companies are already using IoT applications in different phases of production, investment decisions, and analyzing customers' data and reached to the beneficial solutions. Although there are many limitations and threats of IoT applications pertaining to privacy and security, but industries for its transformation will continue using IoT because advantages are much more and limitations can be solved and prevented by the applications of other IS and IT tools.

We studied many firms and checked the challenges and benefits that they have faced while adopting IoT in ITN. Below given Figs. 9 and 10 show the challenges and benefits for the different factors.

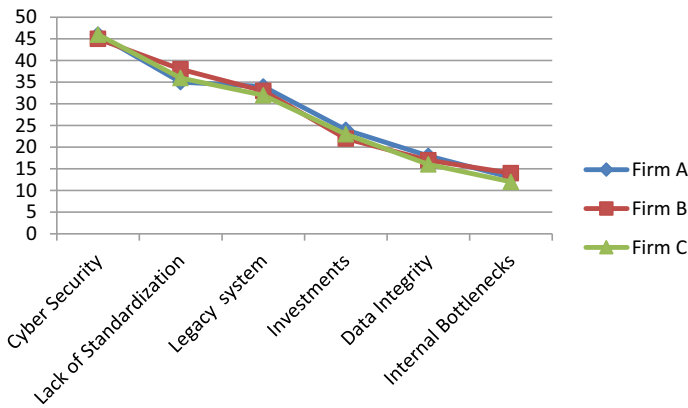


Fig. 9 Challenges of using IoT in ITN by three firms [25]

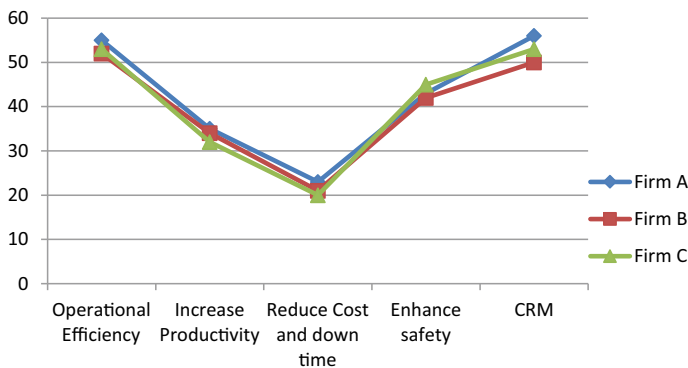


Fig. 10 Benefits of using IoT in ITN by firms [26]

The study from three firms shows similar percentage of challenges and benefits that they have experienced for applying IoT in their ITN. For the purpose of privacy, the names of these firms are not disclosed.

The Internet has made fundamental changes in all the functions of ITN, and with the IoT, online environment has expanded to a greater extent. The notion of joining the TP to IoTP, growth of new product mix for collecting, analyzing the data remotely and automatically that eventually aids CRM. PAs are done by the IoTP for product usage and its value for the customers and sharing this information with other customers, firms, and their competitors.

The implementation of a GM means firms are using eco-friendly material in the BPM and all the phases of PLC, which clearly means that all goods are made from non-toxic materials and biodegradable materials. Also, it is noticed that change in material has not only affected the finished products but also affected the PD and its SCM and may increase in cost for some materials, but in return advantage of GM is so high that these issues were small and easy to manage. Also, firms applying GM in the process of BPM are acknowledged as responsible and show care for the nature and customers and contribute to the social benefits as a result, they achieve CRM and increase in profit.

6 Conclusion

The industries will increase their IoT spending in the future for ITN and currently, most of the firms are already investing in IoT applications for ITN because of high productivity and low OC. IoT has also facilitated in for prediction analysis for ITN that aided in decision-making process and analyzing the trends and risk management. In the current scenario, ITN sees its dependence on IoT for its all the functions and trusts the applications of IoT to grow more in the future.

Firms who have implemented GM in BPM and all phases of PLC have made major contribution in preserving the environment, but the process was not easy. They have to create and choose options which reduced wastes and pollution in the BPM. Sometime, the inputs for GM cost higher and also firms have to change or modify the entire PD to produce desired finished products. All the efforts for applying GM are rewarding in terms of customer satisfaction, increasing productivity, and environmental care.

References

1. Yadegaridehkordi E, Hourmand M, Nilashi M, Alsolami E, Samad S, Mahmoud M, Abdulsalam Alarood A, Zainol A, Majeed HD, Shuib L (2020) Assessment of sustainability indicators for green building manufacturing using fuzzy multi-criteria decision making approach. *J Clean Product* 277:122905
2. Bagdadee AH, Zhang L, Saddam Hossain Remus M (2020) A brief review of the IoT-based energy management system in the smart industry. *Artif Intell Evol Comput Eng Syst* 443–459

3. Milojevic M (2017) Digital industrial transformation with the internet of things
4. Chhatrawala J (2020) Industrial transformation through IOT and Industry 4.0
5. Bhardwaj A, Al-Turjman F, Kumar M, Stephan T, Mostarda L (2020) Capturing-the-invisible (CTI): behavior-based attacks recognition in IoT-oriented industrial control systems. *IEEE Access* 8:104956–104966
6. Boltz N, Walter M, Heinrich R (2020) Context-based confidentiality analysis for industrial iot. In: 2020 46th Euromicro conference on software engineering and advanced applications (SEAA), pp 589–596. *IEEE*
7. Aheleroff S, Xu X, Lu Y, Aristizabal M, Pablo Velásquez J, Joa B, Valencia Y (2020) IoT-enabled smart appliances under industry 4.0: a case study. *Adv Eng Inform* 43:101043
8. Li CZ, Chen Z, Xue F, Kong XTR, Xiao B, Lai X, Zhao Y (2021) A blockchain-and IoT-based smart product-service system for the sustainability of prefabricated housing construction. *J Clean Product* 286:125391
9. Abdel-Basst M, Mohamed R, Elhoseny M (2020) A novel framework to evaluate innovation value proposition for smart product–service systems. *Environ Technol Innov* 20:101036
10. Bag S, Wood LC, Mangla SK, Luthra S (2020) Procurement 4.0 and its implications on business process performance in a circular economy. *Resour Conserv Recycl* 152:104502
11. Mendling J, Pentland BT, Recker J (2020) Building a complementary agenda for business process management and digital innovation, pp 208–219
12. Stearns PN (2020) The industrial revolution in world history. *Routledge*
13. Mahmood Z (2020) Connected vehicles in the IoV: concepts, technologies and architectures. In: *Connected vehicles in the internet of things*, pp 3–18. *Springer*
14. Sankaranarayanan S, Rodrigues JJPC, Sugumaran V, Kozlov S (2020) Data flow and distributed deep neural network based low latency IoT-Edge computation model for big data environment. *Eng Appl Artif Intell* 94:103785
15. Varga P, Peto J, Franko A, Balla D, Haja D, Janky F, Soos G, Ficzer D, Maliosz M, Toka L (2020) 5G support for industrial iot applications—challenges, solutions, and research gaps. *Sensors* 20:3828
16. Khan N, Naim A, Hussain MR, Naveed QN, Ahmad N, Qamar S (2019) The 51 v's of big data: survey, technologies, characteristics, opportunities, issues and challenges. : *Proceedings of the international conference on omni-layer intelligent systems* (pp 19–24)
17. Cheng X, Zhang J, Tu Y, Chen B (2020) Cyber situation perception for internet of things systems based on zero-day attack activities recognition within advanced persistent threat. *Concurrency Comput Pract Exper* e6001
18. Alsaedi A, Moustafa N, Tari Z, Mahmood A, Anwar A (2020) TON_IoT telemetry dataset: a new generation dataset of IoT and IIoT for data-driven intrusion detection systems. *IEEE Access* 8:165130–165150
19. Pachayappan M, Ganeshkumar C, Sugundan N (2020) Technological implication and its impact in agricultural sector: an IoT based collaboration framework. *Proc Comp Sci* 171:1166–1173
20. Al-Sarawi S, Anbar M, Abdullah R, Al Hawari AB (2020) Internet of things market analysis forecasts, 2020–2030. In: 2020 fourth world conference on smart trends in systems, security and sustainability (WorldS4), pp 449–453. *IEEE*
21. Oztemel E, Gursev S (2020) Literature review of industry 4.0 and related technologies. *J Intell Manuf* 31:1127–182
22. De Carvalho LS, Stefanelli NO, Carolina Viana L, de Siqueira Camargo Vasconcelos D, Oliveira BG (2020) Green supply chain management and innovation: a modern review. *Manage Environ Qual Int J*
23. Lisi W, Zhu R, Yuan C (2020) Embracing green innovation via green supply chain learning: the moderating role of green technology turbulence. *Sustain Dev* 28(1):155–168

24. Dornfeld DA (2014) Moving towards green and sustainable manufacturing. *Int J Precis Eng Manuf Green Technol* 1(1):63–66
25. Naim A, Khan MF, Hussain MR, Khan N (2019) “Virtual doctor” management technique in the diagnosis of ENT diseases. *JOE* 15(9):88
26. Naim A, Alahmari F, Rahim A (2021) Role of artificial intelligence in market development and vehicular communication. *Smart Antennas: Recent Trends Design Appl* 2:28

Yaw Motion Control of a Ship Based on Improved Quasi-Sliding Mode



Rajashree Taparia and Priya Gautam

Abstract This chapter describes the control of a ship yaw motion, based on an improved quasi-sliding mode reaching law. The ship in the sea is an example, which exhibits highly nonlinear behaviour. It faces external perturbations which may include hydraulic forces, wind currents, tidal waves, etc. For smooth sailing, it is required to design a robust control. Sliding mode control has the ability to reject external disturbances. In this chapter, improved quasi-sliding mode reaching law-based control is discussed for the yaw movement of a ship. The idea is to reduce the quasi-sliding band so that the yaw motion is bounded.

Keywords Yaw motion · Quasi-sliding mode band · Robust control · Discrete systems

1 Introduction

The sailing ship in the sea faces number of external perturbations such as ocean waves, windy weather and effect of tides. To get safe and smooth seamanship in the sea, it is important to analyse and control the behaviour of a ship as regards to the above-stated perturbations. In the literature, classical PID controllers and linear optimal regulators are used to design the control law. The problem with these controllers is that these controllers cannot handle the multi-variable and nonlinear environment experienced by the ship. For the ship, still in the sea, the effect of winds and tidal waves can be considered as locally stationary stochastic process. Regulator control for decoupled yaw–roll motions are considered in [1]. In the ship steering based on linear optimal control, the ship is modelled as linear system and the stochastic disturbances are considered as Gaussian noise [2]. The Riccati equation solution to the ship steering problem, which minimizes a scalar cost function, does not ensure the individual objective satisfaction. Some authors [3–5] do consider a straightforward method of control design for individual objectives but they do not consider presence

R. Taparia (✉) · P. Gautam
University Departments, Rajasthan Technical University, Kota, Rajasthan, India
e-mail: rtaparia@rtu.ac.in

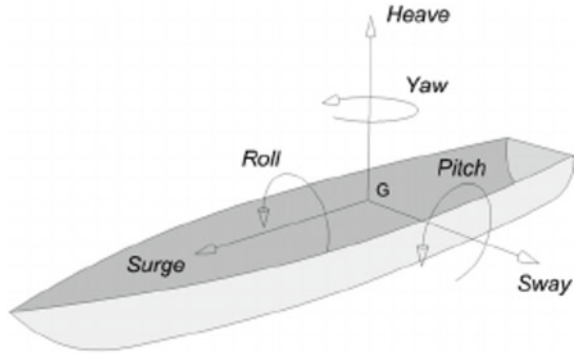
of perturbations. Therefore, the system may become unstable in the presence of perturbations for such a control. In [6], control of yaw motion of the ship based on Ackermann's formula is discussed. Another approach called upper bound covariance control UBCC is discussed in [3]. It satisfies variance constraints with perturbations but requires large gain in the presence of perturbations.

Sliding mode control (SMC) theory is well-established, both in continuous and in discrete time domain. It exhibits robustness in the presence of disturbances. The computer implementation of control algorithm necessitates to develop in discrete domain as direct application of continuous time law leads to large chattering, discretization errors or even instability. Many discrete sliding mode control (DSMC) algorithms have been developed by the researchers, and more popular among them are [7, 8]. Discrete time sliding mode control strategy for periodic review inventory systems is discussed in [9, 10]. It is important to point out here that the width of the quasi-sliding mode band (QSMB) is dependent on the rate of change of system uncertainties. In most of the chapters, this rate change is considered as the first-order difference of the system uncertainties. As this rate is large, the width of the quasi-sliding band is increased. The discontinuity of the sign function further adds to the chattering. While deriving the reaching law in [11], the rate of change of system uncertainties is taken as the difference function of the second order. Also, the sign function is replaced by its continuous approximation. The major contribution of this chapter is the objective to reduce the quasi-sliding mode band. This is done by taking the second-order difference of the system uncertainties. As such the ship in the sea faces many disturbances. Improving the control law in this way makes the ship control robust, in a sense that ship follows the desired trajectory in a relatively smooth fashion. The control law discussed in [11] is applied to the ship yaw motion. The variations in the system parameters and disturbance in the system are analysed through simulations. In the first case, it is considered that there is no disturbance in the system. In the second case, disturbance is added in the system. Further, the effect of switching parameter and sampling rate are seen on the quasi-sliding mode band. Ideally, the ship should follow smooth sailing such that the motion follows a bounded track. An improved version of discrete sliding mode control law, which reduces the quasi-sliding band is derived for the ship yaw motion. The reduced band will make the ship steering to follow a bounded path, facilitating smooth sailing.

2 The Ship Model

In general, any ship has two types of motion, linear and rotational. The linear motion further can be classified as surging, swaying and heaving. And the rotational motion can be classified as rolling, pitching and yawing (Fig. 1). The considered model of the ship takes the single crew as the control. For a ship like tanker, the prevailing weak coupling between different modes of the ship allows us to consider the steering dynamics separately. Yaw and roll motions are developed by the changes in the rudder positions. So to control the yaw motion, the input is the rudder angle. The rolling

Fig. 1 Ship motions model



and pitching angles are observed, and the yaw movement of the ship is controlled. The model equations for yaw motion of ship as described in [6] are given as,

$$\begin{aligned}
 m \left[\frac{d\check{u}(t)}{dt} - \check{v}(t)r(t) - X_g r^2(t) \right] &= X \\
 m \left[\frac{d\check{v}(t)}{dt} - \check{u}(t)r(t) + X_g r^2(t) \right] &= Y \\
 I_m \frac{dr(t)}{dt} + mX_g \left[\frac{\check{v}(t)}{dt} + \check{u}(t)r(t) \right] &= N
 \end{aligned} \tag{1}$$

The parameter descriptions in the above equations are given in Table 1.

The state space model from the above differential equations for the yaw motion of the ship can be written as [6],

Table 1 Description of the parameters

Parameter	Description
m	Mass of the ship
X_g	X co-ordinate of centre of mass
I_m	Moment of inertia
$\check{u}(t)$	Surge velocity
$\check{v}(t)$	Sway velocity
$r(t)$	Yaw rate
$\psi(t)$	Heading angle
X	Hydrodynamic force at X-axis
Y	Hydrodynamic force at Y-axis
N	Hydrodynamic moment

$$\begin{aligned} \begin{bmatrix} \dot{r}(t) \\ \dot{\psi}(t) \end{bmatrix} &= \begin{bmatrix} -1/T_s & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} r(t) \\ \psi(t) \end{bmatrix} \\ &+ \begin{bmatrix} K_s/T_s \\ 0 \end{bmatrix} \delta(t) + \begin{bmatrix} 0 \\ \eta \end{bmatrix} e(t) & (2) \\ &= Ax(t) + B\delta(t) + De(t) & (3) \end{aligned}$$

where

$$A = \begin{bmatrix} -1/T_s & 0 \\ 1 & 0 \end{bmatrix} \quad B = \begin{bmatrix} K_s/T_s \\ 0 \end{bmatrix} \quad D = \begin{bmatrix} 0 \\ \eta \end{bmatrix}$$

and

$$x(t) = [x_1(t) \ x_2(t)]^T = [r(t) \ \psi(t)]^T$$

In Eq. (2), $\delta(t)$ is the rudder angle of the ship, which is also the control input to the system. T_s , K_s are the time constants. $e(t)$ is force generated by the waves, which acts as a disturbance in the system.

3 Control Law

3.1 Methodology

To derive the robust control law for yaw motion control of the ship, the discrete model of the ship is considered. The dynamics of the reaching law $s(k+1)$ is described. The control law $u(k)$ based on this reaching law is derived from the discrete state-space model of the system and the sliding variable dynamics $s(k+1)$. Simulations are carried on the discrete model of the system with this control law, in MATLAB.

3.2 The QSM Law

As mentioned earlier, the ship will have a steady sail if the controller controls the yaw movement such that it remains bounded. An improved discrete sliding mode control (DSMC) reaching law is described in [11], which guarantees small width of quasi-sliding mode band (QSMB). The small quasi-sliding mode band for yaw movement of the ship will ensure its motion to be bounded. In an improved quasi-sliding mode domain, a new reaching law for discrete sliding mode control (DSMC) is described for the system with uncertainties. It is well known that there is a direct relation between the amplitude of chattering, i.e. width of the quasi-sliding mode band and the change rate of the system uncertainties. In most of the previous research, the rate change is taken as the first-order difference of the uncertainties and is defined as,

$$\delta_1(k) = C'[d(k) - d(k-1)] \quad (4)$$

$$|\delta_1(k)| \leq \delta_1 \quad (5)$$

where δ_1 is the upper bound on the rate of change of uncertainties. As this change rate is quite considerable, the width of the quasi-sliding mode band will be increased. Also, the discontinuity in the *sign* function further adds to increasing the chattering. Thus, it is required to reduce this quasi-sliding mode band, so that there is a smooth sail of the ship and robustness in control is improved. The switching function is described as,

$$s(k) = C'x(k) \quad (6)$$

where C' is $1 \times n$ vector and $C'B \neq 0$ which ensures that the system is controllable. To reduce the chattering, second-order difference is used as the change rate. In this chapter, sign function is replaced by a continuous approximation of it. This reaching law thus ensures the smaller width of the quasi-sliding mode band (QSMB). The switching function is described as,

$$s(k+1) = \frac{-1}{1-z^{-1}} \left(\frac{\varepsilon Ts(k)}{|s(k)| + \rho} + C'[d(k) - d(k-1)] \right) \quad (7)$$

where $\varepsilon > 0$ is the switching parameter, $\rho > 0$ is the control parameter, and z^{-1} is the unit delay parameter. Multiplying $(1 - z^{-1})$ both sides in the (7) we get,

$$(1 - z^{-1})s(k+1) = -\frac{\varepsilon Ts(k)}{|s(k)| + \rho} + (1 - z^{-1})C'[d(k) - d(k-1)] \quad (8)$$

$$s(k+1) = s(k) - \frac{\varepsilon Ts(k)}{|s(k)| + \rho} + C'[d(k) - 2d(k-1) + d(k-2)] \quad (9)$$

The rate change of uncertainty is expressed as,

$$\delta(k) = C'[d(k-1) - 2d(k-1) + d(k-2)] \quad (10)$$

$$|\delta(k)| \leq \delta \quad (11)$$

For regulating the convergence, the convergence rate parameter q is introduced and is taken as $q > 0$ and $0 < 1 - qT < 0$ [11]. Then the reaching law becomes,

$$s(k+1) = (1 - qT)s(k) - \frac{\varepsilon T s(k)}{|s(k)| + \rho} + \delta(k) \quad (12)$$

The *sign* function is replaced by a continuous approximation in the above-reaching law. From (6)

$$s(k+1) = C'x(k+1) \quad (13)$$

$$s(k+1) = C'[Ax(k) + Bu(k) + d(k)] \quad (14)$$

Equating (12) and (14)

$$(1 - qT)s(k) - \frac{\varepsilon T s(k)}{|s(k)| + \rho} + \delta(k) = C'Ax(k) + C'Bu(k) + C'd(k) \quad (15)$$

Then the improved switching function becomes,

$$s(k+1) = (1 - qT)s(k) - \frac{\varepsilon T s(k)}{|s(k)| + \rho} + \delta(k) \quad (16)$$

The control law then can be written as,

$$u(k) = -(C'B)^{-1}[C'Ax(k) - (1 - qT)s(k) + 2C'd(k-1) - C'd(k-2) + \frac{\varepsilon T s(k)}{|s(k)| + \rho}] \quad (17)$$

The application of this control law for yaw motion of the ship is discussed in the next section. The width of the the quasi-sliding mode band as described in [11] is given as,

$$\Delta = \frac{\varepsilon T + \delta}{2 - qT} \quad (18)$$

Therefore, it can be interpreted that small value of εT will reduce the quasi-sliding mode band.

4 Simulation Results

A single screw/single rudder ship model, as discussed in [12], is used here for simulations. The parameters of the ship are as described here. Bow and stern draught is 11 m. The metacentric height is 0.45 m. The displacement is 52,010 m³. A 13,000

HP diesel engine is used for delivering the propulsion power. The ship is fitted with a four-blend propeller with a diameter of 6.3 m². From the system equation (2), the system matrices are given as in (19).

$$A = \begin{bmatrix} 0.01525 & 0 \\ 1 & 0 \end{bmatrix} B = \begin{bmatrix} -0.000625 \\ 0 \end{bmatrix} D = \begin{bmatrix} 0 \\ 0.25 \end{bmatrix} \tag{19}$$

With the given parameters, the continuous time state space model is converted into discrete state space equations and is given below.

$$\begin{bmatrix} r(k + 1) \\ \psi(k + 1) \end{bmatrix} = \begin{bmatrix} 1.0002 & 0 \\ 0.0100 & 1.0000 \end{bmatrix} \begin{bmatrix} r(k) \\ \psi(k) \end{bmatrix} \tag{20}$$

$$+ \begin{bmatrix} -0.6250 \\ -0.0031 \end{bmatrix} \delta(k) + d(k) \tag{21}$$

The time varying disturbance is considered affecting the system as given by the following equation.

$$d(k) = a \times D \times e(k)$$

where $e(k) = 0.5 \sin(2k)$ and the initial conditions as,

$$x(0) = \begin{bmatrix} 7 \\ -7 \end{bmatrix}$$

Simulations are carried out for the following four cases as described here. In the first case, the disturbance is not considered in the system dynamics. In the second case, disturbance is introduced in the system, the third case shows the effect of ϵT to the system, and in the fourth case sampling time, variable parameter and ϵT are changed (Table 2).

Case-1: In the first case, we consider that there is no disturbance in the system. The parameter $a = 0$ (i.e. $d(k) = 0$) and $\rho = 0.7$. In this case, no disturbance occurs in the system so $\delta = 0.103$. The value of ϵT is selected according to the condition given in [11] as, $\epsilon T = 140$. The sliding variable $s(k)$, for this case, is shown in Fig. 2.

Table 2 Dataset used

Parameters	δ	ϵT	qT	a
Case-1	0.7	140	0.45	0
Case-2	0.7	140	0.45	1
Case-3	0.7	105	0.45	1
Case-4	0.7	121	0.45	0.1

δ —control parameter, ϵ —switching parameter, T —sampling time, a —adjustable parameter, q —convergence rate parameter

Fig. 2 Switching function with the reaching law for case-1

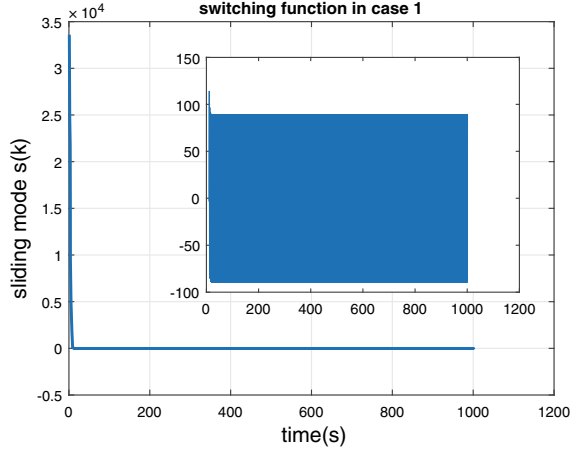
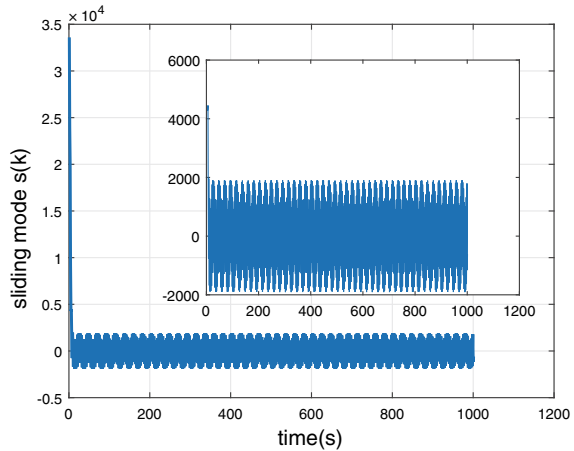


Fig. 3 Switching function with the reaching law in case-2



Case-2: In this case, we consider the presence of external disturbance in the system. The adjustable parameter is $a = 1$ in this case. Figure 3 shows the plot of switching function in the presence of disturbance $d(k)$.

Case-3: The adjustable parameter is taken as $a = 1$ in this case. Width of QSMD band is increasing with respect to ϵT . Hence, in order to obtain the minimum QSMD, ϵT should be kept small. The qT , ρ and $d(k)$ are same as the case-2. $\epsilon T = 105$. Simulation results are shown in Fig. 4.

Case-4: In this case, the adjustable parameter is changed to $a = 0.1$, and $\epsilon T = 121$. The disturbance affecting the system has been reduced with lesser a . As described in [11] width of QSMD is reduced with the sampling time as shown in Fig. 5.

The control input for this case with the described reaching law is shown in Fig. 6.

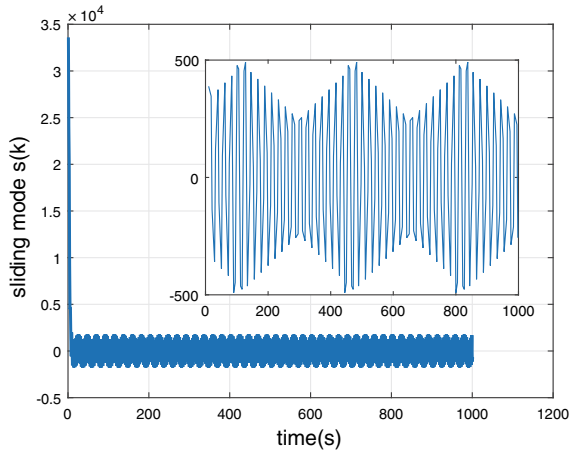


Fig. 4 Switching function with the reaching law in case-3

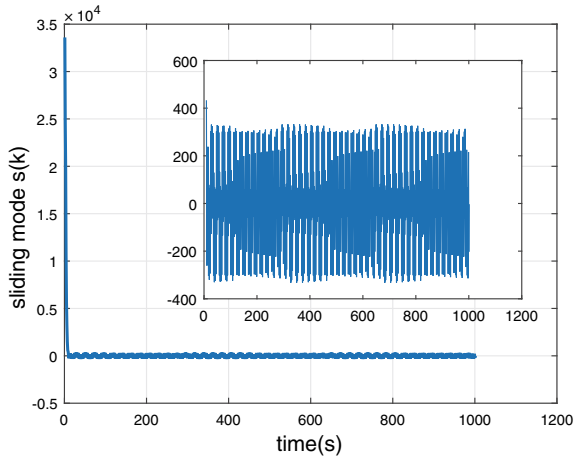


Fig. 5 Switching function with the reaching law in case-4

The system states, yaw rate $\psi(k)$ and the heading angle $r(k)$, are shown in Fig. 7.

The phase plane portrait as shown in Fig. 8 is of a stable system.

Fig. 6 Control input u

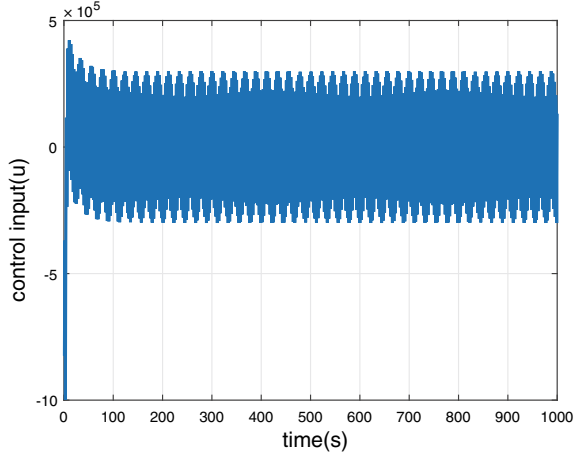
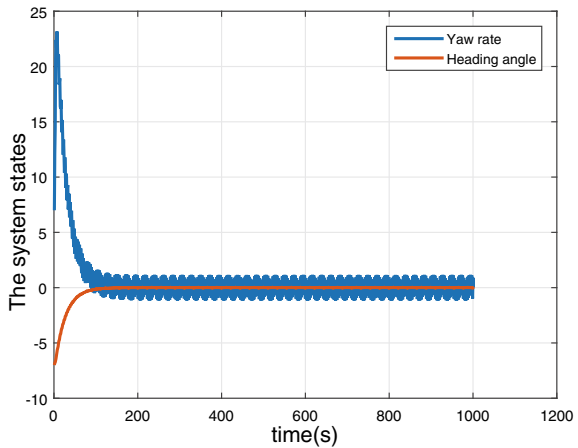
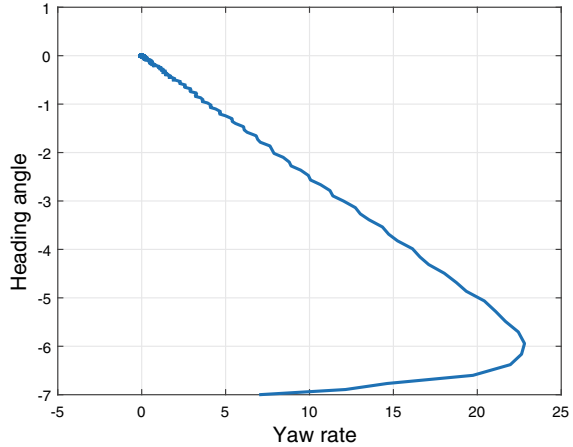


Fig. 7 Yaw rate and heading angle



5 Conclusion

In this chapter, improved discrete sliding mode reaching law-based control is applied for the yaw motion of the ship. This reaching law ensures the stability and the robustness of the system in the presence of external disturbances. The chapter also describes the effect of variations in the parameters of the system on the width of the quasi-sliding band. They are considered as separate cases in the chapter. The simulation results show the changes in the QSMB width, with the changes in the different parameters of the system dynamics. The phase portrait of the system is stable. The effectiveness of the control law is in reduced quasi-sliding band and less chattering, thus rendering the smooth sailing of the ship.

Fig. 8 Phase plane portrait

References

1. Carley J, Duberley A (1972) Design considerations for optimum ship motion. In: 3rd ship control system symposium-SCSS, Bath, UK
2. Kallstrom CG (1981) Control of yaw and roll by a rudder/fin stabilization system. In: Proceedings 6th SCSS
3. Chang WJ, Chung HY (1992) Upper bound covariance control of discrete perturbed systems. *Syst Control Lett* 19(6):493–498
4. Chang WJ (1997) Constrained dynamic controller design for ship steering yaw control. *J Marine Sci Technol* 5(1):83–93
5. Yeh HH, Banda SS, Heise SA, Bartlett AC (1990) Robust control design with real-parameter uncertainty and unmodeled dynamics. *J Guidance Control Dyn* 13(6):1117–1125
6. Malav DK, Taparia R (2017) Sliding mode control of yaw movement based on Ackermann's formula. In: International conference on computer, communications and electronics (Comptelix-2017). IEEE, pp 628–632
7. Gao W, Wang Y, Homaifa A (1995) Discrete-time variable structure control systems. *IEEE Trans Ind Electron* 42(2):117–122
8. Bartoszewicz A (1998) Discrete-time quasi-sliding-mode control strategies. *IEEE Trans Ind Electron* 45(4):633–637
9. Taparia R, Janardhanan S, Gupta R (2016) Management of periodically reviewed inventory systems with discrete variable structure control. In: 11th international conference on industrial and information systems (ICIIS), pp 49–53
10. Taparia R, Janardhanan S, Gupta R (2017) Management of inventory for periodically reviewed goods using discrete higher order sliding mode control. *IETE J Res* 63(4):552–557
11. Ma H, Wu J, Xiong Z (2016) Discrete-time sliding-mode control with improved quasi-sliding-mode domain. *IEEE Trans Ind Electron* 63(10):6292–6304
12. Chang WJ, Chang KY (2000) Multivariable performance-constrained sliding mode control for ship yaw-motion systems with perturbations. *Int J Adaptive Control Signal Process* 14(4):393–409

Natural Language Inference on Imbalanced Datasets



Nidarshan Kumar, Anirudh V. Ragam, GBS Akhil, and H. R. Mamatha

Abstract Reasoning ability along with the skill of arriving at logical inferences is central to artificial intelligence and a key requirement to get one step closer toward natural language understanding. Once such task which involves learning this skill is natural language inference (NLI) which involves the task of categorizing a given premise and hypothesis into one of three classes—neutral, entailment or contradiction. The availability of datasets such as the Stanford Natural Language Inference (SNLI) dataset [1] has allowed the use of deep learning networks to tackle this problem. The paper presents the results of two models: (1) LSTM model and (2) LSTM with self-attention where both models are trained on different subsets of the SNLI dataset. The paper tests and compares the performance of both models on different subsets of the dataset containing varying proportions of entailment, contradiction and neutral labels and provides an analysis to explain possible reasons for incorrect predictions.

Keywords Natural language inference · Long short-term memory · Recurrent neural networks · Stanford Natural Language Inference dataset · Natural language understanding · Self-attention

N. Kumar (✉) · A. V. Ragam · G. Akhil · H. R. Mamatha
Department of CSE, PES University Bangalore, Bangalore, India
e-mail: nidarshank@pesu.pes.edu

A. V. Ragam
e-mail: anirudhvragam@pesu.pes.edu

G. Akhil
e-mail: gbs.akhil@pesu.pes.edu

H. R. Mamatha
e-mail: mamathahr@pes.edu

1 Introduction

Replicating human-level understanding, inferences and logical reasoning in machine learning and deep learning models has always been a challenging task. The ability to reason and arrive at a valid inference is a fundamental step toward building artificial intelligence systems that are capable of human-level understanding and making human-level deductions.

Natural Language Inference (NLI) is one such task which is used to determine the relationship between a given premise (p) and hypothesis. The given premise and hypothesis can either agree or contradict with each other, or remain neutral, which means they are not related (h).

p : *The family is currently outside watching a baseball game.*
 h : *The family are eating at a restaurant.*

The above pair of sentences contradict each other as the *family* are involved in two completely different activities. Therefore, it is an example of the contradiction class.

p : *A boy is playing the guitar at the festival.*
 h : *A boy is playing an instrument at a grand stage*

The above pair of sentences are in agreement with each other. Therefore, it is an example of the entailment class

p : *A girl is playing the guitar along with a group at the music academy.*
 h : *The group at the music academy are playing with harmony.*

The above pair of sentences are neither related to each other strongly nor completely contradict each other. Therefore, it is an example of the neutral class.

Natural language inference is an important problem that has applications in a wide range of NLP tasks. For instance, Chen et al. [2] propose a method to verify whether the answers produced by a question–answering system are actually correct, using natural language inference, as NLI requires the premise, which in this case is the question, and hypothesis, which is the answer to the question, to agree with each other. Apart from question–answering systems, NLI also has wide range applications in other NLP problems such as semantic search, semantic parsing, machine translation and so on. The two of the most prevalent datasets for NLI are the Stanford Natural Language Inference (SNLI) and Multi-Genre Natural Language Inference (MultiNLI) [3] datasets, both of which have been important contributions to this problem, as they have facilitated experimentation with more sophisticated deep learning architectures. In this paper, we attempt to train two different models on the SNLI dataset with varying proportions of entailment, contradiction and neutral labels to test

and compare the performance of the models on different subsets of the dataset. This is followed by error analysis to identify mislabelled examples and further improve the performance of the models.

2 Literature Survey

Chen et al. [4] combine external knowledge with the existing NLI models to achieve a higher performance. Existing approaches make an assumption that the data which is used for training contains all the necessary information to make the decision. However, in some examples the context between words is of great significance. The two main models for NLI are sentence encoding and self-attention networks. Sentence encoding involves the encoding of the premise and hypothesis followed by classification. State-of-the-art NLI models follow a particular template. Firstly, the premise and hypothesis are processed to determine appropriate representations. Local inference information is collected to finally arrive at a sentence-level decision. External knowledge is incorporated at the attention layer, local inference collection and final aggregation and composition layer. External knowledge is mainly incorporated using synonymy, antonymy, hypernymy and hyponymy.

Parikh et al. [5] propose performing NLI using an attention model. The approach relies on the alignment of sub-phrases of the hypothesis and premise against each other to determine whether they agree with each other. The main problem is divided into multiple sub-problems of aligning sub-phrases to determine the final outcome, thereby enabling parallelization. A soft alignment matrix is generated using neural network based attention, from the two sentences represented by their embedding vectors. The problem can be decomposed into sub-tasks, each of which can be solved in parallel. The authors have reported that the decomposable attention model though simpler outperforms other more complex neural methods like LSTMs. The attention model is also computationally less expensive than the other models owing to its parallelization capabilities.

Nie et al. [6] present a novel dataset called Chaos-NLI dataset for NLP. According to the analysis done, the models achieve a high accuracy only when the agreement level of the data is high. The model performs very poorly when the agreement level of the subset is low. Chaos NLI attempts to do that by using collective human opinions for examples in several existing NLI datasets. The paper finds out the dependence of human agreement on the existing state-of-the-art model performances. The results show that the models are unable to capture the human distribution. As a result, the models do well only when there is high level of human agreement in the dataset. Chaos NLI attempts to incorporate collective human opinion by providing hundred annotations for each example in three sets of existing NLI datasets such as SNLI. The results showed a significant difference between model output and collective human opinion. The usage of collective distribution for evaluation allows the researchers to trust the decision made by the model and it also helps in allowing the model to capture the collective human opinion.

Dropout has been used with RNNs [7] for various applications such as machine translation, handwriting recognition and neural language models. All the existing work have established that dropout is not suitable with RNNs as it affects the long-term dependencies of the input sequences. Existing work on the use of dropout with RNNs for handwriting recognition revealed that the performance varied depending on which layer dropout was applied. It was also concluded that the application of dropout after recurrent layers or between feed-forward layers led to poor results. The paper presents a RNN model for NLI which incorporates the use of intra-attention to capture the context of the input sequences in a much better fashion. The input sentences are passed through an embedding layer, and this is followed by a BiLSTM layer. The application of dropout on this model was successful in avoiding overfitting. The model trained on the SciTail dataset was able to prevent over fitting by using dropout along with a simpler model. This was done by reducing the number of hidden units from 300 to 100 resulting in a reduction in the number of parameters. Some of the main inferences drawn by the author are that regularization is a must for the embedding layer especially when it used for larger datasets such as the SNLI dataset. For a smaller dataset, the paper suggests the application of regularization in the recurrent layers. The performance of the model dropped when regularization was applied at the intermediate feed-forward layers.

Yang et al. [8] propose using a Hybrid-Siamese Convolutional Neural Network architecture to address the dataset imbalance issue. The imbalanced dataset is composed of two types of classes—head categories and tail categories. Head categories are those which have a relatively large number of instances in the dataset. Tail categories on the other hand have a relatively low representation in the dataset. The architecture consists of two parts: A single convolutional neural network and a Siamese network. The single CNN is used to classify text into head categories, whereas the Siamese network, which is a few-shot learning technique, is used for classification of text into tail categories. The authors report a significant improvement in performance and accuracy, when using the HSCNN architecture to classify categories that are less represented in the dataset.

Wang and Jiang [9] was based on improving upon the existing model given by Rocktäschel et al. [10]. The baseline model uses sentence embeddings for both the hypothesis and premise sequences. The only difference is the inclusion of attention weighted representation of the premise in order to increase the accuracy of the prediction. All word-level relations were considered even though some of them including stop words were not important with respect to the decision made by the model. The NLI model uses LSTM cells to perform word-level matching between the premise and hypothesis. At every point, model tries to match the attention weighted representation of the premise with the current word in the hypothesis which resulted in remembering only the important matching results. In particular, important mismatches which often indicates neutral or a contradiction relationship intend to be remembered and good word-level matching results were forgotten.

3 Proposed Methodology

3.1 Dataset

The model has been trained on the SNLI dataset [1] which consists of approximately 550,000 training samples and 10,000 of each testing and validations samples. The dataset consists of the following in JSON format:

sentence_1_binary_parse: Premises without POS tags

sentence_2_binary_parse: Hypotheses without POS tags

sentence_1_parse: Premises with POS tags

sentence_2_parse: Hypotheses with POS tags

label-1 to label-5: Labels given by five annotators to the current premise-hypothesis pair

gold label: Most frequently occurring label for the current pair

pair_id: Unique ID for each pair

sentence-1: Premise sentence

sentence-2: Hypothesis sentence

For our model, we have considered sentence_1_binary_parse, sentence_2_binary_parse, pair-id, gold label as the primary labels. The hypothesis and premise sentences were extracted from the respective binary parse representations. Training and testing was first done with the complete dataset, and it was later on done with a disproportionate dataset built with having 250,000 samples consisting of user-defined proportions for each class to check the accuracy of the classification with imbalanced proportions of each class in the dataset.

3.2 Preprocessing

Firstly, tokenization of the data is performed using keras library. The word index is built using the relative importance of all the unique words in the corpus. Each word is assigned a unique integer based on the frequency of occurrence where a lower number implies higher frequency. Therefore, stop words tend to have lower word index numbers. The premise and hypothesis are encoded using the Glove (300d) embedding. Embedding matrix of dimensions $vocab_size \times 300$ is created, and this contains only the words present in our vocabulary and their corresponding embedding vectors derived from the glove file. Finally, the embedding matrix is saved into a file for later use.

3.3 LSTM Model

Figure 1 presents the architecture of the LSTM architecture which primarily builds upon the architecture proposed by Bowman et al. [1]. The tuning and use of attention is inspired by Liu et al. [11]. The premise and hypothesis are first passed through an embedding layer to obtain the the 300-dimensional GloVe representation for each word in both the sentences. The premise and hypothesis are then separately passed through an LSTM layer with three hundred units followed by a batch normalization layer for regularization. The premise and hypothesis are then concatenated and fed through three dense layers, having three hundred units each, with the RELU activation function. The final prediction layer contains three units with a softmax activation to obtain the probabilities of each of the three classes. The loss objective on which the model has been trained is categorical cross entropy, and the model has been tested with different optimizers such as Adam and RMSProp. Early stopping has been used to make sure that the training process stops if the validation accuracy changes negligibly for 4–10 continuous epochs based on optimal performance.

3.4 LSTM Model with Self-Attention

Figure 2 presents the architecture of the LSTM model with self-attention. The base model primarily builds upon the architecture proposed by Bowman et al. [1] and the use of attention is inspired by Liu et al. [11]. The premise and hypothesis are

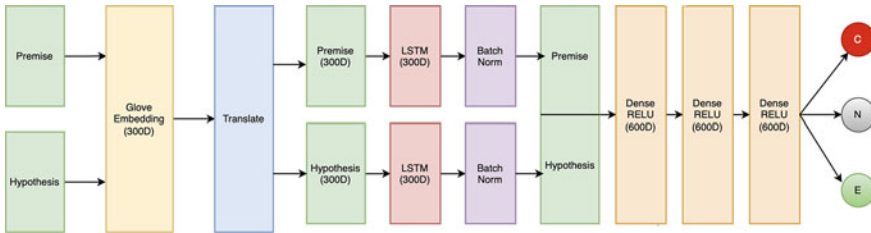


Fig. 1 Architecture of the LSTM model

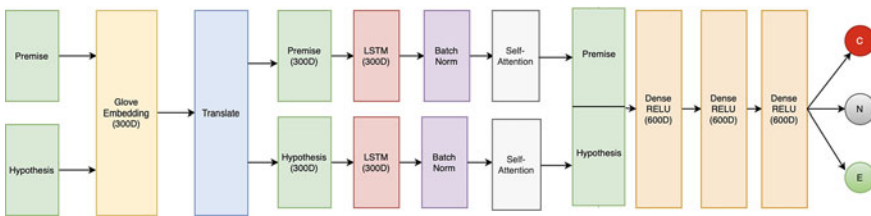


Fig. 2 Architecture of the LSTM with self-attention model

passed through the Embedding layer, followed by the translate layer and LSTM layer with batch normalization. The premise and hypothesis are passed individually passed through an attention layer to capture the relationship within the sentences. The rest of the architecture follows the LSTM model leading up to the three-way Softmax output.

4 Results and Discussions

The models use early stopping to maximize the validation accuracy and avoid overfitting. The original LSTM model was trained for 8 epochs with a final train accuracy of 88.30% as given in Table 1. The training accuracy can go up to 97% if trained for more number of epochs. However, this led to poor validation and test performance. The validation and test accuracy peaked at 80.38% and 80.06%, respectively. The LSTM with self-attention model was trained for 15 epochs with a train accuracy of 82.47%. The validation and test accuracy was 75.45% and 75.55%, respectively.

To test the model on imbalanced dataset, a subset of the dataset was used to train both the models. The subset had the following ratio of labels: 60% entailment, 20% contradiction and 20% neutral. This model was also trained for 8 epochs to maximize the validation accuracy and avoid overfitting. The training accuracy improved to 90.09 due to the imbalance of the dataset. The validation accuracy reduced to 75.42% and the test accuracy reduced to 74.51%. When compared to the original dataset, the model was able to give a similar performance with slight reduction in validation and testing accuracy. The LSTM model with self-attention gave a better test accuracy on the imbalanced dataset as seen in Table 2.

Table 1 Performance of both models on complete SNLI dataset

Model name	Train accuracy	Validation accuracy	Test accuracy
LSTM	88.30	80.38	80.06
LSTM with Self-attention	82.47	75.45	75.55

Table 2 Performance of both models on imbalanced SNLI dataset

Model name	Train accuracy	Validation accuracy	Test accuracy
LSTM	90.09	75.42	74.51
LSTM with Self-attention	77.90	70.19	77.48

4.1 Error Analysis

In order to improve the test accuracy of the models, the misclassified examples were observed to check for any visible patterns. One of the reasons was the consideration of “.” as a separate token. The relative importance of the “.” token was the highest due to higher frequency among other tokens. This caused the model to change its prediction label just based on the presence and absence of the “.” token rather than looking at the word-level features of the sentence. The removal of the “.” token reduced the number of misclassified examples. The SNLI dataset was also found to contain a lot of sentences with spelling errors. The difference in spelling caused the entire context of the sentence to change causing a change of the predicted label from the original label. Example *p*: The girls walk down the street and *h*: Girls *sat* down in the street. Is an example that should come under the contradiction label. However, the training set contained the hypothesis as *h*:Girls *set* down in the street. This has caused the model to classify this pair of sentences as an entailment. The performance of both models can thus be improved by adding spell check as an additional preprocessing step.

5 Conclusions and Future Work

This paper presents the performance of both models on varying proportions of the SNLI dataset and highlights the differences. The LSTM model gave a better performance in all scenarios. The performance of both the models can be improved with additional preprocessing steps such as correcting the semantics of all the sentences before training the model. The paper extends both the models with varying proportions of labels. The similar performance on the imbalanced dataset indicates that this model can generalize well. The reduction in validation accuracy is compensated by the simplicity and the ease of use of the model compared to the complex architectures involving transformers. The usage of self-attention along with more sophisticated preprocessing techniques is one of the future directions of this work.

References

1. Bowman SR, Angeli G, Potts C, Manning CD (2015) A large annotated corpus for learning natural language inference. In: Proceedings of the 2015 conference on empirical methods in natural language processing, Lisbon, Portugal. Association for Computational Linguistics, pp 632–642
2. Chen J, Choi E, Durrett G (2021) Can NLI models verify QA systems' predictions? CoRR, abs/2104.08731

3. Williams A, Nangia N, Bowman SR (2018) A broad-coverage challenge corpus for sentence understanding through inference. In: Proceedings of the 2018 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long papers), New Orleans, Louisiana. Association for Computational Linguistics, pp 1112–1122
4. Chen Q, Zhu X, Ling ZH, Inkpen D, Wei S (2018) Neural natural language inference models enhanced with external knowledge. In: Proceedings of the 56th annual meeting of the association for computational linguistics (volume 1: long papers), Melbourne, Australia. Association for Computational Linguistics, pp 2406–2417
5. Parikh A, Täckström O, Das D, Uszkoreit J (2016) A decomposable attention model for natural language inference. In: Proceedings of the 2016 conference on empirical methods in natural language processing, Austin, Texas. Association for Computational Linguistics, pp 2249–2255
6. Nie Y, Zhou X, Bansal M (2020) What can we learn from collective human opinions on natural language inference data? In: Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP), Online. Association for Computational Linguistics, pp 9131–9143
7. Gajbhiye A, Jaf SF, Al Moubayed N, Stephen McGough A, Bradley S (2018) An exploration of dropout with rnns for natural language inference. CoRR, abs/1810.08606
8. Yang W, Li J, Fukumoto F, Ye Y (2020) HSCNN: a hybrid-siamese convolutional neural network for extremely imbalanced multi-label text classification. In: Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP), Online. Association for Computational Linguistics, pp 6716–6722
9. Wang S, Jiang J (2016) Learning natural language inference with LSTM. In: Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies, San Diego, California. Association for Computational Linguistics, pp 1442–1451
10. Rocktäschel T, Grefenstette E, Hermann KM, Kociský T, Blunsom P (2016) Reasoning about entailment with neural attention. CoRR, abs/1509.06664
11. Liu Y, Sun C, Lin L, Wang X (2016) Learning natural language inference using bidirectional LSTM model and inner-attention. CoRR, abs/1605.09090

Robotics Process Automation Implementation in Project Management



Sharad Garg, Pooja Dehraj, and Ritvik Shrivastava

Abstract Technological developments have aided businesses in increasing their efficiency throughout history. Organizations may now use robotic process automation to improve the efficiency of knowledge-based labor (RPA). With this potential in hand, organizations must figure out how to successfully and efficiently integrate RPA into project management. As a result of automation in project management, human employees' attention and effort have dropped, allowing for more production per person. As a result, the company would be able to grow economically and technologically. Automation revolutionized industries that required a great deal of manual labor. The RPA concept in project management has a commercial influence on companies since it speeds up the project management process and eliminates the need to repeat activities. RPA bots are capable of executing several tasks at the same time and communicating with software and they can be used successfully, in operational operations that demand consistency, volume, and repetition.

Keywords Project management · Robotic process automation · RPA in project management · RPA use cases in project management · RPA in execution

1 Introduction

A project is a set of inputs and outputs required to achieve a particular goal. According to the Project Management Institute (PMI), the term Project refers “to any temporary endeavour with a definite beginning and end”. Depending on its complexity, it can be managed by one person or a hundred and it can range from simple to complex. The characteristics of a project are:

- A project has a clear start and end date.

S. Garg (✉) · P. Dehraj · R. Shrivastava
Computer Science and Engineering Department, Noida Institute of Engineering and Technology,
Greater Noida, India
e-mail: sharadgarg055@gmail.com

P. Dehraj
e-mail: drpooja.cse@niet.co.in

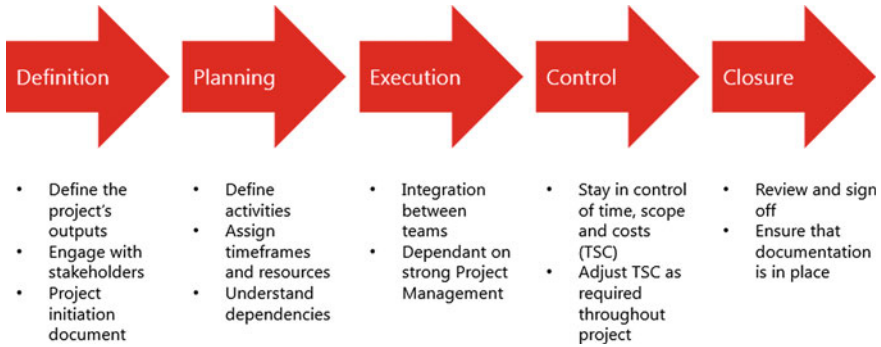


Fig. 1 Five phases of the project management [Source TCS Project]

- A project has boundaries
- A project creates something new
- A project is not boundaries as usual.

A well-defined “Project Management” is very important in an organization for managing the goods and services. Project management is a process of planning; organizing and managing meet the project and client’s requirements. A Project Management can be started as an application of knowledge, skills, tools, and techniques that can be applied on the project activities to achieve the project objective. Project management processes are divided into five steps as shown in Fig. 1 [1].

1.1 Robotics Process Automation and Its Benefits

Robotics Process Automation (RPA) refers to specific software where program can be done with much difficulty to do normal tasks across applications, which are very similar to the performance by the human workers. RPA is an advanced technology of the twenty-first century that enables anyone to build computer software or a “robot” to perform human actions using AI. Using the RPA, the boot communicates the digital system and helps in executing the business process. RPA technology automates the time-consuming, repeatable manual actions [2]. RPA enables the projects and business processes to run efficiently by eliminating human error. RPA, also known as intelligent process automation has been around for years. In today’s time, around the globe millions of hours of staff in customer service, business support, and operations are being working manually. In project management, the Project Manager works in conjunction with the IT automation manager. The project manager’s ultimate goal is to make suggestions to the automation leadership team to deploy RPA successfully in the most effective and efficient manner throughout the organization. RPA may be a huge change for the project manager, especially when the project managers are used to managing the multi-sprint deployment.

Robotic Process Automation is getting more and more attention; it's being adopted and implemented in organization of every industry [3]. But there are some decision makers who think whether RPA is worth the investment. Let's take a look at a few of the many benefits RPA can provide to the organization. Some of the benefits of RPA in project management are:

- It's very efficient [4].
- Less manual errors: Rather of using keyboards or select-copy-paste, robots communicate data using element or image recognition [4].
- Cost effective: It's cost effective as compared to other option [4].
- Scalability: It is easier and less expensive to duplicate robots than it is to hire and train fresh people within the company [4].
- Simple solutions: Upgrades are straightforward to get, and they may work with existing applications without the requirement for costly or time-consuming internal system rebuilding [4].
- Staff Wellbeing: Using the RPA to perform the rule-based tasks on your project and it will minimize the amount of time staff needs to perform mundane data-entry and administrative tasks [4]. When the team members focus on the tasks that challenge them, the risk of burnout is reduced and team member's wellbeing is improved.

2 Requirement of Robotics Process Automation in Project Management

Projects are not always successful. In a report published in 2017 from the Project Management Institute (PMI), 14% of IT projects fail [5]. However, this number only represents the total failures of the projects that didn't fail outright, 31% projects didn't meet their goals, 43% projects exceeded their initial budgets, and 49% projects were late. Most common reasons for IT project failure are as shown in Fig. 2 [6].

Many companies are facing a staggering rate of retirement, which means a lot of great talent is walking out of the box. Nowadays, Robotic process automation is one of the best investments in the IT industry. With increasing technology, Robotic Process Automation (RPA) is becoming a part of daily life from online shopping and data transfer to customer registration. RPA increases the attention of corporate world [7] and industrial automation over the past couple of years [7]. Australian organizations are increasingly looking to Robotic Process Automation (RPA) to improve operational efficiency, productivity, and quality and customers satisfaction. Over 54% of Australian organizations [8] have reported cost savings from automation systems and processes and seen an effective result in quality and compliance. According to a 2018 KPMG survey, nowadays companies are significantly investing in RPA software, more than they are in cloud, IoT. Investment in RPA in different field for saving the operational cost is shown in Fig. 3 [9].

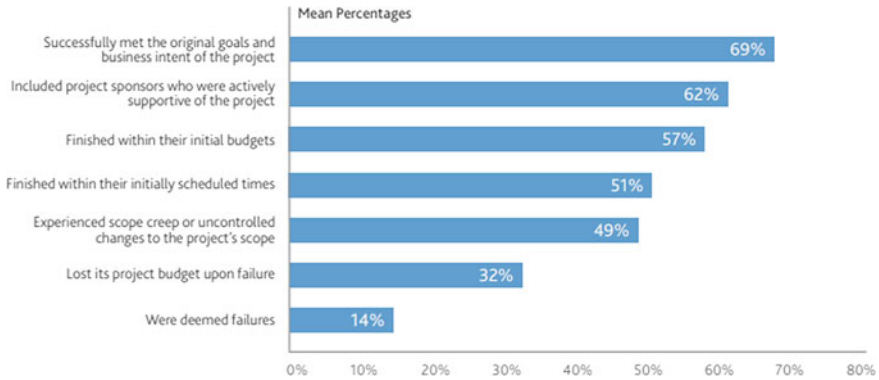


Fig. 2 Leading causes of IT project failure. [Source Atspoke]

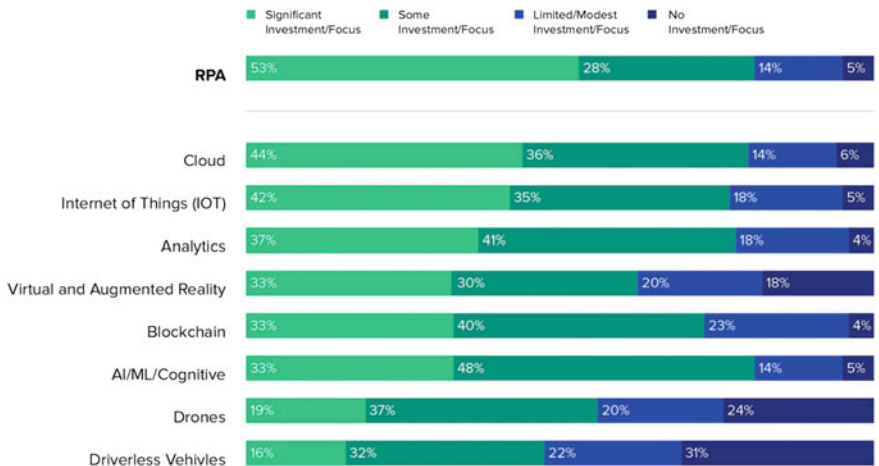


Fig. 3 Investment in different field for saving the operational cost. [Source Toptal]

Using the software tool powered by RPA in project management [10], the project managers can easily calculate the risks of layering up and are familiar with the reduction in operating costs they can expect. RPA has high scope to transform the project management success. Reduction, the errors in performance, and increase staff wellbeing help the team members to focus more on creative tasks and offers a great customer experience. In conclusion, we can expect the role of RPA in project management to keep on growing [10]. In project management, Robotic Process Automation (RPA) is taking on greater importance, with software bots standing out as essential assistants; it is a fundamental technology that automates structured and repeatable processes to run with greater efficiency and productivity [11]. RPA software solutions automate increasingly complex tasks and projects unlike traditional software tools that manage simplified, basic processes [12].

3 Robotic Process Automation in Project Management Lifecycle

Using Automation in project management helps the project manager in handling the regular reports, data collection and transfer, scheduling, and notifications triggered by events in the project management [13]. A successful project is one that is delivered “on time, within budget and with the required quality”. In most of the organization, Jira is widely used in the project management for collecting and generating the project reports but it takes a lot of work when performed manually. RPA provide the solution on different stages of the project lifecycle, to simplifying repetitive tasks [10].

3.1 Project Initiation

Project Initiation is the first stage of the project management lifecycle. This stage is very important to start the project in a right way, as it involves starting up a new project where an abstract idea turns into a meaningful goal. In this stage, the business complications or opportunity is point out and based on that a solution is spell out for which the businesses cases are developed to define the project on a broad level based on the requirement gathering, which is very important not only for software project but for all types of projects [14]. It is also known as Requirement’s elicitation or Capture; it is described as the process of gathering needs from multiple stakeholders (functional, system, technical, etc.). (Customers, users, vendors, IT staff, and etc.). RQ12PA might become more extensively used at this time since Bots can continuously gather and combine data, freeing up team members to focus on analysis [13].

3.2 Project Planning

Project planning is an organized step by step manner and requires complete diligence to draw the project’s roadmap [15]. In project planning, we can use RPA widely. Because, RPA depends on the historical data and can be a powerful tool in creating predictive models based on the historical performance of similar data and by seeing the processes visually the project manager would be able to take the appropriate decision because using the RPA methodologies in the project management, the project manager will have the most accurate recent update. Using the RPA, the project manager can do the easily monitored and managed the risk related to the project. When few repetitive activities in the project management are automated, it can be easily tracked which helps to reduce risk even further [16, 17]. This technique also saves time and prevents bottlenecking, giving employees more time to focus on project-specific decisions that improve delivery.

3.3 Project Execution

In this stage of the project management, the team does the actual work. It is critical for a project manager to develop effective processes and constantly monitor the team's progress throughout this phase especially during the COVID19 pandemic, in the current situation, most IT businesses are working from home during the COVID19 epidemic [15, 18] and they are confident in continuing with this paradigm of work from home (WFH). So for a project manager it's very challenging to maintaining a proper channel of communications and data reporting and to make sure it shouldn't slow down. Using the RPA technology, it automated the forms and allows the team members to quickly and accurately submit information. For a successful project execution, real-time reporting is very important. This is important not only for tracking the progress of the team but for the further decision making as well. Using the RPA, the project manager can have the real-time reporting and automatically created reports, assuring project managers that team members are accessing the most recent version of the necessary information [7, 17].

3.4 Project Monitoring and Controlling

Project monitoring and controlling runs simultaneously with the project execution to ensure that the objectives and project deliverables are met [15]. For proper monitoring and controlling, testing is very important. Using the RPA software, the team or the project manager can test the functionality of user interface elements, data input, process flows, API calls, integration, and system response of the product which is being developed. Using the RPA technique, the project can take immediate action on any fluctuations in the project progress [17]. For a project manager proper monitoring and controlling of both the project and the team members is very important and RPA makes this task very easy because any changes in the project will be notified to the project manager, which results in eliminating any barrier in the project success [7].

3.5 Project Closure

Project closure is the last phase in the project management lifecycle. In this phase, the development company passing the documentation to the customer, transfer all the deliverables, review all contracts and documentation, rerelease resource and support the client post launching or closure [15]. Nowadays, software is upgraded into RPA integrated with intelligent system that includes features such as, image recognition, text analytics, optical character recognition, sentiment analysis, and natural language processing. Using the RPA, the companies have started to build the smarter and responsive chat-bots [7]. This plays as very important during the maintenance phase

of the software development lifecycle [17]. RPA may be used by project managers and upper management to analyze operational metrics, which give an in-depth picture of the project's performance in terms of productivity, budget, and efficiency, and these important metrics from prior projects can be utilized to enhance and optimize future initiatives.

4 Other RPA Use Cases in Project Management

4.1 Stakeholder Communication

Existing process visuals should be shown clearly and transparently so that all stakeholders can understand how any modifications, if any, would impact them. Capture new project processes, keep all stakeholders informed, deliver notifications to their dashboards, and alert them to any necessary steps.

4.2 Process Risk Management

Manage and monitor risks in the process management platform, and create an environment where potential threats may be tracked.

4.3 Data Driven Progress Management

Assist project managers in submitting data-driven project progress estimates using process maps. Streamline operations, increase efficiency, and boost staff morale.

4.4 Process Updates and Documentation

Within a process management platform, you may update processes and save relevant documentation. Ensure that everyone in the company is up to date during the changeover.

4.5 *Process Improvement Initiations*

As teams evaluate current practices and seek methods to improve them, assist project managers in staying on top of the change requests they make.

5 **Process to Estimate Cost Benefits**

Wibbenmeyer defined the method of calculating the cost–benefit calculation of what RPA implementation could mean to the organization in her book, “The Simple Implementation Guide to Robotics Process Automation (RPA)”. It is an extremely manual process that multiple people perform in the organization. There are some basic questions that we should ask:

- How does the process change frequently?
- Is this a rules-based process?
- Number of the Human Resource work on it?
- How many hours will they work on it a year?
- Is the process a time-intensive non-value-added activity? Do the people performing the work get bored easily?

How people are prone to mistakes due to let’s take an example: If the task takes 10 people 8 h a day at a rate of \$50.00:

$$10 \times 8 \times \$50.00 = \$4000.00\text{perday}$$

Since, people work 2080 h per year (2080/8 h per day), this means approximately 260 workdays per year. At an hourly rate of \$4000 and 260 work days at 0.5 (the process takes everyone half a day = 130 days), there is estimated savings of \$1,040,000 for automating this process.

Another example is to take a task that takes three people two hours a day at a rate of \$50 per person:

$$3 \times 2 \times \$50.00 = \$300\text{perhour}$$

2 h a day is: 25 of a day = $0.25 \times 260 \text{ days} = 65 \text{ full days of work} \times \$300.00 \text{ an hour} = \$19,500.00$ Even a few small firms will easily get the cost benefit to \$100,000 in savings.

6 Challenges and Solution in Adopting Robotics Process Automation

RPA helps businesses to automate their repeated operation tasks and saves the time and the cost that it would have taken previously. RPA is also non-intrusive, it can be implemented using current infrastructure without disrupting underlying systems, which would be difficult and expensive to replace. Expense savings and compliance are no longer an operational cost when RPA is used; instead, they are a consequence of the autonomic process [4].

Robotic Process Automation RPA has changed many industries in the last couple of years. It is a remarkable, rapid change that has changed many industries; also, it is driving a colossal increase in productivity among the organization [4, 19]. Nowadays, it is providing automation solutions. It is helping individual companies to enhance their operation style and helping them to automate the tasks that need more focus and skills. RPA enables the bots help in executing the repetitive and predictable tasks having a high volume of data. It focuses on capturing data, interpreting responses, and then effectively communicating with multiple devices strategically. Unfortunately, there are same challenges while implementing RPA in an organization. Some challenges [20, 21] and their solutions are mentioned in the Table 1.

7 Discussions and Conclusion

The term “digital transformation” describes a long-term strategic strategy aimed at streamlining processes and increasing employee productivity. The effectiveness of digital transformation methods in the workplace is determined by management’s capacity to manage projects and teach employees to adapt to new work patterns. The key to success and decreasing the risk of project execution failure is always good planning. This organization’s RPA implementation process is also effective, according to the findings of this study, because it has done prior planning by standardizing and managing their resources to guarantee a smooth new project implementation process.

Changes in employment and the fear of losing one’s job have a detrimental impact on and can even derail automation efforts. As a result, the company’s management team has remarked that in order for RPA deployment to operate smoothly and efficiently, they must have excellent communication and change management skills with employees at all phases. Respondents also claimed that excellent two-way communication from top management to them helps them accept work changes in this company. To ensure that the RPA in project management works easily and effectively, every stage must be communicated to the entire team. If team members understand each aspect of the implementation process, including the risks that may be involved, the firm’s margin of error may be reduced, if not eliminated.

Table 1 Challenges and their solution in RPA

Factor	Challenges	Best Practices
Process selection	<ul style="list-style-type: none"> • Difficult to automate high level process • Lack of process standardization 	<ul style="list-style-type: none"> • Start out with a limited path rather than trying to automate a process end to end • Integrate with a process orchestrator • Optimize processes first
Availability of relevant and sufficient talent	<ul style="list-style-type: none"> • Shortage of skills • Lack of practical knowledge to implement AI 	<ul style="list-style-type: none"> • Leverage vendors to boost training • Speed training by bringing in external expertise
Organizational readiness	<ul style="list-style-type: none"> • Managing organizational change stakeholder buy-in • Concerns from IT 	<ul style="list-style-type: none"> • Treat implementation as change programs and use appropriate change methodologies
Lack of sufficient data to train AI	<ul style="list-style-type: none"> • Sufficient data to train AI to attain required accuracy level • Extracting this data and making it relevant for training purposes 	<ul style="list-style-type: none"> • Tap into existing but anonymized, data
Transparency in AI decision making	<ul style="list-style-type: none"> • Ethics and transparency of AI • Tracing the path taken by AI software 	<ul style="list-style-type: none"> • Allay concerns with supervised learning
Implementation	<ul style="list-style-type: none"> • Scaling up smart RPA • Developers and business people lacking smart RPA skills 	<ul style="list-style-type: none"> • Be realistic about timescales and collaborate widely

RPA is a powerful digital transformation technology that, when used in conjunction with the correct goals and KPIs, may significantly improve a company’s performance. As a result, significant support and motivated training are essential for robotic technology to be used effectively.

References

1. <https://tscprojects.co.uk/blog/the-five-stages-of-a-project/>. Accessed on 7 May 2021
2. Timbadia DH et al (2020) Robotic process automation through advance process analysis model. In: 2020 international conference on inventive computation technologies (ICICT). IEEE
3. Pérez-Morón J, Marrugo-Salas L (2021) Robotics in shared service centers in emerging economies. In: IOP conference series: materials science and engineering, vol 1154, No 1. IOP Publishing
4. <https://www.uipath.com/rpa/robotic-process-automation>. Accessed on 15 May 2021
5. <https://www.linkedin.com/pulse/why-do-information-technology-projects-fail-blair-turley-mba/>. Accessed on 3 June 2021
6. <https://www.atspoke.com/blog/it/reasons-for-it-project-failure/>. Accessed on 21 May 2021
7. <https://www.ibm.com/cloud/learn/rpa>. Accessed on 15 May 2021

8. https://www.accenture.com/t00010101T000000_w_/au-en/_acnmedia/PDF-36/Accenture-16-4014-Robotic-Process-Auto-POV-FINAL.pdf. Accessed on 3 June 2021
9. <https://www.toptal.com/project-managers/it/robotic-process-automation-for-project-managers>. Accessed on 7 May 2021
10. <https://www.forbes.com/sites/forbestechcouncil/2019/12/02/how-to-use-rpa-and-ai-for-project-management/?sh=7986bebe4bd1>. Accessed on 15 May 2021
11. Balasundaram S, Venkatagiri S (2020) A structured approach to implementing robotic process automation in HR. *J Phys Conf Ser* 1427(1). IOP Publishing
12. Siderska J (2020) Robotic process automation—a driver of digital transformation? *Eng Manage Prod Serv* 12(2)
13. <https://project-management.com/robotic-process-automation-for-project-management/>. Accessed on 7 May 2021
14. <https://opentextbc.ca/projectmanagement/chapter/chapter-7-project-initiation-project-management/>. Accessed on 21 June 2021
15. <https://kissflow.com/project/five-phases-of-project-management/>. Accessed on 15 May 2021
16. <https://kissflow.com/project/steps-to-create-successful-project-plan/>. Accessed on 21 May 2021
17. <https://www.nintex.com/blog/the-4-phases-of-project-management-and-how-process-automation-can-help/>. Accessed on 21 May 2021
18. Gallego JS, Ortiz-Marcos I, Romero Ruiz J (2021) Main challenges during project planning when working with virtual teams. *Technol Forecast Social Change* 162:120353
19. Rajawat AS et al Robotic process automation with increasing productivity and improving product quality using artificial intelligence and machine learning. *Artif Intell Future Gen Robot Elsevier* 1–13
20. Dey S, Das A (2019) Robotic process automation: assessment of the technology for transformation of business processes. *Int J Bus Process Integr Manag* 9(3):220–230
21. Choi D, R’bigui H, Cho C (2020) Robotic process automation implementation challenges. In: *International conference on smart computing and cyber security: strategic foresight, security challenges and innovation*. Springer

A New Hybrid Boost Converter with Cuckoo Search MPPT for High Gain Enhancement of PEMFC



CH. Siva Kumar and G. Mallesham

Abstract To reduce the carbon emissions from the conventional energy sources, depending on the imported fuels and to reduce the gap between the generation and demand, the on-site power generation systems called the renewable energy systems have a hedge against financial risks, improve the reliability of the system with a quality of power. The new diversified technologies of fuel cell systems are being used as distributed power generating systems, input to the electrical vehicles and to charge the electrical vehicular system due to its environmental friendliness, higher efficiency and a wide range of production of electrical power. But these fuel cells have the limitation of voltage and current to integrate with the grid. Under these conditions, the power electronics-based converters are playing a major role to control and extract the maximum utilization of the renewable energy systems. In this work, built a proton exchange membrane fuel cell with a cuckoo search method to extract maximum power from the fuel cell system. Further, the work is concentrated to model a new hybrid boost converter with high gain with the help of MATLAB/Simulation. The results shown the effectiveness of the cuckoo search MPPT and the hybrid high gain boost converter.

Keywords Renewable energy systems · Fuel cells · Converters · Proton exchange membrane · Hybrid boost converters

1 Introduction

A modern electrical power system shown in Fig. 1 is the combination of conventional energy sources, non-conventional energy sources, residential loads, commercial loads, sensitive loads, energy storage systems, industrial loads, electrical trans-

CH. Siva Kumar (✉) · G. Mallesham

Department of Electrical Engineering, University College of Engineering, Osmania University, Hyderabad, Telangana 500007, India

e-mail: sivakumar.ch@uceou.edu

G. Mallesham

e-mail: drgm@osmania.ac.in

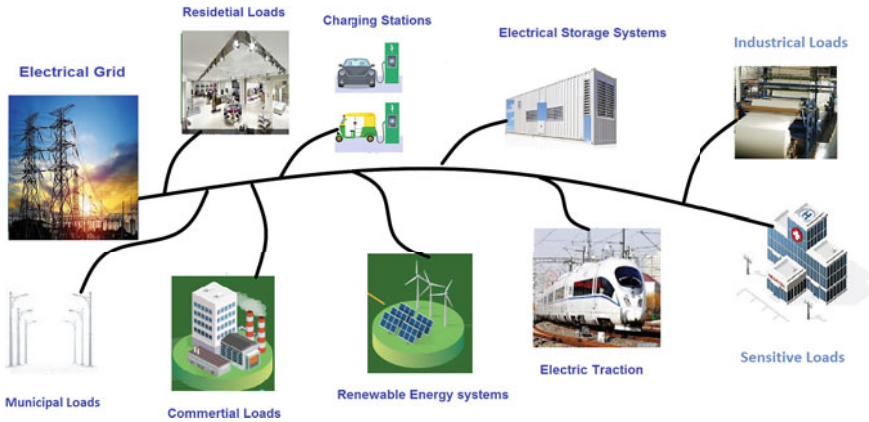


Fig. 1 A modern electrical power system

portation systems road and rail and the electric charging stations, etc. To meet the demand, to reduce the gap between the generation and demand, effective utilization of the available non-conventional energy source for a clean environment the contribution of renewable energy sources is increasing day by day [1]. Along with the wind and solar PV, there are certain other non-conventional methods like fuel cells, biomass energy, geothermal energy, tidal energy, etc. [2–5] which lead their contribution in power sector. The new policies and the reforms introduced in the electrical power sector, encouraging the percentage of contribution of private players into the market. It strengthens the economic efficiency, to be more energy security and to improve the economic growth of the country. India has made an effective improvement in introducing the new and renewable energy systems in to the power sector with lower carbon emission well below the global average value 4.4 tonnes an increase in its per capita consumption [6]. The installed generating capacity of renewable energy is on 31st May 2021 of 383.37 GW out of which 95.7 GW of non-227 GW by 2022 India Energy Outlook and Ahluwalia [7, 8]. The salient features of fuel cells due to increase in advancements and the salient features like efficiency, silent in operation increasing the importance in generation of electrical energy [9]. The fuel cells are classified based on the state of the electrolyte used, temperature, and the power capacity, etc. The most common type of high-density fuel cells is proton exchange membrane fuel cell. Table 1 presents the details of different types of fuel cells and their comparisons.

The output voltage levels of fuel cell range from a smaller value to a higher value. To connect these lower level voltage systems to grid is possible with the help of switched mode power electronics-based converter. Each technique of control circuit has its own advantages and disadvantages. To achieve the highest with pulsating currents and poor regulation, switched capacitor-based DC–DC converter places a major role. Researchers proposed different methods/techniques to switch the converters for different applications like switched capacitor structure [10–12].

Table 1 Fuel cell comparisons

Fuel cell	Fuel	Electrolyte	Temperature °C	Electrical efficiency	Applications/ Uses
PEMFC	H ₂	Polymer	40–90	40–60	Vehicle, small generators
SOFC	CH ₄ , H ₂ , CO	Solid oxide	600–1000	50–60	Power plants, co-generation
MCFC	CH ₄ , H ₂ , CO	Molten carbonate	600–700	40–50	Power plants, combined heat power
PAFC	H ₂	Phosphoric acid	150–220	36–42	Power plants, combined heat power
DMFC	Methanol	Polymer	60–130	~ 40	Vehicle and small appliances
AFC	H ₂	Potassium hydroxide	40–250	60–70	Space shuttles (outer space)

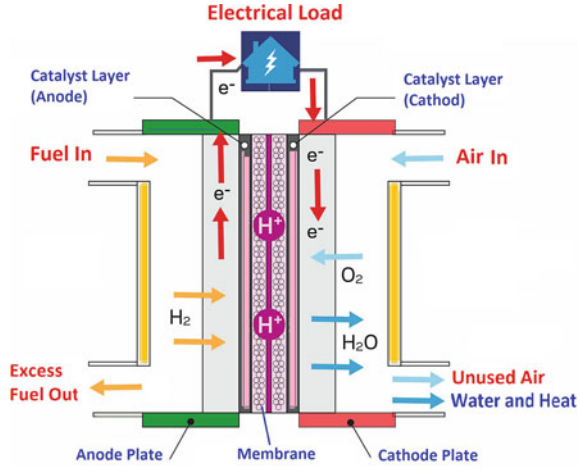
A dual-switch coupled inductor-based high step-up DC-DC converter is one of the best solutions to get high gains [13]. In [14], Allehyani and Ahmed analysed a transformer-less single switch high gain DC-DC converter for renewable energy systems. Ahmad Alzahrani et al. have proposed a non-isolated interleaved DC-DC boost converters with voltage multiplier cells [15], Pasha and Mallika proposed a new hybrid boosting converter (HBC) with a bipolar voltage multiplier (BVM) [16].

To extract the maximum power and improve the efficiency of the system, the maximum power point technique playing a major role. Authors across the globe proposed different methods [17, 18]. In this work, modelled a cuckoo search-based maximum power point algorithm using MATLAB. The complete paper is organized as: modelling of proton exchange fuel cell is presented in Sect. 2, hybrid boosting converter is presented in Sect. 3. Simulation models and cuckoo search MPPT algorithm are presented in Sect. 4. Final conclusion is presented in Sect. 5.

2 Modelling of Proton Exchange Membrane Fuel Cell

Simulation software proving a platform to the engineers to predict the performance of a new model, design and test the build model under different operating conditions either in the form of simulation or a mathematical model represented in the form of the block diagrams, visualizing the outputs and analysing the results with less cost. Fuel cells are actively being researched for use as stationary distributed power units, vehicular power sources due to its portability, quiet in its operation, lower carbon emissions, fast response to load variation, and can be practically implemented for

Fig. 2 Fundamental components of PEMFC



different power capacities. Fuel cell technology: Greener, clean and future technology is an alternative to fossil fuels. Especially, the electrical vehicular systems are expected to change the carbon emission during vehicular operation [19, 20]. The fundamental components of PEMFC are shown in Fig. 2. The chemical reactions that are taking place at the anode and the cathode of a PEMFC are shown below. At anode:



At cathode:



Fig. 3 shown the V-I characteristics of PEMFC. As it is indicated that the voltage is decreasing linearly as the electrical load on the system increases: Due to polarization effect of the fuel cell. In the similar lines, increase in temperature reduces the activation losses of PEMFC. These conditions demand to maintain the temperature and pressure levels of the PEMFC remains constant. The specifications of the modelled PEMFC are shown in Table 2.

3 Hybrid Boosting Converter

The output voltages of renewable energy system with lower voltages need to improve to connect them to the utility system and the grid. Basically, we mostly depend on the boost converters and these are able to step up, step-down, or invert a voltage. The basic switching pattern of the converters has limited voltage conversion ratio. The salient features of the hybrid boosting converter (HBC) [16]: boost structure, regulation and

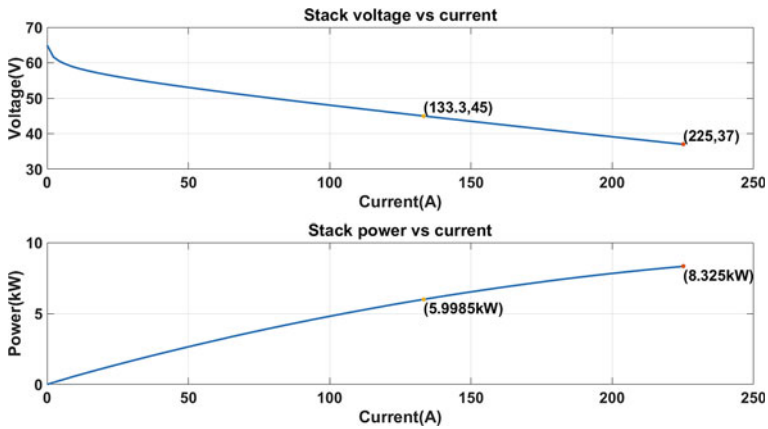


Fig. 3 PEMFC characteristics

Table 2 Details of PEM FUEL CELL

PEMFC rated power	6 kW
Stack power nominal	5998.5 W
Stack power maximal	8325 W
Fuel cell resistance	0.07833 Ω
Nerst voltage of one cell	1.1288 V
Nominal utilization—hydrogen	99.56%
Nominal utilization—oxidant	59.3%
Nominal consumption: fuel	60.38 slpm
Nominal consumption: air	143.7 slpm
Exchange current	0.29197
Exchange coefficient	0.8 s0.60645
System temperature	338 K
Fuel supply pressure	2.81×10^4 kmol/(s atm) 1.5 bar
Air supply pressure	1 bar

gain enhancement due to its arrange of elements with internal bi-polariness to obtain the voltage multiples. The second-order HBC is shown in Fig. 4.

The details of the HBC circuit are shown in Fig. 4. The two versions of HBC: (a) oddorder HBC (b) evenorder HBC as shown in Figs. 5 and 6. Though, both the even and odd order HBCs possesses the similarity in the characteristics and the analysis of the circuit topology and the method used. The higher component utilization is achieved with the even-order topology with the help of input source as part of the output voltage of the circuit. In this work, even-order topology has been considered for the work.

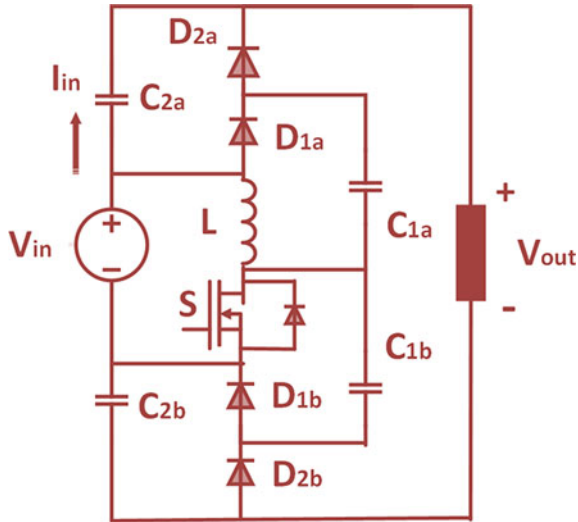


Fig. 4 Second order HBC

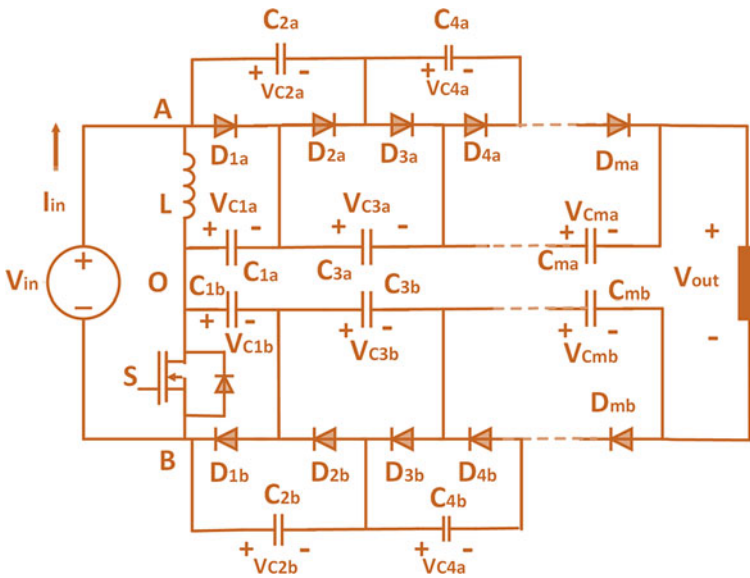


Fig. 5 Odd order HBC

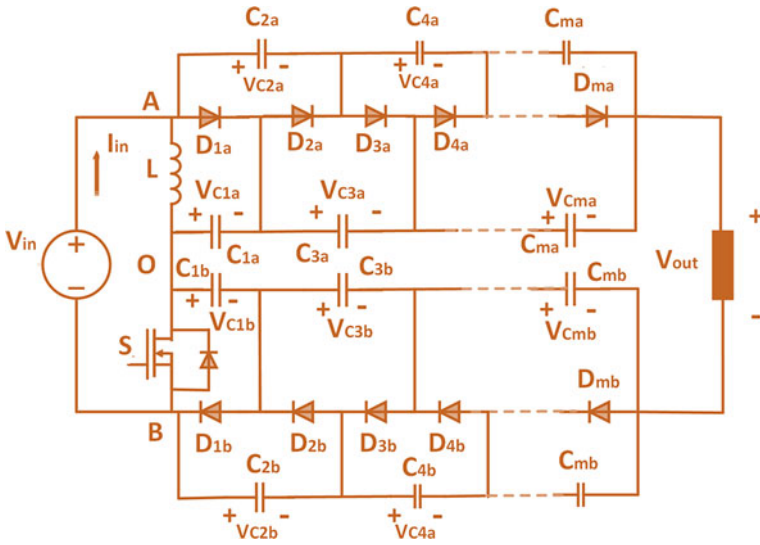


Fig. 6 Even order HBC

3.1 Inductive Switching Core

Inductor has used a major storage device that is used in all the switching patterns/operations of the boost converters with the condition: releasing the energy to load. The energy stored is in the form of magnetic energy. In an HBC topology, the inductor along with the switch is playing an important configuration as shown in Fig. 7. The complimentary electrical outputs of the configuration points: AO and OB of the circuit.

4 Simulation Models

4.1 Two Level Hybrid Boost Converter: Theoretical Model

The two level HBC is modelled, which is shown in Fig. 8. The value of each capacitor used is $200e-6$ F. The inductor used in inductive switching core is of value $1000e-6$ H. The switching frequency is 40 KHz. The input DC voltage is 200 V. The simulation is carried out for 5 s. The obtained voltage is 2572 V. The boosted output voltage is shown in the graph of the Fig. 9.

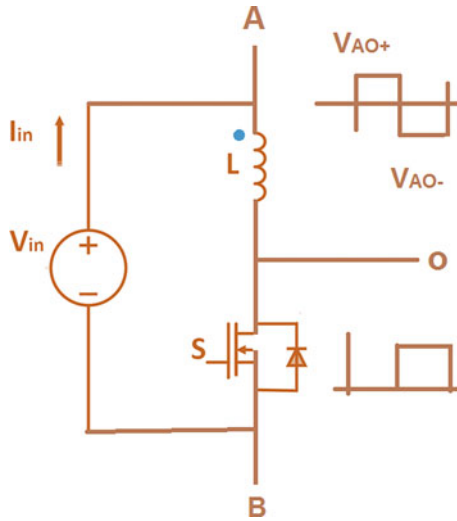


Fig. 7 Inductive switch

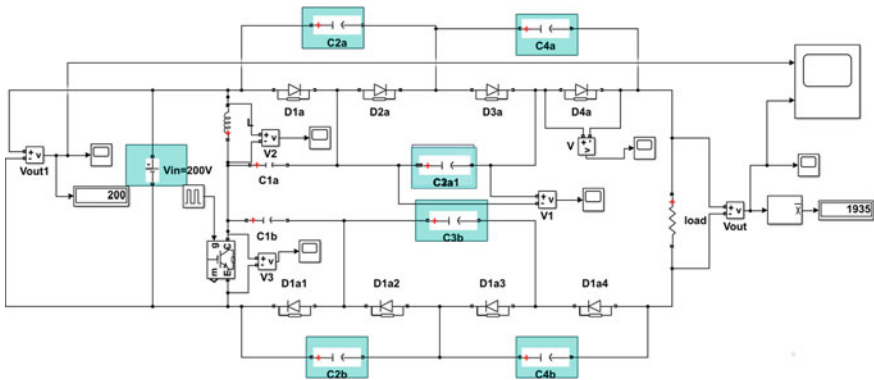


Fig. 8 MATLAB HBC theoretical model

4.2 Equivalent Model of Two Level HBC

The equivalent HBC is modelled which is shown in Fig. 10; value of each capacitor C1a and C1b is $(1000e-6)/3$ F, C2a and C2b are $(1000e-6)/4$ F. The inductor used in inductive switching core is of value $1000e-6$ H. The switching frequency is 40 KHz. The input DC voltage is 200 V. The simulation is carried out for 5 s. The obtained voltage is 1885 V. The output voltage graph obtained from this model is shown in Fig. 11.

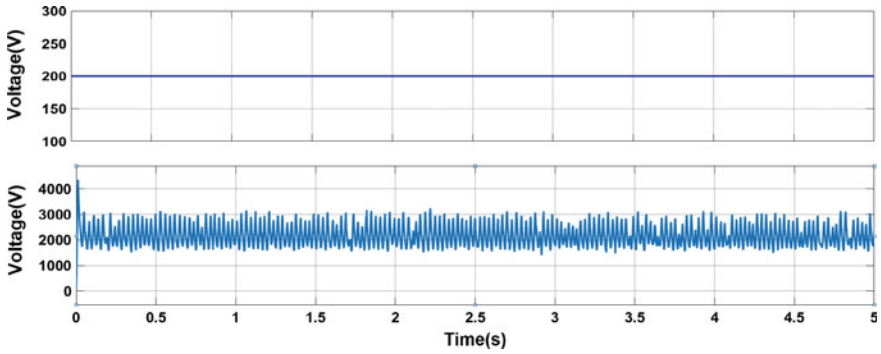


Fig. 9 Input and output voltages of actual HBC model

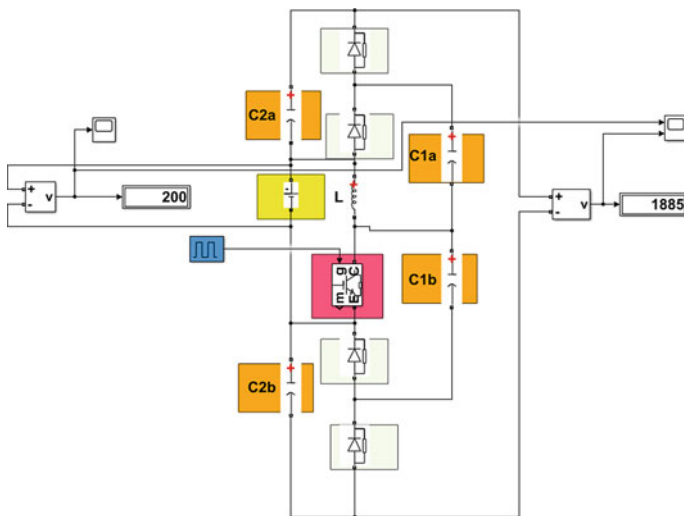


Fig. 10 MATLAB: equivalent model

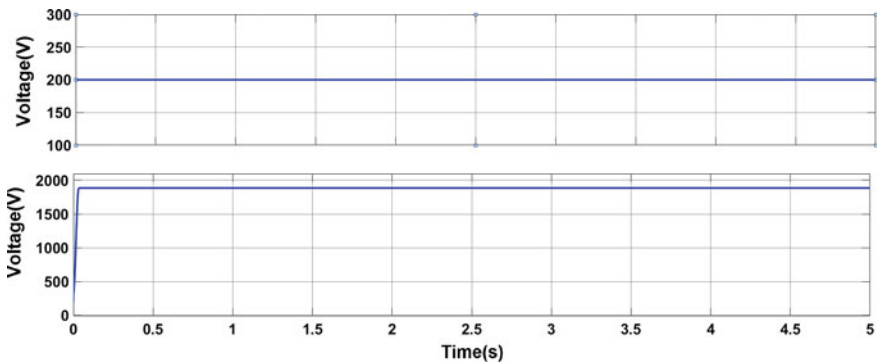


Fig. 11 Equivalent model: input and output voltage

4.3 Cuckoo Search Algorithm to Obtain MPPT

The importance of the maximum power point tracking mechanism and the different conventional method's and soft computing techniques like: variable step size perturb and observe, incremental conductance and modified incremental conductance, fractional open circuit methods, fixed step size methods of radial-based functions, artificial intelligence methods, are proposed and implemented by researchers across the globe [21]. In this work, a cuckoo search algorithm is built to obtain the maximum power point tracking of the solar system. The flow chart to implement the MPPT is shown in Fig. 12.

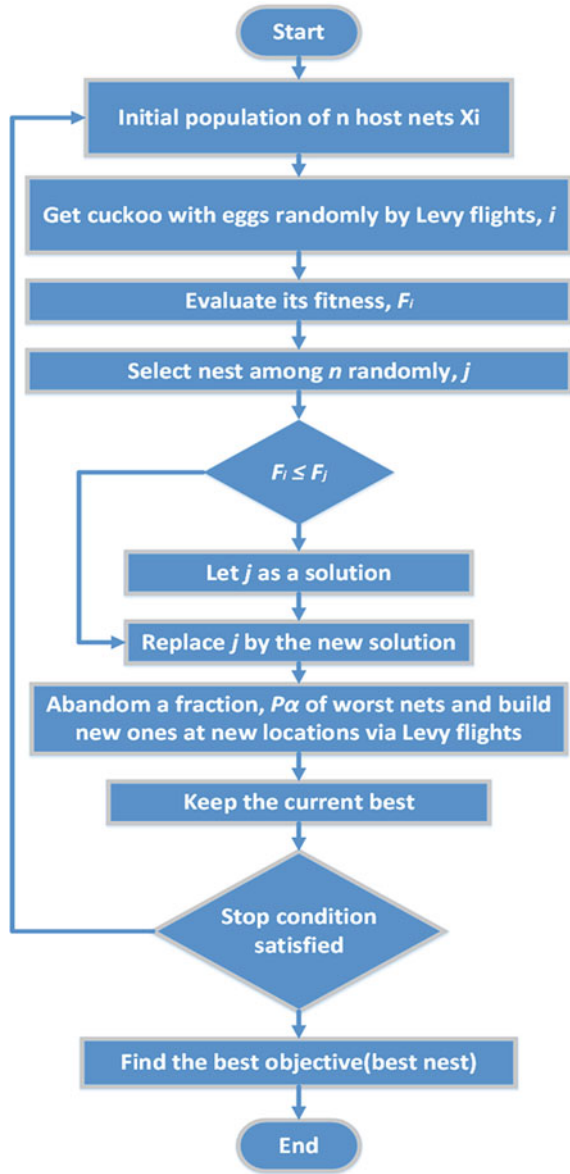
4.4 Simulation of PEMFC HBC Without and with MPPT

PEMFC is connected to normal boost converter which is modelled. The value of each capacitor used is $1000e-6$ F. The inductor used in inductive switching core is of value $1000e-6$ H. The switching frequency is 40 KHz. Figure 13 shows the build MATLAB model of PEMFC with the HBC and cuckoo search MPPT. Fig. 14 shows PEMFC characteristics and DC bus voltage, current. Initially, it carried the simulation without using MPPT. The output DC voltage of the PEMFC is 61.34 V; it is given as an input to boost converter. The simulation is carried out for 2 s. The obtained voltage is 113.2 V. The simulation results of input and output voltages are shown in Fig. 15. Fig. 16 shows the PEMFC with cuckoo search MPPT and HBC with the output voltage of 439.0 V. As it is clearly shown that it increases the output of the HBC converter with cuckoo search-based MPPT.

5 Conclusion

This paper started with the discussion of the importance of the renewable energy systems and the most promising source is building especially for electrical charging stations, fuel cells. Further, it discussed the importance of the boosting converters used in the modern power systems. The simulation work is carried to build proton exchange fuel cell system and the new hybrid boost converter to get the high gain. The simulation studies of actual model, equivalent models are presented. Further, the work is focused to model an HBC with cuckoo search MPPT to extract the maximum power from PEMFC. The simulation results shown the improvement of output voltage of the HBC nearby ten times to its input voltage.

Fig. 12 Flowchart: cuckoo algorithm-based MPPT



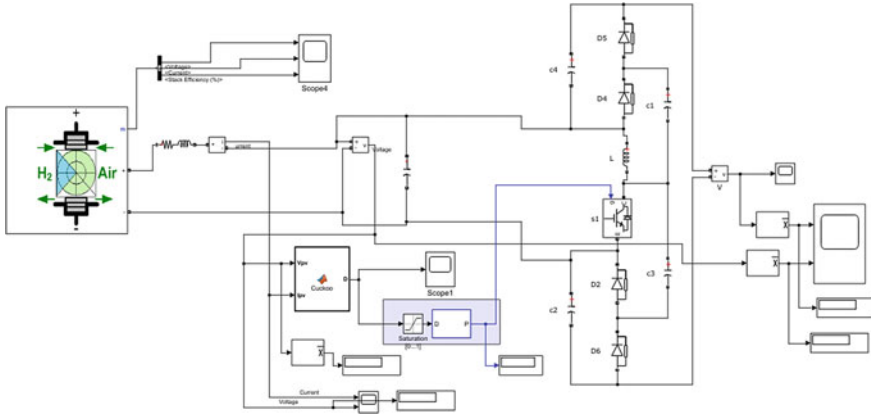


Fig. 13 PEMFC with HBC and cuckoo search MPPT

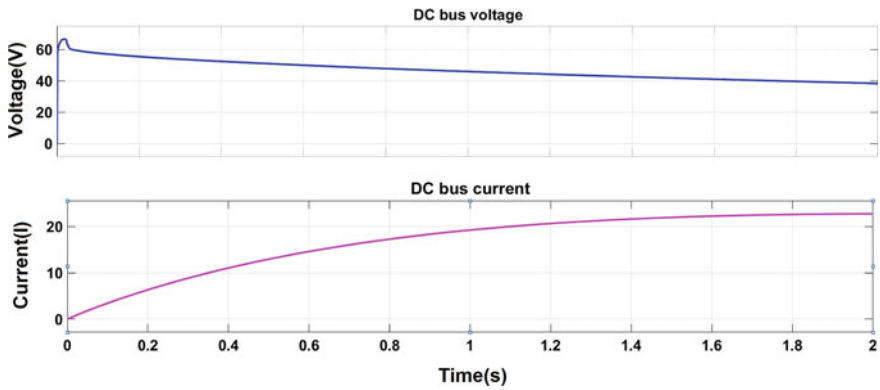


Fig. 14 PEMFC characteristics: DC bus voltage and current

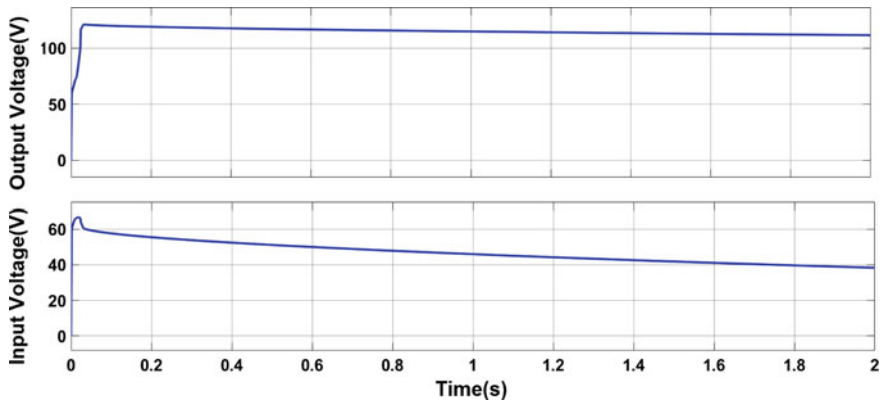


Fig. 15 HBC: PEMFC without MPPT: input and output voltages

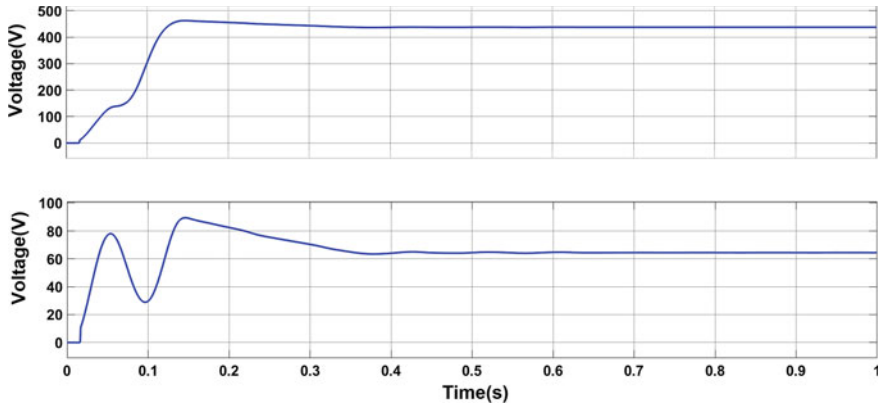


Fig. 16 HBC: PEMFC with cuckoo search MPPT: input and output voltages

References

1. Husin H, Zaki M et al (2021) A critical review of the integration of renewable energy sources with various technologies. *Protect Control Modern Power Syst* 6(1):1–18
2. Brijesh B, Anoop S (2021) Power sector reforms and technology adoption in the Indian electricity distribution sector. *Energy* 215:118797
3. Santosh Ghosh, Vinod Kumar Yadav, Vivekananda Mukherjee, and Shubham Gupta. Three decades of indian power-sector reform: A critical assessment. *Utilities Policy*, 68:101158, 2021
4. Muttaqi KM, Rabiul Islam M, Sutanto D (2019) Future power distribution grids: integration of renewable energy, energy storage, electric vehicles, superconductor, and magnetic bus. *IEEE Trans Appl Supercond* 29(2):1–5
5. Verma MK, Mukherjee V, Yadav VK, Ghosh S (2020) Indian power distribution sector reforms: a critical review. *Energy Policy* 144:111672
6. Energy Policy Review India (2020) India energy outlook 2020 energy policy review India 2020
7. Ahluwalia S (2021) Special report no. 140 national electricity policy, 2021
8. India Energy Outlook (2021) India energy outlook 2021
9. Muthukumar M, Rengarajan N, Velliyangiri B, Omprakas MA, Rohit CB, Kartheek Raja U (2021) The development of fuel cell electric vehicles—a review. *Mat Today Proc* 45:1181–1187
10. Fang Lin Luo and Hong Ye (2004) Positive output multiple-lift push-pull switched-capacitor Luo-converters. *IEEE Trans Ind Electron* 51(3):594–602
11. Sudarsan Reddy DV (2021) Review on power electronic boost converters. *Australian J Electric Electron Eng* 18(3):127–137
12. Starzyk JA, Jan Y-W, Qiu F (2001) A dc-dc charge pump design based on voltage doublers. *IEEE Trans Circuits Syst I Fundamental Theory Appl* 48(3):350–359
13. Rahimi R, Habibi S, Shamsi P, Ferdowsi M (2021) A dual-switch coupled inductor-based high step-up dc-dc converter for photovoltaic-based renewable energy applications. In: 2021 IEEE Texas power and energy conference (TPEC). IEEE, pp 1–6
14. Allehyani A (2021) Analysis of a transformerless single switch high gain dc–dc converter for renewable energy systems. *Arabian J Sci Eng* 1–12
15. Alzahrani A, Ferdowsi M, Shamsi P (2019) A family of scalable non-isolated interleaved dc-dc boost converters with voltage multiplier cells. *IEEE Access* 7:11707–11721
16. Pasha SR, Mallika B (2016) A new hybrid boosting converter for renewable energy applications. *Int J Adv Technol Innov Res* 8(19):2348–2370

17. Nadeem A, Hussain A (2021) A comprehensive review of global maximum power point tracking algorithms for photovoltaic systems. *Energy Syst* 1–42
18. Sarvi M, Azadian A (2021) A comprehensive review and classified comparison of mppt algorithms in pv systems. *Energy Syst* 1–40
19. Ogungbemi E, Wilberforce T, Ijaodola O, Thompson J, Olabi AG (2021) Selection of proton exchange membrane fuel cell for transportation. *Int J Hydrogen Energy* 46(59):30625–30640
20. Spiegel C (2018) *Pem fuel cell modelling and simulation using matlab*. Academic Press
21. Basha CH, Rani C (2020) Different conventional and soft computing mppt techniques for solar pv systems with high step-up boost converters: a comprehensive analysis. *Energies* 13(2):371

Survey on Smart Personalized Healthcare System in Fog-Assisted Cloud Environments



T. Veni

Abstract The proliferation of the Internet of things has enabled the penetration of many smart devices in order to enhance the life quality of people. As a growing healthcare trend, many enterprises have released their smart wearable personalized healthcare devices to monitor the health status of individuals anywhere at any time. These health devices exploit various IoT sensors to collect the user's health parameters which has to be analyzed quickly to meet stringent requirements of latency sensitive healthcare applications. The IoT devices are not sufficient for performing such large-scale and compute-intensive analytics due to its resource constraints. Current cloud-based solutions play a significant role in execution of IoT applications, but it has limitations in terms of geographical centralized architecture, multi-hop distance from the data source which adversely affects the latency sensitivity of the IoT services. To combat this issue, the fog computing has emerged as a promising paradigm that provides cloud-like elastic services to the close proximity of end devices. This paper provides detail survey on the concerns and challenges associated with the development of smart personalized fog-assisted healthcare system and highlights the promising future research directions.

Keywords Internet of things · Healthcare system · Fog computing · Cloud computing

1 Introduction

The proliferation of the Internet of things (IoT) has led to the penetration of many smart devices into people's daily life to enhance their life quality. As a growing healthcare trend, many enterprises such as Apple, Google, Microsoft, Nike, and Samsung have released their smart wearable personalized healthcare devices to monitor the health status of individuals anywhere at any time [1]. These health devices exploit

T. Veni (✉)
National Institute of Technology, Calicut, Kozhikode, India
e-mail: veni@nitc.ac.in

various IoT sensors to collect the user's health parameters (blood pressure, body temperature, etc.). Typically, the IoT sensors generate massive volume of data which has to be analyzed quickly to meet stringent requirements of latency sensitive healthcare applications. The IoT devices are not having sufficient capability to process such compute-intensive analytics. Currently, cloud computing paradigm is widely used to cater the need of IoT applications. Nevertheless, cloud entails limitations in terms of geographical centralized architecture, lack of mobility support, multi-hop distance from the data source which adversely increases the latency of time-critical applications [2]. To combat this issue, an extension of the cloud computing named fog computing emerged as a promising paradigm which provides cloud-like services at the edge of the network and thereby reduces the frequent intervention of cloud servers at each time. Thus, IoT environments can use fog-assisted cloud environments in order to execute time-critical healthcare applications [3, 4].

Health care is one of the important application areas that necessitates precise and quick results as life-critical decisions are made using the data received from IoT devices. While there exists a wide plethora of fog-based healthcare management approaches, most of them developed for providing healthcare services for indoor users [5–7]. Typically, in case of outdoor region, user may suffer from sudden health problem such as heart attack while traveling in a vehicle. Moreover, there may be a connection interruption during such travel ling which adversely affects the service delivery. In such circumstances, the mobility data should be analyzed together with other health parameters to provide prompt healthcare service. There are few existing approaches focused on providing mobility-aware healthcare services for outdoor users [8–19]. However, these approaches did not consider the real-time information about post-disaster affected regions or any events which is necessary to quickly predict the shortest ambulance's route path to nearby health centers with minimal congestion in case of emergency situations. Besides, the fog servers are vulnerable to various kinds of attacks. Offloading computation task to a malicious fog node affects the integrity of user's health data. Although there are various secure mechanisms in fog environments, most of them depend on an authentic central entity which having a problem of single point of failure.

The smart personalized healthcare system can be developed in order to provide intelligent healthcare recommendations to users anywhere at anytime. This can be achieved by employing efficient machine learning algorithm to analyze the heterogeneous data sources such as healthcare, contextual information, and real-time information. Hence, this survey details the various mechanisms and issues related to the development of personalized healthcare systems. The major contributions of this survey are summarized as follows:

- The topology of IoT-fog-cloud smart healthcare prospective model is presented.
- The novel taxonomy based on existing approaches and algorithms to provision smart and efficient fog-assisted healthcare services is presented.
- The open research challenges and future research directions are highlighted.

To the best of our knowledge, this comprehensive survey will be beneficial for researchers and act as a base stone for smart healthcare fog computing environments.

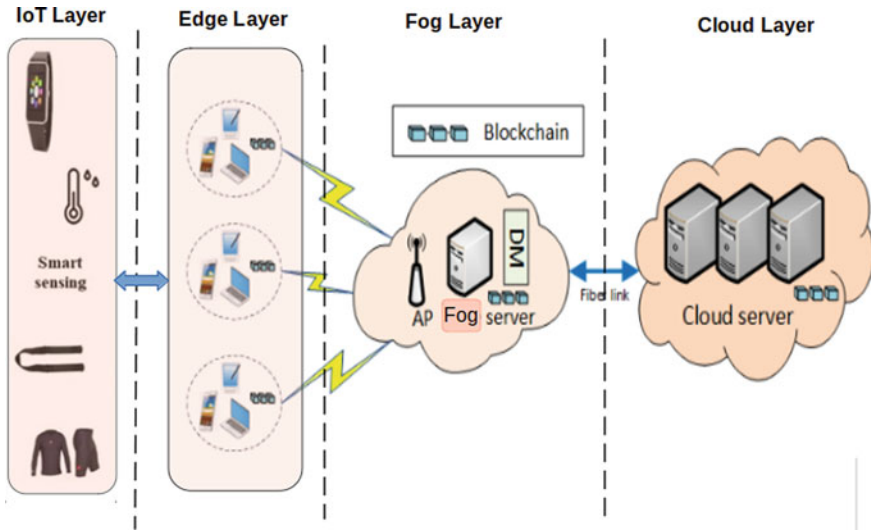


Fig. 1 IoT-cloud-fog smart healthcare prospective model [20]

2 IoT-Cloud-Fog Smart Healthcare Prospective Model

Health care is one of the important application areas that necessitates precise and quick results as life-critical decisions are made by means of the data received from IoT devices. The main objective of the prospective smart health model is to provide efficient healthcare services in a timely manner. The healthcare prospective model is depicted in Fig. 1 [20].

The details of each layer are described as follows:

- **IoT Layer:** In this bottom layer, the diverse IoT sensors such as temperature, blood pressure, proximity sensors etc., are used for sensing the application specific data and transferring the raw sensed data to the attached edge device.
- **Edge Layer:** In this layer, the user mobile devices act as an edge device, which are responsible for processing the sensed data of IoT devices. The edge devices are resource constraint devices, which offload the task to the fog devices when processing is beyond its computational capability. Each mobile device in this layer has blockchain account in order to join into the network and perform offloading task to the fog layer.
- **Fog Layer:** This fog layer constitutes of geo-distributed fog devices, like routers, gateways and micro-data centers etc., which are intelligent enough to process the task received form the edge devices. Typically, the fog devices are capable of small scale processing. If processing is beyond its computational capability, it forwards the task to the cloud layer.
- **Cloud Layer:** This layer comprises of multiple high-end servers which are capable of processing and storing a massive volume of data.

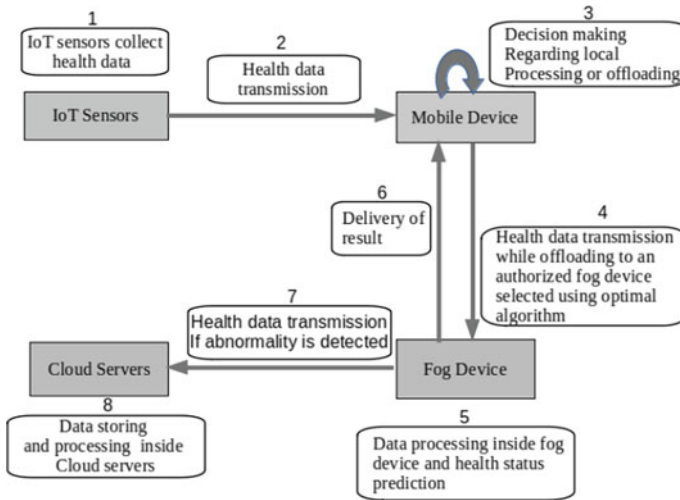


Fig. 2 Workflow of smart healthcare model

The working model of smart healthcare architecture is depicted in Fig. 2. It comprises of the following steps:

1. The IoT sensor nodes sense the various health parameters such as body temperature, blood pressure, etc., and forward it to the attached user’s mobile device.
2. Mobile device initializes an offloading request as a blockchain which is verified and validated by miners. Then, mobile device offloads the sensor data to the DM (authorized fog device under the service coverage area) along with the context information and user’s geolocation information.
3. Decision maker in the fog layer will perform optimization in order to decide which fog device to execute the offloading request. The chosen fog devices process the information with respect to the model which pre-defined by medical experts and send the resultant health status to the user’s mobile device. Moreover, the fog device sends the results to the cloud in addition with the context, geolocation, and real-time information.
4. Cloud can recommend the path to the nearby healthcare center with minimum congestion to the mobile device. In addition, the healthcare centers can access the data from cloud servers for the treatment of patients.

3 Review of Smart Personalized Healthcare Approaches

Smart healthcare management is an extensively researched topic in fog computing environments. The existing literature in fog healthcare management approaches has been broadly classified into fog offloading mechanism for healthcare services and machine learning-based health data analysis method.

3.1 Fog Computing Offloading Mechanism

Abdelmoneem et al. [10] proposed mobility-aware scheduling approach, named MobMBAR that balances the healthcare task distribution dynamically in accordance with the mobility information of patients for minimizing the total execution time. Wang et al. [11] proposed mobility-aware computation offloading method based on the user mobility in terms of mobile device contact patterns. The optimal opportunistic offloading problem was formulated by exploiting the statistic property of contact rates. Guo et al. [12] proposed a game-theoretic greedy approximation offloading scheme to minimize the task computational overhead while satisfying QoS constraints in the fog computing environments. Wang et al. [13] proposed a mobility-aware task offloading and migration scheme in fog environments. A gini coefficient-based fog computing nodes (GCFSA) are proposed to optimize the offloading decisions and the resource allocation problem was solved using genetic algorithm. Javanmardi et al. [14] proposed a FPFTS framework, a task scheduler which exploits the advantage of fuzzy theory and particle swarm optimization with consideration of user's mobility, task latency.

Besides, the fog servers are vulnerable to various kinds of attacks. Offloading computation task to a malicious fog node affects the integrity of users' health data. Although there are various integrity preserving techniques in fog environments, most of them depend on an authentic central entity which is having single point of failure [21, 22].

Luong et al. [23] proposed a decentralized secure fog offloading strategy for mobile blockchain networks. Xiong et al. [24] proposed a blockchain-based fog resource management approach using two-stage Stackelberg game theory approach. Xu et al. [4] proposed a blockchain-based decentralized secure computation offloading strategy in mobile-edge computing environments. The genetic algorithm, multi-criteria decision making (MCDM) was used to choose the most optimal offloading strategy. Nevertheless, most of the afore-mentioned techniques did not consider the user mobility features for the offloading decisions.

Tuli et al. [15] proposed a 'HealthFog' framework for automatic diagnosis of heart diseases using ensemble deep learning method. Ghosh et al. [16] proposed a Mobi-IoST framework which predicts the user mobility in real time in order to deliver the healthcare service to the user with minimal latency and power consumption. Mukherjee et al. [17] proposed a FogIoHT framework, for indoor and outdoor users in order to select the best fog device for health data processing using a weighted game theory approach. Mukherjee et al. [18] proposed a healthcare framework (IoHT) which predicts the mobility pattern of the user for assisting the user to the nearby health center. Sood et al. [19] proposed IoT-fog-cloud framework to prevent dengue fever infection by employing the support vector machine (SVM) and temporal network analysis mechanisms.

3.2 Machine Learning-Based Health Data Analysis Method

Gia et al. [5] proposed a low cost health monitoring (LCHM) IoT-based fog-assisted system to process the health data of various heart patients efficiently. Moreover, the sensor nodes can be used to acquire the data such as ECG, respiration, environment temperature, humidity, and transmit to a smart gateway in order to make automatic analysis and notification. Keke et al. [6] proposed a healthcare system which uses an intelligent agent (IA) to provide an optimal treatment solution with minimal cost.

Muhammed et al. [7] proposed a ubiquitous healthcare (UbiHealth) framework to provide personalized healthcare services at reduced risks and cost. The cloudlet and network layer were used to process the health data. UbiHealth achieved an enhanced network quality of service (QoS) using deep learning-based prediction and classification approach. Nonetheless, the user's context information such as location and corresponding atmospheric condition is having great influence on health status prediction. For instance, the result of the health status will differ from the user who has done swimming for half an hour to the same user who is in relax mode. It can be observed that all the afore-mentioned proposed techniques did not consider the user's context information which results in low accuracy in health status prediction.

Solanas et al. [8] proposed context-aware healthcare system for the people living in smart- cities. Hassan et al. [9] proposed a HAAL-NBFA framework in order to perform remote monitoring of the elderly patients and predict the health status accurately in real time by means of context-awareness techniques. While there exists a wide plethora of fog-based healthcare management approaches, most of them developed for providing healthcare services for indoor users. Typically, in case of outdoor region, user may suffer from sudden health problem such as heart attack while traveling in a vehicle. Moreover, there may be a connection interruption during such traveling which obviously affects the service delivery. In such circumstances, the mobility data should be incorporated and analyzed together with other health parameters to assist the user to the nearby healthcare center according to user's health status and their current location.

3.3 Analysis

Health care is one of the important application areas which needs precise and quick results. In order to achieve accurate and quicker results, in state-of-the-art healthcare approaches, either optimal offloading strategy was proposed in order to offload the computational task to the best fog node or intelligent machine learning algorithm was employed on fog nodes for the accurate analysis of healthcare data. However, both fog offloading mechanism and intelligent health processing mechanisms should be jointly considered to achieve the QoS constraints with minimal energy consumption. Besides, the fog servers are vulnerable to various kinds of attacks. Offloading computation task to a malicious fog node affects the integrity and privacy of users'

Table 1 Summary of the existing literature

Healthcare approaches	Mobility	Real time information	Security	Delay	Energy	Accuracy	Fog offloading mechanism	Efficient health data processing mechanism
LCHM [5]				✓		✓		✓
STPH [6]				✓		✓		✓
UbiHealth [7]				✓	✓	✓		✓
Solanas et al. [8]	✓			✓				
HAAL-NBFA [9]				✓	✓	✓		✓
MobMBAR [10]	✓			✓			✓	
Wang et al. [11]	✓			✓			✓	
Guo et al. [12]	✓			✓			✓	
Wang et al. [13]	✓			✓	✓		✓	
FPFTS [14]	✓			✓			✓	
Luong et al. [23]	✓			✓	✓		✓	
Xiong et al. [24]	✓			✓			✓	
Xu et al. [4]			✓	✓	✓			
HealthFog [15]				✓	✓	✓		✓
Mobi-IoST [16]	✓			✓	✓	✓		✓
FogIoHT [17]	✓			✓	✓		✓	
IOHT [18]	✓			✓	✓	✓		✓
Sood et al. [19]				✓	✓	✓		✓

health data. While there exists a wide plethora of decentralized secure mechanisms in fog environments, with the best of my knowledge, no other research work in the existing literature did not consider the user mobility features for the secure offloading decisions. The mobility feature is an important part of the fog computing environments. Moreover, the existing approaches did not consider the real-time information about affected regions or any events for optimal ambulance's route path prediction. The real-time information about affected regions or any events (accidents) is necessary to quickly predict the optimal ambulance's or vehicles route path to nearby health centers by avoiding the affected regions. The summary of existing literature is described in Table 1.

4 Conclusion

Efficient personalized healthcare service management with the consideration of QoS constraints and security is an important but challenging issue in fog environments. In this survey, the two important categories such as fog service offloading and machine learning-based healthcare analysis methods are discussed. A perspective model for provisioning secure, reliable, and quick response to IoT applications is presented. The state-of-the-art research works is presented; important issues and challenges are identified. Although there are enormous amount of research work in this domain, the few opportunities which can be explored in future are highlighted. With the best of our knowledge, this survey will be beneficial for researchers and act as a base stone for smart personalized healthcare service management in fog-assisted cloud computing environments.

References

1. Gu L, Zeng D, Guo S, Barnawi A, Xiang Y (2015) Cost efficient resource management in fog computing supported medical cyber-physical system. *IEEE Trans Emerg Top Comput* 5(1):108–119
2. Gill SS, Arya RC, Wander GS, Buyya R (2018) Fog-based smart healthcare as a big data and cloud service for heart patients using IoT. In: *International conference on intelligent data communication technologies and internet of things*. Springer, pp 1376–1383
3. Gill SS, Garraghan P, Buyya R (2019) ROUTER: Fog enabled cloud based intelligent resource management approach for smart home IoT devices. *J Syst Softw* 154:125–138
4. Xu X, Chen Y, Zhang X, Liu Q, Liu X, Qi L (2021) A blockchain-based computation offloading method for edge computing in 5G networks. *Softw Pract Exp* 51(10):2015–2032
5. Gia TN, Jiang M, Sarker VK, Rahmani AM, Westerlund T, Liljeberg P, Tenhunen H (2017) Low-cost fog-assisted health-care IoT system with energy-efficient sensor nodes. In: *2017 13th international wireless communications and mobile computing conference (IWCMC)*. IEEE, pp 1765–1770
6. Gai K, Lu Z, Qiu M, Zhu L (2019) Toward smart treatment management for personalized healthcare. *IEEE Netw* 33(6):30–36

7. Muhammed T, Mehmood R, Albeshri A, Katib I (2018) UbeHealth: a personalized ubiquitous cloud and edge-enabled networked healthcare system for smart cities. *IEEE Access* 6:32258–32285
8. Solanas A, Patsakis C, Conti M, Vlachos IS, Ramos V, Falcone F, Postolache O, Pérez-Martínez PA, Di Pietro R, Perrea DN, Martínez-Balleste A (2014) Smart health: a context-aware health paradigm within smart cities. *IEEE Commun Mag* 52(8):74–81
9. Hassan MK, El Desouky AI, Badawy MM, Sarhan AM, Elhoseny M, Gunasekaran M (2019) EoT-driven hybrid ambient assisted living framework with naïve Bayes-firefly algorithm. *Neural Comput Appl* 31(5):1275–1300
10. Abdelmoneem RM, Benslimane A, Shaaban E (2020) Mobility-aware task scheduling in cloud-Fog IoT-based healthcare architectures. *Comput Netw* 179:107348
11. Wang C, Li Y, Jin D (2014) Mobility-assisted opportunistic computation offloading. *IEEE Commun Lett* 18(10):1779–1782
12. Guo H, Liu J, Zhang J, Sun W, Kato N (2018) Mobile-edge computation offloading for ultra-dense IoT networks. *IEEE Internet Things J* 5(6):4977–4988
13. Wang D, Liu Z, Wang X, Lan Y (2019) Mobility-aware task offloading and migration schemes in fog computing networks. *IEEE Access* 7:43356–43368
14. Javanmardi S, Shojafar M, Persico V, Pescapè A (2020) FPPTS: a joint fuzzy particle swarm optimization mobility-aware approach to fog task scheduling algorithm for Internet of Things devices. In: *Software: practice and experience*
15. Tuli S, Basumatary N, Gill SS, Kahani M, Arya RC, Wander GS, Buyya R (2020) HealthFog: an ensemble deep learning based smart healthcare system for automatic diagnosis of heart diseases in integrated IoT and fog computing environments. *Fut Gener Comput Syst* 104:187–200
16. Ghosh S, Mukherjee A, Ghosh SK, Buyya R (2019) Mobi-iiost: mobility-aware cloud-fog-edge-iiot collaborative framework for time-critical applications. *IEEE Trans Network Sci Eng* 7(4):2271–2285
17. Mukherjee A, De D, Ghosh SK (2020) FogIoHT: a weighted majority game theory based energy-efficient delay-sensitive fog network for internet of health things. *Internet of Things* 11:100181
18. Mukherjee A, Ghosh S, Behere A, Ghosh SK, Buyya R (2021) Internet of health things (IoHT) for personalized health care using integrated edge-fog-cloud network. *J Ambient Intell Humanized Comput* 12:943–959
19. Sood SK, Kaur A, Sood V (2021) Energy efficient IoT-Fog based architectural paradigm for prevention of dengue fever infection. *J Parallel Distrib Comput* 150:46–59
20. Nguyen DC, Pathirana PN, Ding M, Seneviratne A (2019) Secure computation offloading in blockchain based IoT networks with deep reinforcement learning. *arXiv preprint [arXiv:1908.07466](https://arxiv.org/abs/1908.07466)*
21. Al-Khafajiy M, Baker T, Asim M, Guo Z, Ranjan R, Longo A, Puthal D, Taylor M (2020) COMMITMENT: a fog computing trust management approach. *J Parallel Distrib Comput* 137:1–6
22. Alli AA, Alam MM (2019) SecOFF-FCIoT: Machine learning based secure offloading in Fog-Cloud of things for smart city applications. *Internet of Things* 7:100070
23. Luong NC, Xiong Z, Wang P, Niyato D (2018) Optimal auction for edge computing resource management in mobile blockchain networks: A deep learning approach. In: *2018 IEEE international conference on communications (ICC)*. IEEE, pp 1–6
24. Xiong Z, Feng S, Wang W, Niyato D, Wang P, Han Z (2018) Cloud/fog computing resource management and pricing for blockchain networks. *IEEE Internet Things J* 6(3):4585–4600

News Bias Detection Using Transformers



Varun Magotra, Ebrahim Hirani, Vedant Mehta, and Surekha Dholay

Abstract The main motive of news is to inform people without any highlighting personal opinions. A lot of news that spreads on social media is first introduced by news articles from unreliable sources. A lot of the news that is spread by these sources is only to generate clicks from the masses and has no intention to educate the common people. A news bias detection system aims to find a robust solution to bifurcate between biased news and genuine news. The goal is to make a single reliable platform that accumulates news from multiple sources and gives them a bias rating. We use the classification tokens generated by the BERT architecture to classify the bias of news articles. We then deploy this model through an API and call it through a web-app. The user interface will make browsing news convenient.

Keywords BERT · Bias rating · NLP · Flask · Transformers

1 Introduction

Social media has hugely changed the way people interact today. It has revolutionised the communication process where anyone can be the source of information. Stories, feeds, posts can spread with unprecedented speed, giving individuals access to almost real-time information. However, this is misused by some individuals or groups to spread fake news or any other misinformation for achieving political and economical benefits. Biased news can persuade and direct consumers to believe false ideas that are shared for specific agendas. The viral spread of such biased news is a cause of great concern for all members of society including the government, organisations, businesses and also common man. Bias news detector aims to provide users a single app where they will get all the latest news along with their bias rating. Bias rating is a metric to weigh the neutrality of an article. This will not only help people make a

V. Magotra (✉) · E. Hirani · V. Mehta · S. Dholay
Bharatiya Vidya Bhavan's Sardar Patel Institute of Technology (SPIT), Mumbai 400058, India
e-mail: varun.magotra@spit.ac.in

S. Dholay
e-mail: surekha_dholay@spit.ac.in

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
H. Sharma et al. (eds.), *Communication and Intelligent Systems*, Lecture Notes
in Networks and Systems 461, https://doi.org/10.1007/978-981-19-2130-8_26

319

rational and informed decision but also prevent spreading of rumors. Also because of its modular architecture it can accept any news from user and can assign its bias rating which will help users to validate various text messages coming from various sources like WhatsApp forwards. Our approach is to use recent attention-based techniques to tackle this age-old problem of news bias. Our solution provides an application that fetches news from different sources and assigns them a bias rating. In this study, we propose a method which uses state of the art deep learning model to supplant expensive equipment and expert personnel which are being used to detect the biasness of the news articles. The rest of the paper is structured as follows: In Sect. 2, we briefly discuss related work in bias detection and making bias aware news recommendation system. In Sect. 3, we discuss about the dataset which has been used in this study to train the model. Section 4 outlines of our method, along with some implementation specific details. In, Sect. 5, research gaps and limitation of the current research in this domain has been discussed, Sect. 6 Result and Sect. 7 Conclusion.

2 Related Work

There have been various attempts to detect bias in news articles as well as different forms of media. The initial methods relied on identifying key words, and using their occurrences to detect bias. Deep learning and language models revolutionised Natural Language Processing, and the more recent attempts to solve this problem have made extensive use of these techniques.

In 2020, Aggarwal et al. in their study [1] investigated the tweets by various media outlets and assessed the polarity in the tweets of these media outlets. Aggarwal et al., used the VADER method to give the polarity score to the tweets depending upon the content of these tweets.

In 2018, Patankar et al. in their study [2] have used a different approach for detecting the bias. Patankar et al. use real-time bias classification to inform the users of bias in the text. Furthermore, they find articles relevant to the topic from different sources and these articles, and their bias scores are suggested to the user involved in that topic.

Vu in his paper attempted to identify the political bias in the news from various news publications [3]. In his study, author tried a different technical approach to this problem and used multilayer perceptrons to do the work. He used fast-text word vectors for initialising the embedding matrix. These embeddings are then fed to the MLP model for classification. The model is used in a browser tracking system as an extension for a browser. This extension classifies the components of the articles as conservative, liberal and unbiased on the basis of the content of the article.

Reddy et al. proposed an attention mechanism applied on the headline of news articles that encoded the articles through bidirectional LSTMs and used attention mechanism on the headline. Their proposed model outperformed all the baselines. They trained the model on Telugu articles and assigned biases according to which political party that article leaned towards [4].

Chen et al. in their study addressed the shortcoming of the neural networks for bias detection. In their study, the authors used the probability distribution method on the context of the article based on the work frequency, position, etc. which further used in the Gaussian mixture model. Using this approach, the authors are able to outperform the existing systems for article-level bias [5].

3 Dataset Description

The dataset used in this study is the combination of two news dataset which is “All the news dataset” and “Indian News Articles”. All the news dataset contain 143,000 articles from 15 American publications while the Indian News Articles dataset contain 15,382 news article from various Indian news Publication Major part of our data consists of articles from “All the news dataset” due to the sheer abundance of them, while we have also used all the articles from “Indian news articles” which are very less compared to “All the news dataset”, but we have included nonetheless due to the fact that our model will be applied on Indian news articles. Once the data is created from these two datasets, we annotated the appropriate labels to the News Publications whether a particular News Publication is left biased, right biased or neutral. The bias of the news publisher is determined using MediaBiasFactCheck [6].

4 Methodology

In our study, we have taken a different approach than the existing ones. Our approach for detecting the news bias from news articles is modularised and explained as.

4.1 An Application that Collects News from Different Sources and Assigns Bias Ratings

Users don't need to search for different articles online as the application acts as a hub for news articles. News-API [7] is used in our application to fetch recent news and it is displayed by our application. These articles are then fed to model, and their bias is decided. The articles are segregated in different genres (politics, sports, financial, etc.) to make the searching for articles efficient and as per the liking of the user. Generally trending articles are also displayed on the home-screen. The title of articles has different colour schemes to indicate bias. Title with red font indicates that the article is biased, while green indicates that it is neutral. We refrain from showing the inclination of bias (i.e. right bias or left bias) as our main aim is to alert the users of bias.

4.2 *New Technologies*

A model is created that concatenates the text of the article heading and the article content. The model calculates the encodings using the BERT (bidirectional transformer) architecture [8] as shown in Fig. 1. A dense layer then calculates the classification between neutral, left bias and right bias. The text is pre-processed before feeding it to the model. The use of transformers incorporates attention mechanism to calculate bias in text. Attention maps recognise relationships between tokens and also consider their positions, that allow the model to recognise the significance of permutations in the arrangement of tokens. The CLS (classification) token that is outputted by the model is used as input for the dense layer. This layer uses softmax activation to determine the bias class of the article. Use of transformers allow faster training through parallel computation within the same layer unlike RNN (Recurrent Neural Network)-based networks that require output from the node for the previous token for computation. Though the bias of the article is determined by the publisher in our dataset, the model isn't given any information about the publisher and is trained solely based on the content of the article. This makes the model predictions independent of the article source.

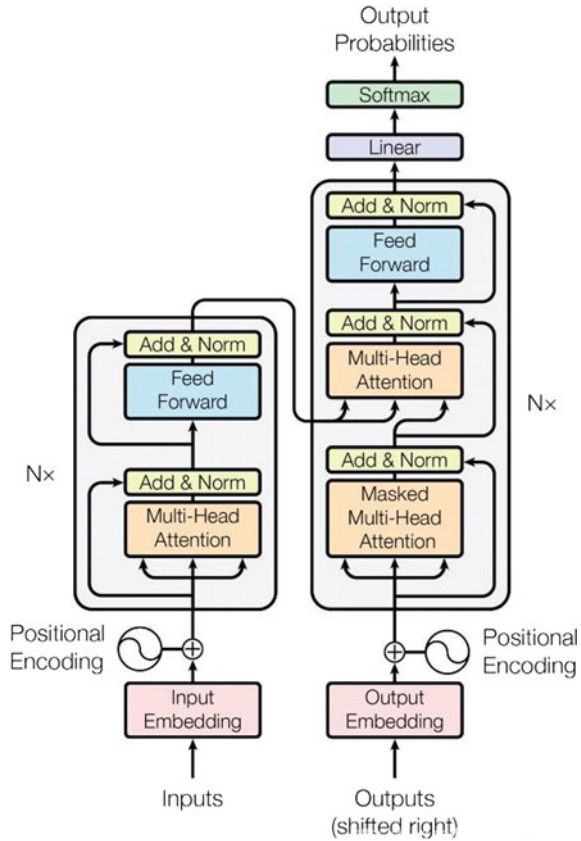
4.3 *An API that Takes in News Articles or Any Textual Data and Returns Bias Ratings for That*

An API is provided to the users where they can post the text of any news article and a bias rating is returned. This allows the application to be applicable to different platforms such as social media or whatever is appropriate to the user. It also allows the user to check existing articles that aren't visible in the application (The app only displays articles fetched from News-API) included in the application. The API is created using Flask-RESTful framework that allows us to host the model and the back-end processes of calling News-API and applying the model to articles as well as the login and registration of users. This allows us to use react in the frontend and make simple calls to the backend without having to do any processing on the end devices. The flow of data from API to user end is shown in Fig. 2.

5 *Research Gaps and Limitation*

There are very few applications that provide the information about the biasing of a particular news. And there aren't any applications which aggregate news from different news sources and assign bias ratings to different articles on a single platform. The work that has been done in this regard mostly uses old state-of-the-art technologies and architecture, and there is very limited research done on this topic as there are not

Fig. 1 Bert architecture [8]



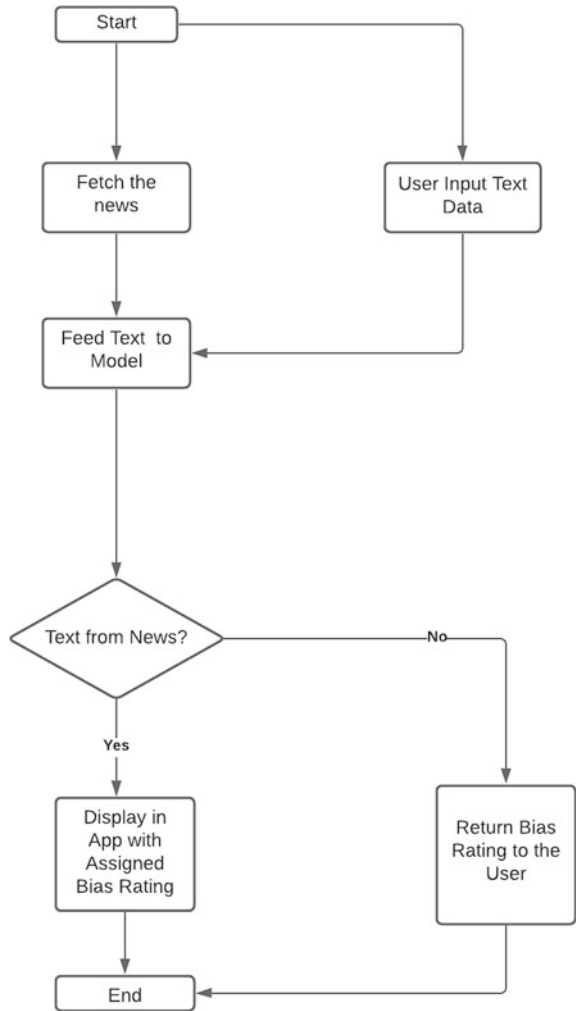
a lot of datasets. The problem with training a model for bias is that there is a chance that the model itself picks up the biases of the individual that has assigned them in the datasets. So in our paper, we have assigned the biases from a source that allows the input of multiple people in order to classify a news source as biased. This prevents individual biases to creep into our model. We also take advantage of state-of-the-art technologies and are able to achieve better results.

6 Results

The BERT model which is used to provide the bias rating to the articles gave a training accuracy of 99.10% and validation accuracy of 95.54% when trained for 5 epochs. A web app is developed that uses this model to assign bias ratings to the news articles fetched by News-API which can be seen in Fig. 3.

As we can see, the model is giving the bias rating according to the context of the news provided by the News-API. For our research purpose, we used the Un-Paid

Fig. 2 Application flow



version of the API, so the requests to the API are limited to the 100 per day with no uptime SLA. Furthermore, we purposely ran the inference on a CPU device(Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz 1.80GHz). Still, the average response time was 4–5 (We were calculating biases for all the articles fetched from the News-API in one request).

As we can see from Fig. 3, the red highlight shows that the news article is biased and the author of the article presents their own views, whereas the article which is highlighted as green is neutral and the author only presents facts. In addition to this, our app also segregates news articles by their genre, allowing users to select news articles according to their interests as shown in Fig. 4.

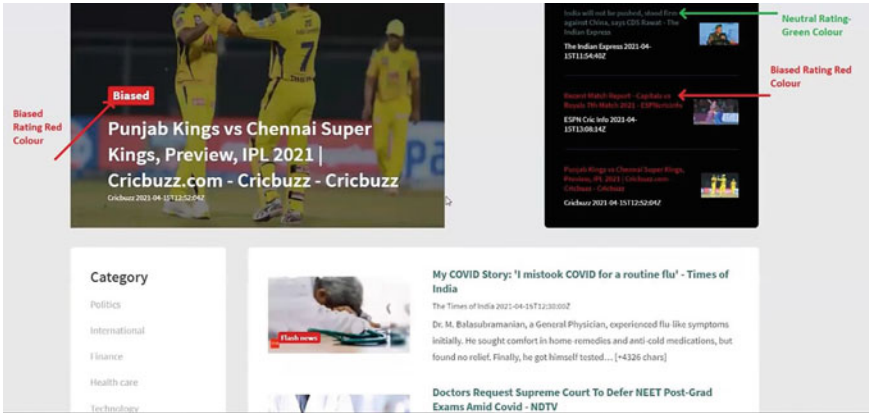


Fig. 3 News bias assignment in application

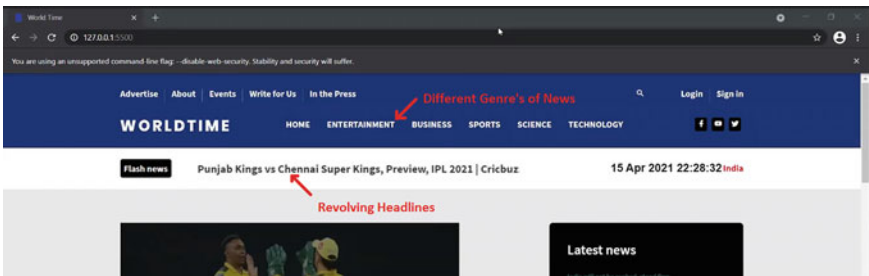


Fig. 4 Different news genre

Furthermore, the application hosts an API that allows clients to request bias ratings by posting news articles or any other text. So that the users can also check the bias of the news text they found on some other networking sites.

7 Conclusion

Up until now, RNN has played a major role for all the tasks related Natural Language Processing whether its text generation, text conversion, etc. But the introduction of transformers has changed the whole overview of all these tasks. The use of transformers in NLP domain has been a welcome change. These attention networks are powerful and require less time to train. Our solution offers an API for clients to use which expands the applicability of our application beyond news articles. Bias recorded from different articles from separate genres and sources can be used to perform data analysis and research, which can further help us determine trends in news media regarding bias.

References

1. Aggarwal S, Sinha T, Kukreti Y, Shikhar S (2020) Media bias detection and bias short term impact assessment. *Array* 6:100025. ISSN 2590-0056
2. Patankar AA, Bose J, Khanna H (2018) A bias aware news recommendation system. *CoRR* <http://arxiv.org/abs/1803.03428>
3. Vu M, Political news bias detection using machine learning
4. Gangula RR, Duggenpudi SR, Mamidi R (2019) Detecting political bias in news articles using headline attention. In: *Proceedings of the 2019 ACL workshop BlackboxNLP: analyzing and interpreting neural networks for NLP*
5. Chen WF, Al-Khatib K, Stein B, Wachsmuth H (2020) Detecting media bias in news articles using Gaussian bias distributions. *CoRR* abs/2010.10649
6. <https://mediabiasfactcheck.com/>. Accessed on 15 Apr 2021
7. <https://newsapi.org/>. Accessed on 15 Apr 2021
8. Devlin J et al (2019) BERT: pre-training of deep bidirectional transformers for language understanding. *NAACL*

A Novel Feature Reduction Methodology Using Siamese and Deep Forest Classification for Intrusion Detection



V. Gokula Krishnan, K. Sreerama Murthy, Ch. Viswanathasarma, K. Venkata Rao, K. Sankar, and D. Gurupandi

Abstract In all aspects of human existence, internet's enormous popularity has created several threats of hostile attacks. The activities carried out through the network might easily be multiplied and IDS developed. The evolution of an efficient IDS is still a major challenge because of the insidious characteristics of network intrusion behavior, particularly in the digital era when the amount of traffic and dimensions of each feature in travel are significant. A number of articles have studied various dimensional techniques; however, the attacks cannot be understood intuitively nor quickly, nevertheless require a wide range of selected features. Furthermore, a network intrusion detection (NID) method based on Siamese feature optimization and deep forest classification is developed because of the weaknesses in existing ML algorithms in NIDS such as poor exactness. The Siamese network is employed in this study for reducing the high-dimensionality characteristics by discovering the link between input data. Siamese networks function well because of common weights, where few parameters are to be learned during training and good results can be achieved with relatively little training data. The classification procedure is conducted by deep forests, a decision tree approach based on an ensemble that emphasizes the creation of deep models utilizing non-different modules. In order to demonstrate its efficacy using a number of well-known existing methodologies,

V. Gokula Krishnan (✉)

CSIT Department, CVR College of Engineering, Hyderabad, Telanagana, India
e-mail: gokul_kris143@yahoo.com

K. Sreerama Murthy

IT Department, Sreenidhi Institute of Science and Technology, Hyderabad, Telangana, India

Ch. Viswanathasarma

CSE Department, Vignana's Institute of Information Technology, Visakhapatnam, India

K. Venkata Rao

CSE Department, Guru Nanak Institution Technical Campus, Hyderabad, Telangana, India

K. Sankar

CSE Department, CVR College of Engineering, Hyderabad, Telanagana, India

D. Gurupandi

ECE Department, Panimalar Institute of Technology, Chennai, Tamil Nadu, India

the efficiency of the proposed methodology is validated in regard to accuracy of detection, performance reductions of dimensionality, and performance time.

Keywords Intrusion detection systems · Deep forest classification · KDDCup 99 dataset · Siamese network and high-dimensionality features

1 Introduction

Network safety has developed as a crucial field of research with the current interest and advancement in the expansion of internet and communication technologies in the previous decade. To protect the safety of the network and all of its corresponding properties within a cyberspace [1], it uses tools like software and IDS. The NIDS, which delivers the necessary security by continuously observing network traffic in the event of hostile and suspected behavior, was part of these network-based intrusion detection systems [2, 3].

Jim Anderson initially suggested the idea of IDS in 1980 [4]. In order to meet network security needs, several IDS products have now been developed and evolved. The huge technological development over the past decade, however, has led to a considerable increase of network size and the number of network node applications. This creates an enormous amount of critical data and shares them across various network nodes. Due to many new assaults created by the mutation of an existing attack or a new attack, it became a difficult task for safety of these data and system nodes. Nearly, all nodes in the network are exposed to threats to security. For example, for an organization, the data node can be quite essential. Every compromise to the information of the node could have a great effect on the reputation and financial performance of the organization. Inefficiency in detecting different assaults, including zero-days attacks and decreasing FAR, etc. has been demonstrated by existing IDSs [5]. This will eventually mean that the network will be required to ensure a robust security via an efficient and cost-efficient NIDS.

The researchers examined the option of using ML and profound learning (DL) techniques in order to meet the requirements of an effective identification. Both ML and DL are covered by the major components of artificial intelligence (AI) and are intended to learn usable data [6]. Due to the introduction of incredibly powerful graphics processors (GPUs) [7], this technology in network security has been quite popular in the previous decade. ML and DL are effective techniques for learning from network traffic-relevant attributes and forecasting normal and irregular actions based on the learnt models. The ML-based IDS mostly relies on the function engineering of network traffic information [8]. Whereas DL-based IDS relies not on functional engineering and because of their deep structure [9], they are decent at learning complicated features from the fresh data.

With the Siamese network and classification deep forest classifier, the major purpose of the proposed scheme is to lessen the high dimensionality. The rest of the paper is arranged as: Sect. 2 defines the investigation and its shortcoming of

existing methodologies. Section 3 delivers a description of the methodology proposed where validations of the proposed IDS are briefly detailed in Sect. 4, with existing methodologies. Finally, Sect. 5 offers the close of the research study.

2 Related Works

Yan et al. [10] have presented an IDS employing SPAS (SSAE) and SVM layered. As the function extraction approach and SVM as a classifier, the SSAE was utilized. For performing experiments, a problem of binary and multi-class categorization is considered. The findings showed the model superiority proposed when compared to the NSL-KDD dataset for the various feature selection, ML, and DL processes. While the model obtains good recognition rates for U2R and R2L assaults, the comparison of the other classes of the dataset is still less.

The distributed DBN and multilayer SVM models for broad Apache Spark-based network ID were proposed by Marir et al. [11]. DBN was utilized for extracting feature and then transmitted to the SVM ensemble and then a vote mechanism was employed to predict the result. NSL-KDD, KDDCup 99, and CICIDS2017 datasets have been assessed for their efficiency. The proposed approach has demonstrated significant efficiency in the distribution of detection of anomalous behavior.

The Wei et al. [12] DL-based DBN model has been suggested to be optimized through the combination of particle swarm and genetic algorithms. The model has been evaluated using the dataset NSL-KDD. The findings indicated significant improvements in the U2R and R2L class detection rates. The principal drawback of the model proposed is that the model's complicated structure increases its training time.

By merging CNN with long-range bidirectional short-term memory, Jiang et al. [13] suggested an efficient IDS system in a deeper hierarchy. A SMOTE is used to increase marginal samples, which helps the ideal learn the features properly. The class imbalance issue is handled. The CNN has been utilized to extract spatial properties, while temporal functions were utilized by BiLTSM. Use NSL-KDD datasets to experiment. The methodology provided achieves greater accuracy and detection rate performance. Minority data class detection rates have slightly increased, but when comparing, other attack classes are still quite poor. The training time is higher due to the intricate structure.

Zhang et al. [14] suggested a complicated CNN and gcForest multilayer IDS model. In addition, they introduced a new P-Zigzag technique for transforming raw data into two-dimensional gray characteristics. For initial detection, they utilized a better CNN model in an initial coarse grit layer. gcForest (caXGBoost) is then employed in the finely grained layer for further classification of the anomalous classes into N-1 classes. They have combined UNSW-NB15 with CIC-IDS2017 datasets using a dataset. The findings from the experiments indicate that the projected model improves greatly in comparison with single algorithms the accuracy and detection rate while minimizing the FAR.

A model IDS based on the new idea of DL few-shot learning was proposed by Yu and others [15] (FSL). The goal is to train on balanced data from the dataset using a modest quantity. DNN and CNN are embedded in the model for the extraction and scaling of the critical feature. The model efficiency of reasonable rates of detection for minority classes was demonstrated in the experimental findings obtained using NSL-KDD datasets. In order to obtain such exceptional performance for the considered dataset, only less than 2 percent of data were used for training.

Efficient CNN-based IDS is proposed by Xiao et al. [16]. The key idea is initially to use principle component analysis and AE to do feature extraction. The one-dimensional (feature set) vector is transformed into a 2-D matrix and entered into the neural network. KDDCup 99 dataset experiments were conducted. Experiments demonstrate its usefulness in terms of algorithmic time consumed during training and testing. The key problem in comparison to other attack classes is poorer detection rates for the R2L classes.

3 Proposed System

The ICT system nowadays is far more complicated, networked, and involved in the generation of extraordinarily huge data volumes, often known as big data. This is primarily because technological advances are being made, and many applications are deployed quickly. Big data is a motto that incorporates ways for extracting valuable facts from massive volumes of data. It is extremely important to allow access to large data skill, especially IDS, in the area of cyber security [17]. The progress in the field of big data technology allows diverse patterns of legal and malicious activity to be extracted from enormous volumes of data from network and system operations on time so that IDS performance can be improved. However, it is typically challenging to process big data with standard technology [18]. This part aims to explain the computer architecture and advanced methodologies chosen within the framework proposed, such as text representation methods, Siamese optimization, and the classification of deep forests. Figure 1 defines the design of the proposed flow.

3.1 Dataset Description

The tcpdump data from the DARPA intrusion detection [19] challenge dataset for 1998 were developed by KDDCup 99 dataset. For extractor characteristics from raw tcpdump data, the mining data for automated ID models (MADMAID) framework have been employed. Table 1 contains detailed statistics on the dataset. The 1998 KDDCup dataset was developed with 1000 UNIX machines and 100 people accessing the machines in the MIT Lincoln laboratory. Network data have been collected and saved for 10 weeks in tcpdump format. The data from the first 7 weeks were used for training and the rest for testing. There are two variations of KDDCup 99 dataset.

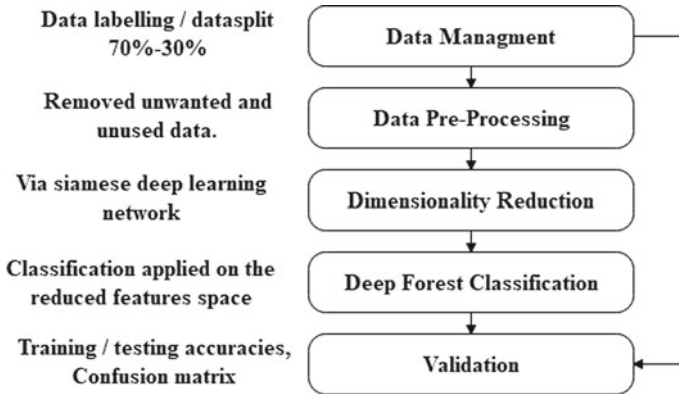


Fig. 1 Architecture of the proposed

Table 1 Training and testing sets of KDDCup 99

Attack category	Description	Train	Test
Normal	Normal connection records	97,278	60,593
Probe	Obtaining network outline details	4107	4166
DoS	Attacker goals at making network resource down	391,458	229,853
R2L	Illegal access from remote computers	1126	16,1189
U2R	Obtaining the access on a specific computer	52	228
	Total	494,021	311,029

It has complete data and a dataset of 10 percent. It consists of 41 characteristics and 5 classes (Normal, Probe, DoS, R2L, and U2R); this dataset has 41 characteristics. These characteristics are classified as below in different categories:

Basic features: tcpdump’s packet capture (Pcap) files are used to extract fundamental information instead of payload from packet headers, TCP sections, and UDP datagram. This task was performed utilizing the redesigned Bro IDS network analysis context.

Content features: The content capabilities are mined from the complete payload of domain-based TCP/IP packages in tcpdump files. In recent years, the payload feature analysis has remained a research focus. Recently, [20] developed a profound learning strategy to examine the complete load data rather than the procedure of extracting features. Content traits are utilized mostly for identifying attacks of categories ‘R2L’ and ‘U2R.’ For example, the most noticeable feature to identify the malicious behavior of the whole payload is several failed login attempts. Contrary to other attacks, the categories R2L and U2R are not significant because of occurrences occurring in one connection.

Time-based traffic features: A unique temporal two-second frame extracts time-based traffic characteristics. These are categorized into ‘the same host’ and ‘the same service’ in the past two seconds based on connection features. The aforementioned features are adjusted based on a 100-connection space to the similar host to handle slow scanning assaults. These are usually known as host-based or connection-based traffic.

3.2 Siamese Network for Dimensionality Reduction

Siamese networks are often utilized to determine the association between two objects that are comparable. Some popular applications in Siamese networks are face recognition, authentication of signatures [21], or detection of paraphrase [22]. Siamese networks operate well in these tasks because their weights portion less of the parameters during training, and with relatively little amounts of training data, they can deliver good results.

If there are huge numbers of classes with modest amounts of observations, Siamese networks are highly useful. In such circumstances, there are not sufficient data to train a deep neural network to categorize the features. If two features are in the same class, a Siamese network can instead determine. By lowering the dimension of data and utilizing a distance cost function to distinguish between classes, the network does so. The architecture of the Siamese network is exposed in Fig. 2.

The gradients model function takes dlnet and the mini-batch input data dlX1 and dlX2 on the Siamese dl network objects with its labels PairLabels. The function revenues the loss values and loss gradients in relation to the network’s learning parameters.

The Siamese network aims to provide each feature with a feature vector that comparable features and particularly different features are provided with feature vectors. The network can thus distinguish the two inputs. Looking at the contrast loss of feature vector features 1 and 1 of pair Features 1 and pairfeatures2, between the outputs of the last linked layer. The loss for a pair is contrasted with (1)

$$\text{loss} = \frac{1}{2}yd^2 + \frac{1}{2}(1 - y)\max(\text{margin} - d, 0)^2 \tag{1}$$

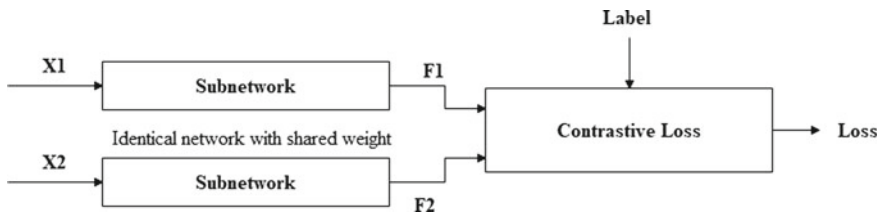


Fig. 2 Siamese network architecture

where y is identified as the value of the pair label ($y = 1$ for alike features; $y = 0$ for dissimilar features), vectors f_1 and f_2 : $d = ||f_1 - f_2||_2$.

The contrast loss has two terms, but for a certain feature pair, only one is ever nonzero. With identical characteristics, a first term cannot be null, and the gap between features f_1 and f_2 is reduced as much as possible. For different characteristics, another term can be non-null, and the interference between the features can be minimized to at least a margin distance. The lesser the margin amount, the less restrictive it is about how close a different couple can be to a loss. The decreased characteristics are, therefore, offered as a contribution to the deep forest classification, which is as follows.

3.3 Deep Forest

The deep forest is an ensemble-based decision trees approach which emphasizes on building deep models using modules which are non-differentiable. It is built around three major principles which are considered to be the reasons behind the rich accomplishments of deep models. The reasons are as follows:

- Layer-by-layer processing: It is considered one of the major factors since, no matter how complex the flat model becomes, the features of layer-by-layer processing cannot be achieved.
- In-model feature transformation: Basic machine learning models work on the original set of features. However, new features are generated during the learning process of a deep model.
- Appropriate model complexity: The fact that large datasets need complex models, basic machine learning models are limited in terms of complexity; however, it is not the case with deep models.

The overall structural working of the deep forest is separated under two broad parts: cascade forest structure and multi-grained scanning. Cascade forest structure is employed to ensure the layer-by-layer processing, while multi-grained scanning allows the model to achieve sufficient complexity.

Cascade Forest Structure

A cascade structure is employed to represent the layer-by-layer processing of raw features. Each layer in the cascade takes input (processed information) from the previous layer and feeds it into the next layer. A layer in the structure can be defined as an ensemble of decision tree forests. It is ensured that diversity is maintained while creating ensembles by including different kinds of forests.

The working in cascading stage proceeds as follows, for a given case, an approximate of class distribution will be generated by each forest. This is done by taking into consideration the training examples and fraction of different classes at the terminal or leaf node where the particular instance falls followed by averaging across all the trees in the same forest. This has also been depicted in Fig. 3. The approximated

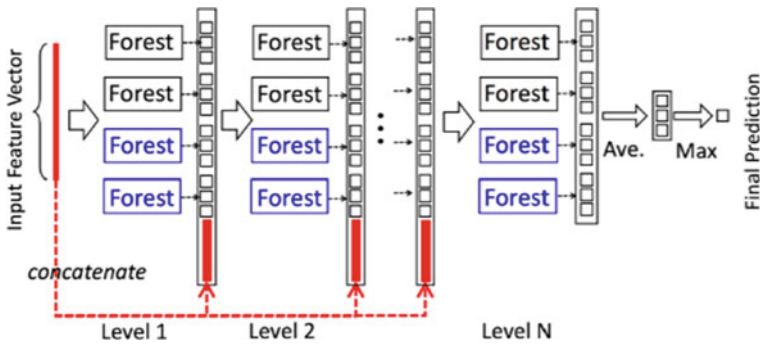


Fig. 3 Cascade forest construction

class distribution so obtained forms a vector of classes with the help of k-fold cross-validation, the vector is then concatenated with the original set of features. The result is then forwarded to the next cascading layer. K-fold cross-validation helps in reducing the risk of overfitting. The number of levels is determined automatically based on the performance on the validation set.

A striking difference in the working of deep forest and other deep models is the ability to adaptively change the model complexity by terminating the amount of training data when tolerable. This provides a considerable advantage when working with datasets of varying sizes.

Multi-grained Scanning

The cascading forest procedure is enriched with the procedure of multi-grained scanning. The inspiration behind the inclusion of the multi-grained scanning procedure was that deep models are generally well suited and also good at handling feature relationships. The whole procedure has been depicted in Fig. 4. Raw features are scanned by the sliding windows, and feature vectors are produced. The feature vectors are regarded as either negative or positive instances based on the extraction from the training sample; they are then used to produce class vectors. Completely, random forests are trained using the instances extracted from windows having the same size. Transformed features are obtained by the concatenation of generated class vectors.

The actual label of the training sample is used to assign the instances that are extracted from the windows. Though these assignments can be incorrect, they can be attributed to the flipping output method. Also, feature sampling can be performed if transformed feature vectors are too long. The sliding windows size is varied to obtain grained features vectors that are different.

The deep forest has shown a lot of promise, and its success can be attributed to the following factors:

- Fewer hyper-parameters
- Data-dependent tuning of model's complexity
- Less dependence on GPU.

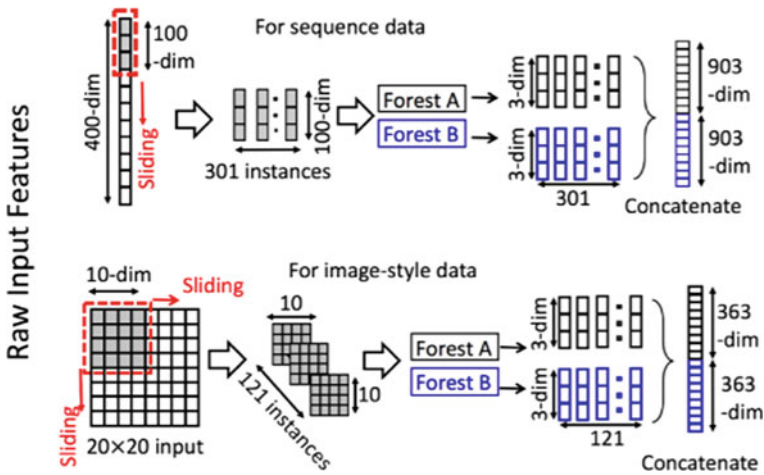


Fig. 4 Multi-grained scanning

The scalable manner uses distributed parallel ML algorithms with several optimization strategies that enable it to manage very large network and host event volumes. The scalable design also enables a quick and parallel examination of network and host-level actions by using the overall graphic processing unit (GPU) processing capacity.

4 Result and Discussion

All tests were conducted on an Ubuntu 14.0.4 LTS with Python. Use Scikit-learn to implement all traditional machine learning algorithms. Using GPU-enabled TensorFlow4, three DNNs were developed with a higher Keras5 framework backend. The GPU was NVidia GK110BGL Tesla K40, and the CPU was configured to run on 1 Gbps Ethernet network (32 GB RAM, 2 TB hard disk). The following test cases were selected to assess the performance of the proposed and different classical deep learning classifiers on KDDCup 99.

4.1 Performance Metrics

The basis truth value is necessary in the evaluation of the various statistical measures. In the instance of binary classification, the foundation truth consisted of several connection registers that were normal or attack. Let L and A be the sum of usual and attack logs in the test dataset and use the subsequent terms to determine the excellence of the classification model:

- True positive (TP)—the sum of connection records properly categorized to the usual class.
- False Negative (FN)—the sum of attack connection records incorrectly categorized to the usual connection record.
- True Negative (TN)—the sum of connection records properly categorized to the attack class.
- False Positive (FP)—the sum of normal linking records wrongly categorized to the attack linking record.

The following evaluation metrics are examined based on the above given terms.

- Accuracy:** The ratio of the predictable connection records to the whole test dataset is estimated. If the precision is higher, then the model of ML is better (Accuracy [0, 1]). Accuracy is an appropriate metric for an experimental dataset with balanced classes.
- Precision:** It guesses the ratio of correctly identified attachment logs to the number of all identified attachment logs. The ML model is better if the precision is higher (Precision [0, 1]).
- F1-score:** It is precision and recall the harmonic mean. The greater the F1-score is the better (F1-score [0, 1]).
- False Positive Rate (FPR):** It calculates the ratio of normal linking records to the number of standard connection records as attacks. The lower FPR will improve the model for ML (FPR [0, 1]).

4.2 Evaluation

The proposed evaluation has segregated into major parts such as binary classification and multi-class classification. The binary classification has detecting the attack or normal communication. Multi-class classification has detecting the various types of attack, which is presented in the dataset.

A. Binary classification

The detailed results for binary classification of several classical ML classifiers and proposed system are reported in this section.

From the Table 2, it is clearly proved that proposed achieved better accuracy (94.32%), precision (99.95%), recall (93.24%), and F1-score (96.02%) than existing techniques. The existing techniques, namely KNN, DT, SVM, and RNN, achieved nearly 92% of accuracy, where the proposed technique achieved nearly 94%. While comparing with all techniques, linear regression provides low results in all parameters.

B. Multi-class classification

The detailed results for classification of several classical machine learning classifiers and proposed system are reported in this section.

Table 2 Comparative analysis of binary class on proposed with various existing algorithms

Algorithm	Accuracy	Precision	Recall	F1-score
Linear regression	81.10	99.41	75.91	86.72
Naive Bayes	87.70	99.41	85.21	91.82
KNN	92.50	99.82	90.98	95.27
DT	92.90	99.78	91.52	95.41
SVM	92.50	99.63	91.38	95.18
RNN	92.70	99.90	91.93	95.32
LSTM	93.27	99.91	92.47	95.63
Proposed	94.32	99.95	93.24	96.02

In multi-class classification, the proposed method achieved only 94.62% of recall and 94.53% of F1-score, where naïve Bayes and linear regression achieved nearly 80–83% of recall and F1-score (refer Table 3). The existing techniques provide low performance, because the features are not reduced for final processing. But, in our proposed method, Siamese network is used for high-dimensionality reduction, which improves the performance of deep forest classifiers. Therefore, the proposed method achieved 95.62% of accuracy and 98.32% of precision.

C. Minimal feature analysis

Optimizing functionality is an essential step to detect intrusion. This is a key step toward identifying more correctly the different sorts of attacks. Without the optimization of features, a misclassification of assaults may be possible and the development of a model would take a long time. In order to detect intrusion via NLS-KDD dataset, the implication of the feature selection approach was described in [23]. The methods for selecting functions reduced the training and testing time greatly, as well as an enhanced rate for detecting intrusions. Two experiment trials are performed on limited feature sets on the KDDCup 99 to assess the performance of the proposed method and static machine learning classifications. Table 4 provides detailed results. In compared to tests in 4 feature sets, the experiments with 11 and 8 feature sets

Table 3 Comparative analysis of multi-class on proposed with various existing algorithms

Algorithm	Accuracy	Precision	Recall	F1-score
Linear regression	80.10	87.21	80.15	80.43
Naive Bayes	85.71	84.32	85.93	83.45
KNN	92.10	92.43	92.15	91.68
DT	92.46	93.48	92.44	91.81
SVM	89.52	90.21	89.54	89.03
RNN	94.53	96.61	92.52	92.24
LSTM	94.16	96.17	92.32	92.10
Proposed	95.62	98.32	94.62	94.53

Table 4 Comparative analysis of test results using minimal feature sets

Algorithm	Accuracy		
	11 features	8 features	4 features
Linear regression	84.42	90.10	87.86
Naive Bayes	89.82	90.86	88.18
KNN	90.82	92.43	89.79
DT	92.44	93.12	90.41
SVM	92.13	93.26	92.27
RNN	94.63	96.16	92.66
LSTM	96.63	94.37	91.99
Proposed	97.89	94.23	94.13

were good. In addition, experiments with 11 sets of functionalities were successful compared to the 8 sets. The performance difference of 11–8 minimum set of features is minor.

D. Detailed test results

Table 5 consists of validated results of existing techniques with proposed method for different attacks-, namely Normal, DoS, Probe, U2R, and R2L. For each attacks, we evaluated the performance of techniques by means of TPR, FPR, and accuracy. The FPR rate is zero for proposed method in the R2L and U2R attacks, where the Probe and DoS attacks has nearly 0.002–0.004. The accuracy of proposed method is 92.8% for normal and 95.5% for DoS attacks, where SVM technique achieved nearly 60–70% of accuracy for both attacks. Therefore, it is clearly proved that our proposed method achieved less FPR and high accuracy than other existing techniques for all attacks.

5 Conclusion and Future Work

The framework used the distribution in real time of very large data with the Siamese network model with deep forest. In comparison to standard machine learning classifications, the proposed model was thoroughly assessed in different IDS datasets. This model was chosen. In all situations, we found that the proposed performance in comparison with standard machine learning classifiers is above that of the classification results. Our proposed architecture is superior to the classical classification of IDS machines implemented previously. To our best knowledge, this is the only framework that is capable of distributing network and host activities via deep forest to more accurately identify an assault. By adding nodes to the present cluster, the implementation time of the proposed scheme can be increased. Furthermore, the system presented does not include specific information on malware structure and features. In general, by training complicated, advanced hardware architectures using

a distributed manner, the performance can also be increased. In this investigation, they were not trained with the IDS benchmark datasets because of considerable computational cost of the sophisticated suggested designs. This is a critical challenge in an adverse environment and one of the main directions for future work.

References

1. Tarter A (2017) Importance of cyber security. Community policing-a European perspective: strategies, best practices and guidelines. Springer, New York, NY, pp 213–230
2. Li J, Qu Y, Chao F, Shum HP, Ho ES, Yang L (2019) Machine learning algorithms for network intrusion detection. *AI in Cybersecurity*. Springer, New York, NY, pp 151–179
3. Lunt TF (1993) A survey of intrusion detection techniques. *Comput Sec* 12(4):405–418. [https://doi.org/10.1016/0167-4048\(93\)90029-5](https://doi.org/10.1016/0167-4048(93)90029-5)
4. Anderson JP (1980) Computer security threat monitoring and surveillance. James P Anderson Co, Fort Washington, PA
5. Hoque MS, Mukit M, Bikas M, Naser A (2012) An implementation of intrusion detection system using genetic algorithm. [arXiv:1204.1336](https://arxiv.org/abs/1204.1336)
6. Prasad R, Rohokale V (2020) Artificial intelligence and machine learning in cyber security. *Cyber security: the lifeline of information and communication technology*. Springer, New York, NY, pp 231–247
7. Lew J, Shah DA, Pati S et al (2019) Analyzing machine learning workloads using a detailed GPU simulator. Paper presented at: Proceedings of the IEEE international symposium on performance analysis of systems and software (ISPASS). IEEE, Madison, WI, pp 151–152
8. Najafabadi MM, Villanustre F, Khoshgoftaar TM, Seliya N, Wald R, Muharemagic E (2015) Deep learning applications and challenges in big data analytics. *J Big Data* 2(1):1. <https://doi.org/10.1186/s40537-014-0007-7>
9. Dong B, Wang X (2016) Comparison deep learning method to traditional methods using for network intrusion detection. Paper presented at: Proceedings of the 8th IEEE international conference on communication software and networks (ICCSN). IEEE, Beijing, China, pp 581–585
10. Yan B, Han G (2018) Effective feature extraction via stacked sparse autoencoder to improve intrusion detection system. *IEEE Access* 6:41238–41248
11. Marir N, Wang H, Feng G, Li B, Jia M (2018) Distributed abnormal behavior detection approach based on deep belief network and ensemble svm using spark. *IEEE Access* 6:59657–59671
12. Wei P, Li Y, Zhang Z, Hu T, Li Z, Liu D (2019) An optimization method for intrusion detection classification model based on deep belief network. *IEEE Access* 7:87593–87605
13. Jiang K, Wang W, Wang A, Network Intrusion WH (2020) Detection combined hybrid sampling with deep hierarchical network. *IEEE Access* 8:32464–32476
14. Zhang X, Chen J, Zhou Y, Han L, Lin J (2019) A multiple-layer representation learning model for network-based attack detection. *IEEE Access* 7:91992–92008
15. Yu Y, Bian N (2020) An intrusion detection method using few-shot learning. *IEEE Access* 8:49730–49740
16. Xiao Y, Xing C, Zhang T, Zhao Z (2019) An intrusion detection model based on feature reduction and convolutional neural networks. *IEEE Access* 7:42210–42219
17. Tang M, Alazab M, Luo Y Big data for cybersecurity: vulnerability disclosure trends and dependencies. *IEEE Trans Big Data* (to be published)
18. Kezunovic M, Xie L, Grijalva S (2013) The role of big data in improving power system operation and protection. In: Proceedings IREP symposium bulk power system dynamic control-ix optimising, security control emerging power grid (IREP), pp 1–9
19. Debar H, Dacier M, Wespi A. Towards a taxonomy of intrusion-detection systems. *Comput Netw*. 1999;31(8):805–822. [https://doi.org/10.1016/S1389-1286\(98\)00017-6](https://doi.org/10.1016/S1389-1286(98)00017-6).

20. Wang Z (2015) The applications of deep learning on traffic identification, BlackHat, pp 21–26
21. Bromley J, Guyon I, LeCunn Y, Säckinger E, Shah R (1994) Signature verification using a “Siamese” time delay neural network. In: Proceedings of the 6th international conference on neural information processing systems (NIPS 1993), pp 737–744
22. Wenpeg Y, Schütze H (2015) Convolutional neural network for paraphrase identification. In: Proceedings of 2015 conference of the north American chapter of the ACL, pp 901–911
23. Alazab A, Hobbs M, Abawajy J, Alazab M (2012) Using feature selection for intrusion detection system. In: Proceedings International Symposium Communication and Information Technology (ISCIT), pp 296–301

A Review on Deepfake Media Detection



Rajneesh Rani, Tarun Kumar, and Mukund Prasad Sah

Abstract Deepfake is a machine learning and artificial intelligence-based technique which is used to generate fake faces and replaces them in a video or image. These days deepfakes are mostly used to spread fake news, audio, video, etc. Deepfakes are also used in political campaigns to manipulate the videos of leaders and spread hatred. Today, it has become very easy to generate deepfake images as there are various commercial softwares available for generating deepfakes; also, there are various free of cost apps available like faceapp, fakeapp, etc. The website thispersondoes-notexist.com generates a fake person image that does not exist every time you click it. The automation in video manipulation generates more threat to original content as it is becoming more and more easier to manipulate images and generate fake news. It can be very dangerous in the upcoming time to detect what is fake and what is real. The main component which makes deepfakes more and more real is general adversarial networks (GANs); by using GAN, we can generate high-quality deepfakes which cannot be detected by the human eye. There are various techniques generated to detect deepfakes, but to the best of our knowledge, we can say that there is no foolproof method to detect deepfakes, and there is a strong need for a technique which can prevent current facial recognition systems from deepfakes. In this paper, we try to give a brief review of existing deepfake detection techniques. There are many techniques developed in this area but most of them can be categorized in facial artifacts, neural networks, 3D head position.

Keywords Deep neural networks · Face manipulation · Computer vision · Deepfakes · CNN · RNN · Face swapping

R. Rani · T. Kumar (✉) · M. P. Sah
Dr. B R Ambedkar National Institute of Technology, Jalandhar, Punjab, India
e-mail: tk4tarunkumar@gmail.com

R. Rani
e-mail: ranir@nitj.ac.in

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
H. Sharma et al. (eds.), *Communication and Intelligent Systems*, Lecture Notes
in Networks and Systems 461, https://doi.org/10.1007/978-981-19-2130-8_28

343

1 Introduction

Deepfake images or videos refer to the images/videos of a person or a group of people that can be recognized doing acts that they have never done; these kind of multimedia are false and machine generated. The actions performed by people in these fake media can be deceptive for the general public and can tarnish the image of such people in public. Nowadays, there are several methods that can generate life-like real deepfakes (images and videos).

Earlier before the use of sophisticated tools to generate deepfake images or videos, Adobe Photoshop or GNU Image Manipulation Program (GIMP) was used in most cases. Those fake images generated by these old methods can be detected easily as compared to the newer methods such as face swapping, using GAN or machine learning techniques.

The expression ‘deepfake’ alluded to a deep learning-based strategy ready to make counterfeit pictures or recordings by trading the substance of an individual by the essence of someone else [1]. This term first came into light after a Reddit client named ‘deepfakes’ asserted in late 2017 to have an AI calculation, built up that assisted him with trading famous actors’ faces into pornography recordings. Notwithstanding phony erotic entertainment, a portion of the more hurtful uses of such phony substance incorporate phony news, scams, and monetary misrepresentation. Customarily, the number and authenticity of facial controls [2] have been restricted by the absence of modern altering devices, the area skill required, and the complex and tedious cycle involved. In reaction to those undeniably refined and practical controlled substance, huge endeavors are being completed by the examination network to configuration improved techniques for face control recognition.

In a study, published in 2017 by Cognitive Research [3] asked people to detect and classify by visual inspection if the shown image is doctored or not. Only 62–66% were correctly classified. And the results were lower than classification in finding the area of manipulation in doctored images. In another study of similar type [4], only 58% images were correctly classified.

From the above results, one can conclude that the human brain finds it difficult to differentiate between a real person’s image and a synthetic generated image. Hence, people’s thinking or views built around a personality can be changed or altered by the use of these deepfake images and videos.

Several algorithms and tools have been developed to tackle this day-by-day increasing problem of automatically machine generated deepfake images and videos. At present times, there are several applications available which can easily generate deepfakes. The quality of fake images and videos is life-like and realistic to a much greater extent by making use of these fake media generating applications like faceapp, FaceSwap, etc.

Nowadays, these advancements in this field have made it a lot easier for a person to make a fake video of a famous leader or a famous personality doing some sort of illegal things that may lead to violence, or an army leader saying some foul language,

Fig. 1 Examples of altered and fake images



or doing some funny things. These fake contents lead to a threat to society, security of a nation and democracies.

In Fig. 1, not a single person actually exists in reality. These are 100% artificially fabricated images sourced from <https://thispersondoesnotexist.com>, is a website which shows a new fake image every time you visit the website. These images in Fig. 1 are an example to illustrate what can be achieved by bringing these technologies into play.

This review paper tries to do a review on a new kind of threat of fake media circulation on the internet which can change views, perspective of masses, and can be manipulated, be it an election or in general daily life of a popular personality. There are greater risks and much higher potential threats to public impression of the most influential people around. A large number of their followers actually take their advice and relate to them at a personal level, which can be ruined easily after the circulation of fake media like images, videos, GIFs, etc. to turn the general public into doing unexpected and unwanted things or activities.

Deepfakes became popular because of the quality of result it can produce; it is a technique which is easy to use as there are many tools available which can be used to make deepfake videos that too with a very high quality. Deepfake creation is started with the encoder and decoder networks which can be used to swap faces and create deepfakes; later to improve the quality of deepfakes, GAN is introduced which can generate a high-quality deepfake which can be indistinguishable by eyes.

General adversarial network (GAN) is a technique of machine learning in which two neural networks compete against each other. While competing, if one of the networks wins, the other will be considered a loss, so it can be said like a zero sum

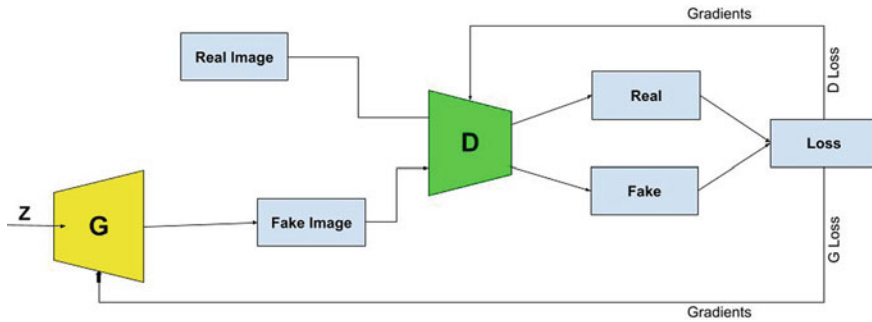


Fig. 2 GAN architecture

game. Though GAN can also be utilized for working on a certain count of special things, the major application of GAN is the creation of deepfakes. In this technique, input is the training set. GAN learns from the training set, and its output is the data having the same details as the input training set. If the input is pictures, GAN can generate new photographs from the input pictures. Interesting thing is that the output will be so realistic that a human cannot say it is fake. Figure 2 will give a rough idea of the General Adversarial Network.

The GAN architecture consists of two neural system network:

- **Discriminator D:** In this system, the input is two media files; one is real, and the other one is fake. This Discriminator D has to identify which media file is real and which one is fake.
- **Generator G:** This system G is made to fool the discriminator, so that the discriminator gets confused between the real media and the fake media [5].

Both the neural systems networks are altogether given training during the learning stage. In every stage, for a real media, the generator will generate fake media, and then both real and fake pictures will be sent to the discriminator, till the time when the discriminator is able to distinguish between real and fake media.

2 Datasets

2.1 Celeb-DF

Celeb-DF is a large-scale dataset; this dataset has a collection of 590 real videos and 5639 deepfake videos Fig. 3. Both of them are of high quality, and they are of celebrities that are generated using the modernly improved synthesis of GAN network [6]. Each video is 13 s long and has a standard frame rate of 30 frames per second (fps). This dataset can be found at <http://www.cs.albany.edu/~lsw/celeb-deepfakeforensics.html>.

Fig. 3 Celeb-DF [6]

2.2 FaceForensics++

FaceForensics++ is a facial forgeries dataset that has over 1.8 million manipulated images; it can be found here. The dataset is created by applying manipulation techniques like Face2Face, FaceSwap, DeepFakes, and NeuralTextures to the videos collected from YouTube, to make the dataset more realistic [7].

FaceSwap is an algorithm which is based on graphics. It is about transferring the facial region from source video to the target video [7].

Face2Face is a facial re-modification system whose main focus is on expressions [7]. Its algorithm moves a source video expression to a target video expression, while considering to maintain the identity of the target human [7].

Thies et al. [8] are an example of a rendering approach based on NeuralTextures. It utilizes the first video information to gain proficiency with a neural surface of the objective individual, including a delivering organization. It is taught by reconstructing the photometric system of loss on combining with an adversarial loss.

2.3 *DeepFake-TIMIT*

DeepfakeTIMIT is an information base of recordings where countenances are traded utilizing the open source GAN-based methodology which, thusly, was created from the first autoencoder-based deepfake calculation. In the creation of TIMIT database, 16 similar pair of people are selected from the TIMIT database which is publicly available. For each 32 subjects, we consider two models: an LQ quality and an HQ quality. Since there are ten recordings for every individual in TIMIT information base, 320 recordings are generated compared to every rendition, bringing about 620 absolute recordings with faces traded. From these 320 recordings, a total of 10537 real and 34,023 fake images are generated and stored in the database. No audio manipulation is done in the channel, so audio originality is preserved [9].

3 Deepfake Detection

As the techniques for developing deepfake are getting advanced day by day, we have a huge threat to originality of content and material and news that we use in our daily life. There are various types of manipulations possible on an image like: Manipulation of facial artifacts, i.e., changing eye color, changing of lips shape, changing of nose color and shape, etc., another type of manipulation can be done via general adversarial networks (GANs) which are specially designed to develop deepfakes, this technique comes under deep learning a branch of machine learning, GAN is firstly trained on a particular dataset, and then they can generate deepfakes. GAN creates new data that resembles training data, like if we train a GAN on a human face, then GAN can generate a human face that does not exist. GAN is mainly composed of two components generator and discriminator, generator tries to develop an image that can pass through discriminator and discriminator tries to stop them (detecting them), this way GAN can generate fake human faces that do not exist. We have various techniques available for detection of deepfakes but as we are going to try to categorize them in three major categories, i.e., facial artifacts-based approaches, neural networks-based approaches, and head position-based approaches. In facial artifacts-based approaches, facial components like eyes, nose, lips, mouth, and ears are used to detect deepfakes; whenever a deepfake is generated, they mostly have issues with these artifacts; for example in an image, left eye color is different from right eye color, shading artifacts on nose because of light illumination, missing reflection details in eyes, imprecise face geometry, teeth are taken as single white part, etc. So when we consider some techniques which can identify those detects, then we can easily detect deepfakes. In neural networks-based approach, we have various techniques like CNN, RNN, VGG, etc.; there are some mixed techniques also where we use some combination of facial artifacts and neural networks. We also have some ensemble methods in neural networks. In head position technique, we consider rotation of head around x -axis, y -axis, distance between various positions, etc.

3.1 Facial Artifacts

First step for detecting deepfakes is to segment the face from the image with the help of various tools or techniques available. Then we process that segmented face for possible faults which are not visible from the human eye and can only be detected after applying different techniques; one of them is using facial artifacts for detecting. Same procedure is applied for detecting deepfakes in videos; we work on frame by frame in case of video. When using facial artifacts, we mean to expose some common mistakes in the images like different eye color, or incorrect face geometry or incorrect teeth structure as the author used in [1] as shown in Fig. 4. The author segmented different parts of face from image and applied various techniques on them and completely relied on those imperfections in the images but when a deepfake is generated, we can eliminate those imperfections by doing some simple postprocessing on them and after eliminating those imperfections will not work on deepfakes as author only tried this technique on deepfakes and face2face, have some prerequisites and can only be applied on certain images (e.g., open eyes, visible teeth etc.), but on those images author got AUC score of 0.866.

In [10], author tried to detect deepfake based on discrepancies between faces and their context; deepfake is made by some face swapping methods aiming to make faces realistic leaving the context unchanged in Fig. 5. Author used an approach where he made two networks, one which considers face region by segmenting it from image and other considers its context like ears, hair, etc. Those two networks are used to detect the discrepancies from the faces; by using these two networks, the author got very good results. The author used the datasets FaceForencis++, Celeb-DF-v2, DFDC, to train and detect the deepfakes. The author achieved an AUC score of 0.997 on FaceForencis++ dataset and 0.66 on Celeb-DF which are the highest till date according to our knowledge.

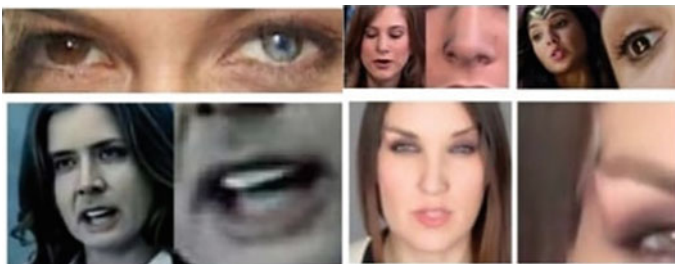


Fig. 4 Various facial artifacts showing defects in face

Fig. 5 Extraction of face from image leaving context behind [10]



3.2 Neural Networks

All the techniques that are being developed in recent times include neural networks in some way. As neural networks we can say is the best thing in machine learning because we can develop many structures of these and these can be used in various fields like in military, weather forecasting, etc. Neural networks have word neural because neural networks are generated by taking inspiration from brain neurons, they also have similar structure and similar workings, neural networks takes input and process it shows result, neural networks can consist of various hidden layers, and there are various types of neural networks available like CNN, RNN, VGG, ResNet, XceptionNet, etc. We will discuss the techniques which are developed in recent times for detection of deepfakes. In one of the techniques that are generated for deepfake detection, author used a combination of DenseNet, alignment, and bidirectional recurrent neural networks [11]; in that, author used deepfake, face2face, FaceSwap to train and test his method and got very good results of accuracy 96.9%, 94.35%, and 96.3%, respectively. Model used in [11] is shown in.

In one of the papers, the author used the improved version of VGG that is NA-VGG to detect deepfake images [12]. NA stands for image noise and image augmentation. In this paper firstly, the SRM filter layer is used to detect tampering artifacts which cannot be seen in RGB channels; this will highlight the image noise, and then image noise map is augmented and fed into the network for training and later for prediction on different images. Author tested his model on different datasets like UADFV, DeepFake-TIMIT, FF++ and Celeb-DF and got an average AUC score of 0.857, which is not better than previous method discussed above but certainly better than normal VGG which has an average AUC score of 0.564.

We have another approach for detection in which we deal with mesoscopic properties of image and try to detect based on those properties in [13] author used mesoscopic properties for detection and have developed two different models for training and prediction. Meso-4 and MesoInception-4 are the two networks developed by the author by which he got good accuracy on different datasets; on face2face dataset, he got accuracy of 94.6% and 96.8% on both models, respectively. In both of these models, very few layers and very few parameters are used but surprisingly models have given good results. Main thing that the author described in the paper is compression ratio, as when we increase compression, we start to lose information and so models that we make could not produce good results; if we use light compression, it will be easy for our models to train on and also we can save time and money on hardware we use. A compression rate of 23 which is light compression is used on

FaceForencis dataset, compression rate of 40 which is high produces less accuracy of 83.2% and 81.3%, respectively, on both models.

For detection of deepfake video, we have another approach which uses a combination of CNN and RNN [14]. CNN is used for extracting frame feature from a video, and consequently, those features are forwarded to the LSTM network, which is a type of RNN, LSTM which is 2048 wide and 0.5 dropout is used and in final layer softmax is used for calculation of classification percentage that whether the fed frame is manipulated or not. Author got a testing accuracy of 97.1% when used with 80 frames sequence from a video, which is quite good considering the data is collected from various sources. In comparison to this method we have another method developed which uses XceptionNet for spatial feature extraction and steganalysis feature extraction and by combining these two different techniques they got an accuracy of 98.57 on FF++ and 95.33 on deepfake dataset and called the developed method SST-Net [17]. Some other techniques are also there which include two-branch RNN [18] and CLRNet [24]; in two-branch RNN model, architecture contains two independent blocks one of which works on RGB coloring and other block has a layer of deep Laplacian of Gaussian filter to discard visual face content, and later, these blocks merged and some other layers are used, and at final, bidirectional LSTM is used. In CLRNet, author analyzed that in case of deepfakes videos, there are inconsistencies in consecutive frames which can be sudden change in brightness or contrast on a specific region of face and other things so they developed a new method called convolutional LSTM residual network that can identify these inconsistencies.

Some techniques are based on heart rates also; there are technologies which can predict heart rate of a person by analyzing the face of a person using various techniques like Eulerian video magnification [25], average optical intensity in the forehead, head motions [26], etc. Heart rate of fake videos can be used to distinguish it from original video [27]. NeuralODE [28] which is the first time is used for predicting heart rate of deepfake according to chen et al. [29]. FakeCatcher is a technique developed for detection of synthetic portrait videos using biological signals like heart rate [27]. FakeCatcher achieved a pairwise separation accuracy of 99.39% on datasets like FaceForencis and deepfake.

3.3 Head Positions

Deepfakes are created by splicing synthesized face region into the original one and in doing so introducing errors that we try to detect using different techniques. One of the techniques is developed to detect those errors using 3D head poses from face images [30]; in this techniques, author used 3D head poses and consequently SVM classifier on real face images and deepfakes based on 3D head position for classification; using this technique, they got a AUC score of 0.89, which shows that 3D pose is also a good

way to detect deepfakes; this feature can be further used to classify deepfakes; there is a great scope of increasing accuracy using 3D poses as there is no other research using this parameter for deepfake detection, and it can be further developed more to get higher accuracy.

4 Performance Measures

Here, in the papers cited, the results are shown using different parameters such as overall accuracy or AUC. Here, accuracy refers to the test accuracy of the respective model, and equation for accuracy is given in Eq. 1.

$$\text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN}) \quad (1)$$

where TP is ‘true positives,’ TN is ‘true negatives,’ FP is ‘false positives,’ FN is ‘false negatives.’

The following Table 1 contains compilation of all the results observed/obtained in form of accuracies of several methods used by respective authors in the research papers referred.

Multiple methods were used and also on different datasets to compare results and accuracies on different sets of images or sequence of images.

Also, measurement of the parameter area under curve (AUC) is based on different terms, true positive rate (TPR), specificity, and false positive rate (FPR). Equations 2, 3 and 4 define TPR, specificity, and FPR, respectively.

$$\text{TPR} = (\text{TP}) / (\text{TP} + \text{FN}) \quad (2)$$

$$\text{Specificity} = (\text{TN}) / (\text{TN} + \text{FP}) \quad (3)$$

$$\text{FPR} = (\text{FP}) / (\text{TN} + \text{FP}) = 1 - \text{Specificity} \quad (4)$$

where TP is ‘true positives,’ TN is ‘true negatives,’ FP is ‘false positives,’ FN is ‘false negatives.’

Now, AUC is the area defined under the receiver operating characteristics (ROC) curve which is the plotting of TPR on y-axis and FPR on x-axis. AUC reaching close to 1 means that the model has a higher level of separability of true positives and true negatives detected, AUC near to 0 means it has worse level of separability. AUC with 0.5 value has no capacity of separability whatsoever between true positives and true negatives.

Table 1 Comparison of several deepfake detection approaches methods

S. No.	Year	Paper	Method	Dataset	Manipulation technique	Accuracy (%)	Average AUC	
1.	2018	Afchar et al. [13]	MesoInception-4	FF++ Internet	DeepFake	-	0.984	
			Meso-4				0.969	
			MesoInception-4		Face2Face	-	0.953	
			Meso-4				0.953	
2.	2018	Guera et al. [14]	Conv. LSTM 20	HOHA Internet	-	96.7	-	
			Conv. LSTM 40			97.1		
			Conv. LSTM 80			97.1		
3.	2019	Sabir et al. [11]	DenseNet + BiDirRNN	FF++	DeepFake	96.9	-	
					FaceSwap	96.3		
					Face2Face	94.35		
4.	2019	Matern et al. [1]	KNN	Glow	-	-	0.843	
				PROGAN			0.852	
			MLP	Internet	DeepFake	-	0.851	
			LOGREG				0.784	
5.	2020	Nirkin et al. [10]	FSGAN	FF-DF	FaceSwap	-	0.997	
						Celeb-DF		0.660
						FF++	Face2Face DeepFake FaceSwap NeuralTextures Pristine	-
6.	2020	Montserrat et al. [15]	Logits	DFDC	-	85.51	-	
					AFW		87.90	
					GRU		92.61	
7.	2020	Zhang et al. [16]	CNN model	MUCT-DB	DeepFake	-	0.976	
8.	2020	Chang et al. [12]	NA-VGG	Celeb-DF	-	-	0.857	
9.	2020	Wu et al. [17]	SSTNet	FF++	DeepFake	95.33	-	
					Face2Face	90.48		
					FaceSwap	94.09		
					Combined	98.57		
10.	2020	Masi et al. [18]	Two-branch RNN	FF++ (HQ)	-	-	0.9912	
						FF++ (LQ)		0.9110
						Celeb-DF	-	0.7665
						DeepFake		0.7341

FF++ = FaceForensics++, FF-DF = FaceForensics, GRU = Gated Recurrent Unit, DFDC = DeepFake Detection Challenge, AFW = Automatic Face Weighing, MUCT-DB [19], HOHA [20], GLOW [21], PROGAN [22], DFDC [23]

5 Future Scope

In the field of deepfake detection, there is scope for further research. Such as until now, datasets are created in an unconstrained environment, so new and more challenging datasets can be made to test available models. Measuring different aspects of trained models other than accuracy or area under curve and determination of deepfake on live videos in real time. There is also a great scope of research in audio deepfakes where audio is manipulated instead of face. More robust deepfake detection methods are required which will work on any kind of deepfake either it is facial artifacts manipulation or face swapping or audio deepfakes. There can also be a centralized portal where various datasets and models can be stored with various functionalities like on uploading a deepfake video we can apply different models on it for detection purposes.

6 Conclusion

In this survey paper, we tried to explain the process of creation of deepfake media, which is now much evolved and is able to generate life-like and realistic images. We have also compared several techniques in Table 1. In coming time, we may witness more realistic deepfake videos. There is also scope in detecting deepfake source from reverse engineering. To fight current deepfake media problem, we are not sufficiently equipped with technologies which can detect it. In recent challenge presented by Facebook to detect deepfake, there are not much promising results occurred, and there is a huge scope of research possible in this field, by which we can detect deepfakes with more accuracy and precision and can save society from deepfakes.

References

1. Matern F, Riess C, Stamminger M (2019) Exploiting visual artifacts to expose deepfakes and face manipulations. In: 2019 IEEE Winter applications of computer vision workshops (WACVW). IEEE, pp 83–92
2. Ding X, Raziei Z, Larson EC, Olinick EV, Krueger P, Hahsler M (2020) Swapped face detection using deep learning and subjective assessment. *EURASIP J Inf Secur* 2020:1–12
3. Nightingale SJ, Wade KA, Watson DG (2017) Can people identify original and manipulated photos of real-world scenes? *Cogn Res Principles Implications* 2(1):30
4. Schetinger V, Oliveira MM, da Silva R, Carvalho TJ (2017) Humans are easily fooled by digital images. *Comput Graph* 68:142–151
5. Wang Z, She Q, Ward TE (2020) Generative adversarial networks in computer vision: a survey and taxonomy. arXiv preprint [arXiv:1906.01529](https://arxiv.org/abs/1906.01529)

6. Li Y, Yang X, Sun P, Qi H, Lyu S (2020) Celeb-df: A large-scale challenging dataset for deepfake forensics. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 3207–3216
7. Rossler A, Cozzolino D, Verdoliva L, Riess C, Thies J, Nießner M (2019) Faceforensics++: Learning to detect manipulated facial images. In: Proceedings of the IEEE international conference on computer vision, pp 1–11
8. Thies J, Zollhöfer M, Nießner M (2019) Deferred neural rendering: image synthesis using neural textures. *ACM Trans Graph (TOG)* 38(4):1–12
9. Dataset, <https://www.idiap.ch/dataset/deepfaketimit>
10. Nirkin Y, Wolf L, Keller Y, Hassner T (2020) Deepfake detection based on the discrepancy between the face and its context. arXiv preprint [arXiv:2008.12262](https://arxiv.org/abs/2008.12262)
11. Sabir E, Cheng J, Jaiswal A, AbdAlmageed W, Masi I, Natarajan P (2019) Recurrent convolutional strategies for face manipulation detection in videos. *Interfaces (GUI)* 3(1)
12. Chang X, Wu J, Yang T, Feng G (2020) Deepfake face image detection based on improved vgg convolutional neural network. In: 2020 39th Chinese control conference (CCC). IEEE, pp 7252–7256
13. Afchar D, Nozick V, Yamagishi J, Echizen, I.: Mesonet: a compact facial video forgery detection network. In: 2018 IEEE international workshop on information forensics and security (WIFS). IEEE, pp 1–7
14. Güera D, Delp EJ (2018) Deepfake video detection using recurrent neural networks. In: 2018 15th IEEE international conference on advanced video and signal based surveillance (AVSS). IEEE, pp 1–6
15. Montserrat DM, Hao H, Yarlagadda SK, Baireddy S, Shao R, Horváth J, Bartusiak E, Yang J, Güera D, Zhu F et al (2020) Deepfakes detection with automatic face weighting. arXiv preprint [arXiv:2004.12027](https://arxiv.org/abs/2004.12027)
16. Zhang W, Zhao C, Li Y (2020) A novel counterfeit feature extraction technique for exposing face-swap images based on deep learning and error level analysis. *Entropy* 22(2):249
17. Wu X, Xie Z, Gao Y, Xiao Y (2020): Sstnet: Detecting manipulated faces through spatial, steganalysis and temporal features. In: ICASSP 2020-2020 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, pp 2952–2956
18. Masi I, Killekar A, Mascarenhas, RM, Gurudatt SP, AbdAlmageed W (2020) Two-branch recurrent network for isolating deepfakes in videos. In: European conference on computer vision. Springer, pp 667–684
19. Milborrow S, Morkel J, Nicolls F (2010) The MUCT landmarked face database. Pattern Recognition Association of South Africa. <http://www.milbo.org/muct>
20. Laptev I, Marszalek M, Schmid C, Rozenfeld B (2008) Learning realistic human actions from movies. In: 2008 IEEE conference on computer vision and pattern recognition. IEEE, pp 1–8
21. Kingma DP, Dhariwal P (2018) Glow: generative flow with invertible 1x1 convolutions. arXiv preprint [arXiv:1807.03039](https://arxiv.org/abs/1807.03039)
22. Karras T, Aila T, Laine S, Lehtinen J (2017) Progressive growing of gans for improved quality, stability, and variation. arXiv preprint [arXiv:1710.10196](https://arxiv.org/abs/1710.10196)
23. Dolhansky B, Howes R, Pflaum B, Baram N, Ferrer CC (2019) The deepfake detection challenge (dfdc) preview dataset. arXiv preprint [arXiv:1910.08854](https://arxiv.org/abs/1910.08854)
24. Tariq S, Lee S, Woo SS (2020) A convolutional lstm based residual network for deepfake video detection. arXiv preprint [arXiv:2009.07480](https://arxiv.org/abs/2009.07480)
25. Bennett SL, Goubran R, Knoefel F (2016) Adaptive eulerian video magnification methods to extract heart rate from thermal video. In: 2016 IEEE international symposium on medical measurements and applications (MeMeA). IEEE, pp 1–5
26. Balakrishnan G, Durand F, Guttag J (2013) Detecting pulse from head motions in video. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3430–3437
27. Ciftci UA, Demir I, Yin L (2020) Fakecatcher: detection of synthetic portrait videos using biological signals. *IEEE Trans Pattern Anal Mach Intell*
28. Fernandes S, Raj S, Ortiz E, Vintila I, Salter M, Urosevic G, Jha S (2019) Predicting heart rate variations of deepfake videos using neural ode. In: Proceedings of the IEEE international conference on computer vision workshops, p 0

29. Chen RT, Rubanova Y, Bettencourt J, Duvenaud DK (2018) Neural ordinary differential equations. In: Advances in neural information processing systems, pp 6571–6583
30. Yang X, Li Y, Lyu S (2019) Exposing deep fakes using inconsistent head poses. In: ICASSP 2019-2019 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, pp 8261–8265

Investigation of Error-Tolerant Approximate Multipliers for Image Processing Applications



D. Tilak Raju and Y. Srinivasa Rao

Abstract Low-power, high-speed real-time computing is critical for various applications, with digital signal processing (DSP), image processing, the internet of things, and neural networks. Multiplication and division algorithms account for 86% of the data processing time in a real-time three-dimensional graphics system. The approximate multiplier (AM) may be the key to improving hardware efficiency and speeding up multiplication operations. The AM has been the primary arithmetic component for many applications in the past ten years. However, comprehensive literature on the entire development history and processes of AM findings, error analysis, and applications is missing in one location. As a result, this article outlines the history and advancements of AM architectural design and prospective study topics for future advancements. This thorough study also discusses the methods researchers utilize to enhance AM design and provide an edge over other mentioned AM.

Keywords Approximate multipliers · Approximate computing · Approximate adders · Compressors

1 Introduction

Approximate computing (ACG) is a problematic approach that generates a potentially approximate result rather than an exact guaranteed result; this result may be utilized for applications that need approximate results [1, 2]. ACG was developed as a result of the discovery in various circumstances. On the other hand, the exact calculation requires many resources; permitting bounded approximation might yield asymmetrical increases in speed and energy while keeping fair accuracy. Allowing a 5% decrease in classification accuracy in the k-means clustering algorithm (KMCA),

D. Tilak Raju (✉)

Department of Electronics and Communication Engineering, Vignan's Institute of Engineering for Women, Kapujagaraju Peta, Visakhapatnam, A.P., India
e-mail: tilakraju55@gmail.com

Y. Srinivasa Rao

Department of Instrument Technology, Andhra University, Visakhapatnam, A.P., India
e-mail: srinniwasarau@gmail.com

for example, may save 50 times more energy than a 100% perfect classification [1]. CMOS device physical dimensions have been shrinking and are now nearing a few nanometers. The computer burden is increased by multimedia processing (graphics, image, audio, and video), recognition, search, data mining, and machine learning. In the applications mentioned above, ACG has played a vital role. The fast advancement of integrated circuit technology has become a major driving force for real-time signal processing. As is common knowledge, DSP employs various mathematical operations (such as subtraction, square root, multiplication, division, power, and addition). Multiplication and division algorithms account for 86% of the data processing time [3]. Multiplication is often used in image processing (IM) and DSP [4]. Approximate (APP) multipliers were created essentially utilizing three techniques: approximation in partial product (PP) creation, ii) approximation in the PP tree, and iii) approximation in PP summation. Jiang et al. [5] analyzed the properties of various AM implementations in VHDL utilizing these various methodologies. However, they left out error analysis and its applications.

In the second section, this article primarily provides an overview of AMs, and the third section addresses a few recent famous AMs findings regarding hardware and error metrics. Fourth section examines the two primary uses of AMs before concluding in the fifth section.

2 Literature Survey

Ha et al. [6] used the proposed 4-2 approximate compressor (AC) to create low-power (LP) and high-performance (HP) AM (PPAM), as well as an error recovery module. PPAM applied image sharpening (IMS) and proposed PPAM reduced power 21.5%, area 25.5%, and delay 18% compared to the same multiplier.

Instead of using the APP parallel multiplier, Alouani et al. [7] devised and implemented the heterogeneous blocks (HB) AM (HBAM) to improve speed. Some power trade-offs exist in developing the KMCG and Sobel filter (SF) using the suggested HBAM.

Liu et al. [8] invented the non-iterative APP Logarithmic Multipliers (ALMS). These ALMS designed with three inexact mantissa adders produce a 17.9% NMED and a 37% decrease in power-delay product compared to existing AM (PDP). The ALMS was used to IPBPM and KMCG, two pixel-by-pixel multiplications of images.

Ansari et al. [9] first developed the 4-2 AC, which had significant inaccuracy, and then utilized this block to produce two $4 * 4$ multipliers, which were subsequently used to generate $16 * 16$, $32 * 32$ AM. Compared to previous APP booth multipliers with equivalent accuracy, the proposed 4-2 AC-based AM (CBAM) reduced PDP-MRED by 52% and PDP by 50%. It was then used in the IMS and JPEG applications, yielding more excellent quality outputs while using less power (PC).

Gillani et al. [10] developed an Internal-Self-Healing (ISH) system that enables a computational device to utilize self-healing without needing a pair. First, this article improves the overflow issue by developing the $2 * 2$ AM utilizing ISH. Finally, the

multiplier-accumulated unit (MAC) evolved and produced a 55.1% higher-quality output for precise, efficient designs.

A customizable error negligibly biased APP integer multiplier (BAIM) architecture was suggested by Saadat et al. [11]. This BAIM was improved by removing leading one detection (LOD) and barrel shifting logic and using an error reduction technique. The next step is to use the above reasoning to create a floating-point multiplier. Finally, the constructed Gaussian filter (GF) filters a 256×256 -pixel gray-scale standard using a 77% Gaussian kernel with $\alpha = 1.5$. Image of a drill.

Esposito et al. [12] presented 3-2, 4-2, 5-2, and 6-3 ACs after developing and analyzing several kinds of ACs. At the PP addition step, the proposed AC is utilized in a dadda multiplier (CDAM), and the proposed multiplier provides more power or speed than the previous AM. Finally, this recommended multiplier may be used for image smoothing (IS) using GF and adaptive filtering with least mean square (LMS).

Hammad et al. [13] employed AM to increase the VGG Net deep learning network's performance while decreasing the cost, power, area, and latency. CIFAR-10 and CIFAR-100 were used to assess the network's classification accuracy using the proposed AM employing deep learning analysis accuracy.

Chandran et al. [14] developed that rounding-based inexact AM and suggested AM offered outstanding efficiency and energy utilization than the present rounding-based multiplier (ROBA).

Mazahir et al. [15] used the probability mass function to develop the recursive type AM and accuracy analysis. They used the probabilistic model of the fundamental building block and the probability distributions of inputs to calculate the error probability in an arbitrary bit-width AM (PMF). Finally, the proposed AM may be utilized to combine pictures.

Zendegani et al. [16] developed three different types of rounding-based AM (RBAM), one for unsigned data and two for signed data. The proposed RBAM surpassed the present AM (EAM), and the proposed three AM were subsequently applied to the IS and IMS applications in terms of speed and energy consumption.

In comparison with DRUM and Wallace tree multiplier, Lohray et al. [17] developed rounded-based AM (ROBAM), which increased power, area, and speed (WTM). The likelihood of recurrence for rounded values and the input data rounding pattern have been presented as two essential elements to regulate the degree of precision for each range of data with little hardware expense.

Tung et al. [18] suggested two critical LP and HA 8×8 AM (AAM) characteristics. It was initially built based on distinct importance weights applied to various ACs to compile the terms of their product. Finally, it decreased the time the PC was saved with minor mistakes. Second, it utilized higher-order AC to accomplish LP and HA AM at many middleweights.

For image compression, Ramasamy et al. [19] suggested two architectures: HA hybrid segment APP multiplier (HSAM) and improved HSAM (EHSAM). Static and dynamic segment techniques may be used with the two designs presented. For different inputs, 8-bit HSAM and EHSAM offer 99.86% and 99.9% accuracy, respectively. Compared to the current 8-bit static segment technique, the suggested 8-bit HSAM and EHSAM enhanced speed by 40% and 85%, respectively (SSMAM).

Anil Kumar Reddy et al. [20] developed a radix-8 booth AM with a Kogge-stone adder and an APP full recoding adder. Finally, when compared to the radix-8 booth multiplier, the developed Radix-2 8-point DIT-FFT algorithm utilizing the suggested AM and proposed AM performance of FFT is improved by 21%.

Two APP booth encoders and two redundant-based (RB) 4-2 AC based on RB complete and half adders (HA) are suggested for the RB AM by Liu et al. [21]. (REAM). Two 4-2 AC and two adders are proposed (BM). Finally, the proposed AM employs the FIR filter and the KMCG algorithm.

Ko et al. [22] designed truncated and rounded AM, with a maximum absolute inaccuracy of 1 unit of least position guaranteed. Using rounded and truncated methods, the proposed AM reduces the number of full adders (FA) and HA during the tree reduction stage, saving 31% of the area compared to EAM.

Narayanamoorthy et al. [23] developed an AM for DSP applications that trades off computational accuracy with energy consumption (EC) at design time and achieves a 58.1% reduction in EC over current AM.

Hashemi et al. [24] presented A Dynamic Range Unbiased APP Multiplier (DRUAM) for APP applications. The developed DRUAM multiplier architecture is also scalable, allowing designers to tweak it to their desired accuracy, power output, and propagation delay reduction. The developed DRUAM is compatible with both the GF and the image compression algorithms.

TRSAM (truncated and rounded-based scalable AM) was suggested by Vahdat et al. [25], which decreased the number of PP by truncating each input based on the leading one-bit position input. Compared to EM, utilizing shift and adding a short fixed-width AM operation resulted in a substantial improvement in EC. Finally, the TRSAM was implemented in the IMS application.

A majority logic-based (MLB) AM and approximate adder (AA) was developed by Liu et al. [26]. ACs and reduction circuits make up the planned MLBAM. The suggested architecture used the MLBAM two IPBPM programmed to analyze hardware metrics such as latency, gate complexity, and error metrics.

Nandan et al. [27] developed an efficient logarithm multiplier (ELM) that uses improved operand decomposition (IOD) and a pipeline method to minimize error while utilizing the least amount of hardware and time. Finally, they used ELM to build a finite impulse response (FIR) filter and compared performance to prior findings.

A parallel integer (PI) AM generator for FPGA was introduced by Kakacak et al. [28], and PIAM combines five stages. For column compression, the initial step is to create a new Generalized Parallel Counter (GPC) grouping method. Generate the PP produces placement pragmas in the second stage. To make use of the FPGA's inherent carry chains, the fourth step employs a ternary adder as a final adder. The fifth stage uses a new GPC-based row compression technique to decrease the final adder's width.

For producing the PP that encodes MSBs with correct radix-4 encoding and LSBs with the appropriate higher-order radix, Leon et al. [29] developed signed hybrid high radix encoding AM (HHREAM). Compared to an identical radix-4 multiplier, the proposed HHREAM saved 56.2% energy and 30% area. Finally, the suggested AM

and the SF, FIR filter, and matrix multiplication techniques created the edge detection picture.

Vahdat et al. [30] created a truncation-based AM with minimal energy (LETAM). This article proposes a customizable output quality multiplier that may alter the output quality during multiplication. Compared to EM, the suggested LETAM has an average improvement of 75.9% area and 90% energy. We used 32-bit LETAM in the DCT function of the JPEG encoder application for eight pictures, as proposed in DSP applications.

Akbari et al. [31] developed four 4-2 ACs that easily transition between distinct and APP working modes. The planned four 4-2 AC provides HP, low PC, and poorer accuracy. This 32-bit dadda multiplier (42DAM) created by 4-2 AC was eventually utilized in IMS and IS applications.

Lau et al. [32] examined the concept of energy assignment to probabilistic AM. This work began by deriving specific analytical findings for array multipliers and basic multiplier types and then built these multipliers using an energy assignment method. The proposed AM was then applied to the IMS application.

Venkatachalam et al. [33] created two AM for power conservation (PAM). PAM1 is utilized in all PP columns of an m -bit multiplier, while PAM2 is used in them-1 least significant columns. In comparison with EM, AM saves 72 and 38% of the power and lowers the mean relative error by 7 and 0.02% after synthesizing them. Finally, the geometric mean filter was used to apply two suggested AMs to IP applications.

Liu et al. [34] designed that AM has a lower PC and a shorter critical path than EM, making it ideal for high-performance DSP applications. Compared to WTM, the designed AM saved 20% delay and % power. The designed AM provides a low mean error distance. For rapid PP buildup, AA may restrict its carry propagation to the closest neighbors using this technique.

Momeni et al. [35] proposed four AM after designing two 4-2 AC (4-2 AC1, 4-2 AC2). All 4-2 AC was replaced with 4-2 AC1 in the initial AM design. The second AM comprises 6 HA, 1 FA, and 17 AC, with all four 4-2 AC changed with 4-2 AC2. The third AM is utilized for the 4-2 AC1 in the $m - 1$ least significant column. The 4-2 AC2 in the $m - 1$ least significant column was the subject of the fourth AM. Finally, these four AMs were tested using IPBPM, and two of them gave positive findings.

Bhardwaj et al. [36] created a Wallace Tree AM with a bit of area (WTAWP). The proposed WTAM employs the carry-in prediction technique to minimize the critical route. Compared to WTM, the suggested WTAM decreased the area, power, and latency. Finally, WTAM was applied to DCT and IDCT for IP applications, resulting in a 30% area and 30% power reduction with no loss of picture quality.

Kulkarni et al. [37] created a 2×2 AM with changeable error characteristics and used it to create a recursive type 16-bit AM. The suggested design saved 45.5% of electricity and had a 3.31% inaccuracy compared to EM. The 16-bit AM was then applied to the IMS application.

Error-Tolerant AM (ETAM) was suggested by Kyaw et al. [38]. ETAM was created with inputs divided into two sections: a multiplication portion containing higher-order bits and a non-multiplication part that includes lower-order bits. The lengths

of both portions may not be identical, and multiplication starts when the bits divide and travel in opposing directions concurrently until all bits are taken. Compared to EM, 16-bit ETAM saved 45.1% power and 36.7% area.

Lin et al. [39] were the first to use this suggested 4-bit Wallace tree AM to build a 4-2 counter (WTAM). Compared to the precise WTM, the suggested design saved 10.75% of electricity and added 9.9% to the delay. In addition, this study created an error correction method and used it with the suggested WTAM to reduce error.

The proposed AM operated accurate mode or APP mode, decreased decode latency and dynamic PC, and ultimately used 16-bit proposed AM to discrete cosine transform (DCT) and compared the performance of various accuracy modes, according to Ashutosh Mehta et al. [40].

Pranay et al. [41] developed a modified booth AM (MBAM) that uses AA to detect errors in real time. First, two AA and MBAAM1 and MBAAM2 were created in this study. MBAAM1 obtained a 15.2% delay and a 5.1% power savings over BM, whereas MBAAM2 achieved a 15.8% delay and a 2.1% power savings over BM. MBAAM1 and MBAAM2 were then used for IP and IoT applications.

Liu et al. [42] created a radix-4 2 booth encoder AM (BEAM). They compared it to an inexact bit width of other booth AM for error reduction and comparability utilizing key error factors. Finally, IPBPM was applied to these two BEAMs.

Kim et al. [43] suggested an EE AM (EEAM) with appropriate error characteristics that uses lesser energy AA. Finally, they created a 16-bit EEAM comparison with a binary multiplier, which produced more detailed findings.

Pabithra et al. [44] created four 5-3 AC and built 15-4 AC utilizing the four 5-3 AC they suggested. This article proposed 16-bit AM by combining four 15-4 ACs and then applying it to an IP application.

Jiang et al. [45] proposed two-LP and reduced critical path delay AM (PDAM) for high-performance DSP applications. Using either OR gates or the suggested AA, this article adds the PP and its carries independently to improve speed and minimize mistakes. Finally, two PDAMs for IMS and IS applications were proposed.

Venkatachalam et al. [46] created a three-booth encoder AM system (BEAM1, BEAM2, and BEAM3). Each of the three designs has hardware-saving features for PP creation and adding. In comparison with EM, BEAM1 achieved a 23.1% area reduction and a 15.2% power decrease. In comparison with EM, BEAM2 achieved a 51.2% area reduction and a 46.1% power decrease. In comparison with EM, BEAM3 achieved a 56.1% area reduction and a 46% power reduction. Finally, image matrix multiplication and FIR filter applications are implemented using BEAM1, BEAM2, and BEAM3.

Using the novel PP perforation method, Zervakis et al. [47] created a hardware-level AM (HLAM) (PPPT). Critical mistakes in PPPT are limited and predictable and are only dependent on the input supply. In comparison with EAM, HLAM decreased PC by 50.2%, area by 45.3%, and critical latency by 35.2%. Finally, HLAM was used to identify edges and calculate KMCG.

High-speed AM was suggested by Jagadeeswara Rao et al. [48]. At the reduction stage, an HP AA with 4-2 and 8-2 AC was utilized, which improves speed by 25.1% above the stated WTM.

Jagadeeswara Rao et al. [49] have developed a Russian peasant AM with high-speed AAs, with an 8-2 adder AC. Compared to the conventional RP multiplier, it also lowers the latency by 25.2%.

Modified Static Segment AM was developed by Jothin et al. [50] (MSSM). Using a Significance Estimator Logic Circuit enhanced the accuracy by eliminating lower-order important information of input operands. The proposed 16-bit MSSM saves 83.4% LUTs, 38.7% power, 24.5% latency, and 0.7% worse computational accuracy than the current dynamic significant AM.

Garg et al. [51] present a new Leading one-bit Based AM (LoBAM) architecture that chooses k -bits from n -bit inputs ($k < n/4$) using leading one bit (LOB) and then computes approximation product using these tiny inputs. The precision is enhanced even further by choosing the next k -bits depending on the LOB location and calculating the final product using the partial product.

Rounding-based AM (RBAM1) and reconfigurable rounding-based AM are two energy-efficient designs developed by Garg et al. [52]. (RRBAM1). Compared to the current AM, the proposed 8-bit RBAM10 needs 59.9% (54.7%) less space (delay). Compared to the filter with existing AM, the Gaussian filter built using RBAM10 uses 32.5% less energy.

Strollo et al. [53] published the first thorough review and comparison of 4-2 AC previously suggested in the literature. Then, compared to twelve existing 4-2 AC, they created three error-efficient 4-2 ACs. They evaluated the findings after analyzing the performance comparison and developing two new AMs. Finally, they find that proposed AMs with proposed 4-2 ACs outperform suggested AMs with existing 4-2 ACs in terms of performance.

A few recent AMs were compared by Goswami et al. [54]. Based on these comparisons, data route complexity reduction AM1 is the best design in terms of PC and area, with 58% less power and 61% less space than the broken array multiplier.

Recent 45 AMs and their applications are reviewed by Jagadeeswara Rao et al. [55]. Finally, they conclude that no recent AM does not meet both the error and design criteria.

3 Results Analysis

We analyze all of the most recent and popular designs from 2015 to 2021 to fully comprehend and analyze the results of a systematic literature review. These are the conclusions reached by recent research articles. The findings of the most current designs are shown in Table 1. Synopsys Design Compiler generated a 28-nm CMOS cell library for CBAM and AM1 (SDC). The Cadence RTL compiler created LoBAM and RAAM1 utilizing a 65-nm PDK file. The SDC tool was used to develop all current designs, while AAM was based on simulation results using the SAED 32-nm CMOS library, CDAM simulates and synthesizes using standard 40-nm CMOS technology, and the other suggested architectures were built using the 45-nm CMOS Standard library.

Table 1 Design parameters comparison of various AMs

AM name	Word size	Technology (nm)	Area (μm^2)	Power (μw)	Delay (ns)
ALMS [8]	8	45	331.44	70.1	0.68
CBAM [9]	8	28	284.32	69.678	1.63
AAM [18]	8	32	1127.08	261.53	0.85
CDAM [12]	8	40	428	0.72	0.419
ERAM [6]	16	45	2048.32	75.84	5.02
ROBAM [17]	16	45	2618.22	103	2.17
REAM [21]	16	45	1338	322.1	0.84
DRUAM [24]	16	45	5700	525	1.31
TRSAM [25]	16	45	473	32	0.95
HHREAM [29]	16	45	4153	499.8	0.75
LETAM [30]	16	45	869	80	1.45
42DAM [31]	16	45	326	154	0.48
PAM1 [33]	16	45	2158.56	503.15	0.47
PAM2 [33]	16	45	3319.20	1102.03	0.66
BEAM [42]	16	45	2066	479.9	0.94
RBAM [16]	16	45	3224	537	1.12
MSSM [50]	16	45	–	30	17.11
LoBAM [51]	16	65	3792	190.4	1.69
RBAM1 [52]	16	65	4106	3113	0.92
AM1 [53]	16	28	727	663	0318

Present Authors developed AMs with a word size of 8 and 16 are shown in Table 1. Compared to AAM, ALMS, and CDAM, CBAM reduced the area by 75%, 14.5%, and 33.5% in 8-bit AM.

When compared to AAM, ALMS, and CBAM, CDAM lowered power by 99.9%, 99%, and 98.7%, respectively. Compared to AAM, ALMS, and CBAM, CDAM decreased delays by 20%, 42.7%, and 74.3%, respectively.

Compared to DRUAM in 16-bit AM, 42DAM reduced the area by 94.3%. In 16-bit, 42DAM took up less space, whereas DRUAM took up more. DRUAM consumes more power than the remaining AM and is replaced by the lower-power TRSAM, resulting in a significant delay in ERAM and a more negligible delay in PAM1. Performance comparison of various AMs is illustrated in Fig. 1.

Compared to the other AMs, AM1 offered less delay, as seen in Table 1. It is also that detected MSSM less PC compared to remaining AMs.

Different AMs have been created in recent years. However, few authors have concentrated on increasing AM accuracy to enhance DSP application performance. In terms of error metrics, Table 2 compares the accuracy of several recent AMs. The formulae and definitions for mean error distance (MED), normalized MED (NMED), and normalized error distance (NED) may be found in [56]. It describes the mean

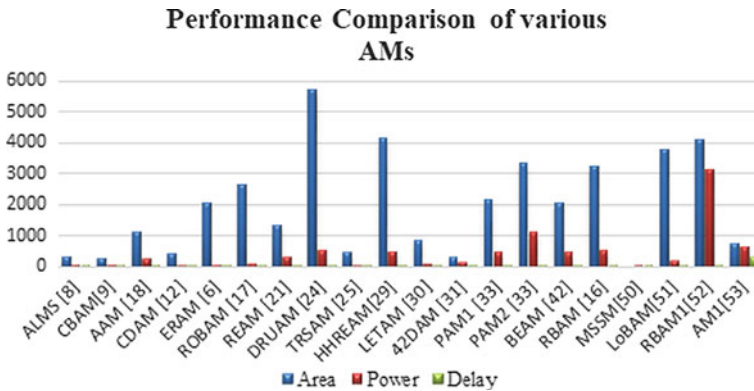


Fig. 1 Performance comparison of various AMs

Table 2 Error parameters of various AMs

AM name	Word size	MED	NMED	MRED	NED	MRE
ERAM [6]	8	28.05	–	–	–	–
ALMS [8]	8	–	9.25×10^{-3}	–	–	–
CBAM [9]	8	–	–	0.014	0.18	–
CDAM [12]	8	51.0	–	–	–	–
AAM [18]	8	–	55	0.09	–	–
REAM [21]	16	–	0.17×10^{-7}	0.337	–	–
DRUAM [24]	16	$6.3E+16$	0.0034	–	–	–
HHREAM [29]	16	–	–	0.28	–	–
42DAM [31]	16	$2.11E+08$	–	0.3873	0.0492	–
PAM1 [33]	16	–	–	–	1.78×10^{-2}	7.63×10^{-2}
PAM2 [33]	16	–	–	–	7.10×10^{-6}	2.44×10^{-4}
BEAM [42]	16	–	3.10×10^{-5}	1.76×10^2	–	–
RBAM [16]	16	$1.3E+17$	0.0069	–	–	292
LoBAM [51]	16	2.9×10^7	–	–	0.114	3.2×10^{-4}
RBAM1 [52]	16	2.9×10^7	–	–	0.114	3.2×10^{-4}
AM1 [53]	16	–	2.61×10^{-6}	4.80×10^{-5}	–	–

relative error distance (MRED) and the concept of mean relative error [57]. ALMS gives less value of NMED, and AAM provides an enormous value of NMED and reduces the error of 54.6% of PAM2.

4 Image Processing Applications

In AMs, there are numerous real-time applications, but we will concentrate on two of them. ACG is a signal processing and image processing technique. The IMS and IS techniques and the various common masks used to smooth and sharpen the picture are covered in this section.

Table 3 uses the most recent AMs to analyze the performance of IMS with the performance metrics such as peak to signal noise ratio (PSNR) and structural similarity index matrix (SSIM). We first created the AMs MATLAB code and then applied it to the baseline benchmark picture of Leena with various masks. This ERAM has higher PSNR values and lowers SSIM in 42DAM than others. Table 3 reveals that DRUM, RBAM, and AM1 all have the identical SSIM value of 1.0. In addition, AM1 has a high PSNR value compared to existing AMs. IS is the second application, which employs the most recent AMs with various masks. Table 4 shows the PSNR and SSIM values for the IS application after developing the AM in MATLAB and the standard convolution of the primary test image with various standard masks. Compared to the other two RBAMs, this one has a higher PSNR value.

Table 3 Performance parameters comparison of IMS using existing AMs

AM name	Word size	PSNR	SSIM
ERAM [6]	8	48.3	–
CBAM [9]	16	26.7	0.9
DRUAM [24]	16	37.9	1.0
TRSAM [25]	16	30.9	0.9
LETAM [30]	16	32.3	0.9
42DAM [31]	16	–	0.6
RBAM [19]	16	43.5	1.0
LoBAM [51]	16	28.2	0.7
RBAM1 [52]	16	30.3	0.8
AM1 [53]	16	∞	1.0

Table 4 Performance parameters comparison of IS embedded with existing AMs

AM name	Word size	PSNR	SSIM
DRUAM [24]	16	34.7	0.9
42DAM [31]	16	–	0.4
RBAM [19]	32	44.5	1
AM [53]	16	55.3	1

5 Conclusion

We covered systematic progress in AMs and their applications in this paper. The methods employed by researchers to enhance the design of AMs are detailed in this thorough paper. The AM1 is, nevertheless, the optimal design option for accuracy concerns, according to this study. However, the CDAM and TRSAM are the most excellent options when it comes to power and delay. CBAM and 42DAM are good options if the location is the main issue. This article also discusses the author's IMS and IS applications, with the IMS application providing superior AM1 and RBAM values. In an IS application, RBAM is the superior option. This study is anticipated to lead to significant advancements in image processing applications.

References

1. Mittal S (2016) A survey of techniques for approximate computing. *ACM Comput Surv* 48(4):1–4
2. Sampson A et al (2011) EnerJ: approximate data types for safe and general low-power computation. In: *Proceeding on 32nd ACM SIGPLAN conference on programming language design and implementation*, California, pp 164–174
3. Yoshida K, Sakamoto T, Hase T (1998) A 3D graphics library for 32-bit microprocessors for embedded systems. *IEEE Trans Consumer Electron* 44(3):1107–1114
4. Klinefelter JR, Tschanz J, Calhoun BH (2015) Error-energy analysis of hardware logarithmic approximation methods for low power applications. In: *Proceeding on IEEE international symposium on circuits and systems (ISCAS)*, Lisbon, Portugal, pp 2361–2364
5. Jiang H, Liu C, Maheshwari N, Lombardi F, Han J (2016) A comparative evaluation of approximate multipliers. In: *Proceeding on IEEE/ACM international symposium on nanoscale architectures (NANOARCH)*, Beijing, pp 191–196
6. Ha M, Lee S (2018) Multipliers with approximate 4-2 compressors and error recovery modules. *IEEE Embedded Syst Lett* 10(1):6–9
7. Alouani I, Ahangari H, Ozturk O, Niar S (2018) A novel heterogeneous approximate multiplier for low power and high performance. *IEEE Embedded Syst Lett* 10(2):45–48
8. Liu W, Montuschi P (2018) Design and evaluation of approximate logarithmic multipliers for low power error-tolerant applications. *IEEE Trans Circuits Syst I* 65(10):2856–2868
9. Ansari MS, Cockburn BF (2018) Low-power approximate multipliers using encoded partial products and approximate compressors. *IEEE J Emer Sel Topics Circuits Syst* 8(3):2856–2868
10. Gillani GA, Shafique M (2019) MACISH: designing approximate MAC accelerators with internal-self-healing. *IEEE Access* 7:2169–3536
11. Saadat H, Bokhari H, Parameswaran S (2018) Minimally biased multipliers for approximate integer and floating-point multiplication. *IEEE Trans Comput Aided Des Integr Circuits Syst* 37(11):2856–2868
12. Esposito D, Napoli E, Petra N (2018) Approximate multipliers based on new approximate compressors. *IEEE Trans Circuits Syst I* 65(12):2856–2868
13. Hammad I, El-Sankary K (2018) Impact of approximate multipliers on VGG deep learning network. *IEEE Access* 6:60438–60444
14. Chandran V, Elakkiya (2017) Energy efficient and high-speed approximate multiplier using rounding technique. *J VLSI Des Signal Process* 3(2):1–8
15. Mazahir S, Hasan O, Hafiz R, Shafique M (2017) Probabilistic error analysis of approximate recursive multipliers. *IEEE Trans Comput* 66(1):1982–1990

16. Zendegani R, Kamal M, Bahadori M, Afzali-Kusha A, Pedram M (2017) RoBa multiplier: a rounding-based approximate multiplier for high-speed yet energy-efficient digital signal processing. *IEEE Trans Very Large Scale Integr (VLSI) Syst* 25(2):393–401
17. Lohray P, Gali S, Rangiseti S, Nikoubin T (2019) Rounding technique analysis for power-area & energy efficient approximate multiplier design. In: *Proceeding on IEEE 9th annual computing and communication workshop and conference (CCWC)*, Nevada, pp 420–425
18. Tung C-W, Huang S-H (2019) Low-power high-accuracy approximate multiplier using approximate high-order compressors. In: *Proceeding on 2nd international conference on communication engineering and technology*, Nagoya, pp 163–167
19. Ramasamy J, Nagarajan S (2016) Hybrid segment approximate multiplication for image processing applications. *Circuits Syst* 7:1701–1708
20. Anil Kumar Reddy Y, Sathish Kumar P (2018) Performance analysis of 8-point FFT using approximate radix-8 booth multiplier. In: *Proceedings on international conference on communication and electronics systems (ICCES 2018)*, Coimbatore, pp 42–45
21. Liu W, Cao T, Yin P, Zhu Y, Wang C, Swartzlander EE (2018) Design and analysis of approximate redundant binary multipliers. *IEEE Trans Comput* 68(6):804–819
22. Ko H-J, Hsiao S-F (2011) Design and application of faithfully rounded and truncated multipliers with combined deletion, reduction, truncation, and rounding. *IEEE Trans Circuits Syst II* 58(5):304–308
23. Narayanamoorthy S, Moghaddam HA, Liu Z, Park T, Kim NS (2015) Energy-efficient approximate multiplication for digital signal processing and classification applications. *IEEE Trans Very Large Scale Integr (VLSI) Syst* 23(6):1180–1184
24. Hashemi S, Bahar RI, Reda S (2015) DRUM: a dynamic range unbiased multiplier for approximate applications. In: *Proceeding on IEEE/ACM international conference computing-aided design (ICCAD)*, Austin, pp 418–425
25. Vahdat S, Kamal M, Afzali-Kusha A, Pedram M (2019) TOSAM: an energy-efficient truncation and rounding-based scalable approximate multiplier. *IEEE Trans Very Large Scale Integr Syst* 27(5):1161–1173
26. Liu W, Zhang T, McLarnon E, O'Neill M, Montuschi P, Lombardi F (2019) Design and analysis of majority logic based approximate adders and multipliers. *IEEE Trans Emer Topics Comput* 1–69
27. Nandan D, Kanungo J, Mahajan A (2017) An efficient VLSI architecture design for logarithmic multiplication by using the improved operand decomposition. *Integr Elsewhere* 58:134–141
28. Kakacak A, Guzel AE, Cihangir O, Gören S, Fatih Ugurdag H (2017) Fast multiplier generator for FPGAs with LUT based partial product generation and column/row compression. *Integr Elsewhere* 57:147–157
29. Leon V, Zervakis G, Soudris D, Pekmestzi K (2018) Approximate hybrid high radix encoding for energy-efficient inexact multipliers. *IEEE Trans Very Large Scale Integr Syst* 26(3):421–430
30. Vahdat S, Kamal M, Afzali-Kusha A, Pedram M (2017) LETAM: a low energy truncation-based approximate multiplier. *Comput Electric Eng* 63:1–17
31. Akbari O, Kamal M, Afzali-Kusha A, Pedram M (2017) Dual-quality 4:2 compressors for utilizing in dynamic accuracy configurable multipliers. *IEEE Trans Very Large Scale Integr (VLSI) Syst* 25(4):1352–1361
32. Mark SKL, Ling K-V, Chu Y-C (2009) Energy-aware probabilistic multiplier: design and analysis. In: *Proceedings on International conference on compilers, architecture, and synthesis for embedded systems*, New York, pp 281–290
33. Venkatachalam S, Ko S-B (2017) Design of power and area efficient approximate multipliers. *IEEE Trans Very Large Scale Integr Syst* 25(5):1782–1786
34. Liu C, Han J, Lombardi F (2014) A low-power, high-performance approximate multiplier with configurable partial error recovery. In: *Proceeding on 2014 design, automation & test in Europe conference & exhibition (DATE)*, Dresden, pp 1–4
35. Momeni A, Han J, Montuschi P, Lombardi F (2015) Design and analysis of approximate compressors for multiplication. *IEEE Trans Comput* 64(4):984–994

36. Bhardwaj K, Mane PS, Henkel J (2014) Power- and area-efficient approximate wallace tree multiplier for error-resilient systems. In: Proceeding on fifteenth international symposium on quality electronic design, Santa Clara, pp 263–271
37. Kulkarni P, Gupta P, Ercegovic M (2011) Trading accuracy for power with an under designed multiplier architecture. In: Proceeding on 24th annual conference on VLSI design, Chennai, pp 346–351
38. Kyaw KY, Goh WL, Yeo KS (2011) Low-power high-speed multiplier for error tolerant application. In: Proceeding on IEEE international conference of electron devices and solid-state circuits, Hong Kong, pp 1–4
39. Lin C-H, Lin I-C (2013) High accuracy approximate multiplier with error correction. In: Proceeding on 2013 IEEE 31st international conference on computer design (ICCD), Asheville, pp 1–6
40. Mehta A, Maurya S, Sharief N, Pranay BM, Jandhyala S, Purini S (2015) Accuracy-configurable approximate multiplier with error detection and correction. In: Proceeding on TENCON 2015—2015 IEEE region 10 conference, Macao, pp 1–4
41. Pranay BM, Jandhyala S (2015) Accuracy configurable modified booth multiplier using approximate adders. In: Proceeding on IEEE international symposium on nano-electronic and information systems, Indore, pp 1–5
42. Liu W, Qian L, Wang C, Jiang H, Han J, Lombardi F (2017) Design of approximate radix-4 booth multipliers for error-tolerant computing. *IEEE Trans Comput* 66(8):1435–1441
43. Kim S, Kim Y (2017) High-performance and energy-efficient approximate multiplier for error-tolerant applications. In: Proceeding on 2017 international SOC design conference (ISOC), Seoul, South Korea, pp 278–279
44. Pabithra S, Nageswari S (2018) Analysis of approximate multiplier using 15-4 compressor for error tolerant application. In: Proceeding on 2017 international conference on control, power, communication and computing technology (ICCPCT), China, pp 410–415
45. Jiang H, Liu C, Lombardi F, Han J (2019) Low-power approximate unsigned multipliers with configurable error recovery. *IEEE Trans Very Large Scale Integr Syst* 66(1):189–202
46. Venkatachalam S, Adams E, Lee HJ, Ko S-B (2019) Design and analysis of area and power efficient approximate booth multipliers. *IEEE Trans Comput* 68(11):1697–1703
47. Zervakis G, Tsoumanis K, Xydis S, Soudris D, Pekmestzi K (2016) Design-efficient approximate multiplication circuits through partial product perforation. *IEEE Trans Very Large Scale Integr Syst* 24(10):3105–3117
48. Jagadeeswara Rao E, Jayram Kumar K, Prasad TV (2018) Design of high-speed wallace tree multiplier using 8-2 and 4-2 adder compressor. *Int J Eng Technol* 2386–2390
49. Jagadeeswara Rao E, Rama Vasantha A (2018) Design and implementation of high speed modified Russian peasant multiplier using 8-2 adder compressors. *Int J Res Electron Commun Eng* 6:379–383
50. Jothin R, Vasanthanayaki C (2018) High performance modified static segment approximate multiplier based on significance probability. *J Electron Test* 34:607–614
51. Garg B, Patel SK, Dutt S (2020) LoBA: a leading one bit based imprecise multiplier for efficient image processing. *J Electron Test* 36:429–437
52. Garg B, Patel S (2021) Reconfigurable rounding based approximate multiplier for energy efficient multimedia applications. *Wirel Pers Commun* 118:919–931
53. Strollo AGM, Napoli E, De Caro D, Petra N, Meo GD (2020) Comparison and extension of approximate 4–2 compressors for low-power approximate multipliers. *IEEE Trans Circuits Syst I Regul Pap* 67(9):3021–3034
54. Goswami SSP, Paul B, Dutt S, Trivedi G (2020) Comparative review of approximate multipliers. In: Proceedings of 30th international conference radioelektronika (RADIOELEKTRONIKA), Bratislava, Slovakia, pp 1–6
55. Jagadeeswara Rao E, Samundiswary P (2021) A review of approximate multipliers and its applications. In: Komanapalli VLN, Sivakumaran N, Hampannavar S (eds) *Advances in automation, signal processing, instrumentation, and control. Lecture notes in electrical engineering*, vol 700

56. Liang J, Han J, Lombardi F (2013) New metrics for the reliability of approximate and probabilistic adders. *IEEE Trans Comput* 62(9):1760–1771
57. Akbari O, Kamal M, Afzali-Kusha A, Pedram M (2018) RAP-CLA: a reconfigurable approximate carry look-ahead adder. *IEEE Trans Circuits Syst II Express* 65(8)

Artificial Intelligence Technological Revolution in Education and Space for Next Generation



S. L. Chetradavee, K. Anushka Xavier, and N. Jayapandian

Abstract The goal of this research is to discover the various potential for the educational system using artificial intelligence (AI). The world today is dealing with AI in different sectors. This study specifically looked into the prospects for acquiring efficient and high-quality education for each student, automating administrative tasks, including regulating adaptive student support systems. AI has been leveraged and used in the education sector in various formations. AI initially took in the form of computers with the cognitive model, transformed to online learning, together with other technologies, the use of AI provides chatbots to perform instructors. Imagine you can access your classroom from anywhere at any time through an online learning system. These functionalities enable the education system to deal with the curriculum effectively. Using these facilities, teachers instruct the students to desire to achieve their goals efficiently. The primary aim of this article addresses the concepts in AI that serve to regulate and improve the overall quality of academic performance. The secondary aim of this article is to discuss AI involvement in the space domain.

Keywords Artificial intelligence · Education · Machine learning · Natural language processing · Students · Space

1 Introduction

AI is the study of any intellectual tasks incorporated into machines that typically require human intelligence. The main contribution of AI is to give the perception to create machines that have the ability of thinking like humans and take measures consequently. This is pushing the boundaries of enabling the creation of new systematic approaches in any field globally. The abundance of machine learning has to be preferably applied in the field of education to provide insights on fast pace learning, other benefits for the teachers, and admin processes that make them more effective and efficient. AI implementation can make globally connected classrooms available

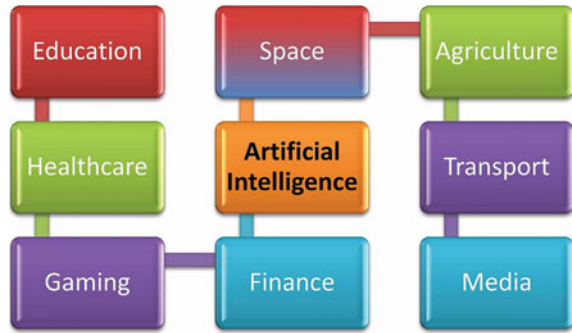
S. L. Chetradavee · K. Anushka Xavier · N. Jayapandian (✉)
Department of Computer Science and Engineering, CHRIST (Deemed to Be University), Kengeri Campus, Bangalore, India
e-mail: jayapandian.n@christuniversity.in

for all students from different countries [1]. This may be more helpful for students who have hearing or visual impairments. This will also be helpful to conduct online tutoring classes where each concept is demonstrated differently and can offer additional support to the student. The major advantage of online coaching is that undergraduates can do their course work at their claimed time and pace [2]. More humans are involved in the department of education where one teacher should handle 30–60 students in a classroom and they also have to complete other admin processes. And also, exams play a vital role in education where student capabilities are graded. Every year, students have 6 exams per semester, which means up to 12 exams per year. This requires plenty of groundwork including preparing the question papers, printing those, and sending the question papers to the respective classrooms in an organized manner. This needs a lot of labour, time, and money. We live in a period where innovation rules the world extraordinarily. The ability of AI can automate tedious tasks, including administrative work, grading homework, preparing question papers online, and evaluating test papers which makes teachers spend more time competently on the individual student and assign personalized work for individuals [3]. The impact of the pandemic in the last couple of years is the proven evidence of how AI actively utilization increased its growth in education. The survey of McKinsey Global predominantly features the importance of AI development for organizations due to the COVID-19 crisis in the survey of ‘The State of AI in 2020’. And also, AI applications have gotten a parcel of recognition in the recent couple of years. The venture is to examine how different collaborators in AI identify vital concepts that are collectively promoted and propose their incentives for doing so. And also informative with the following questions: How much educational quality did the students achieve during the COVID19 crisis? What AI impacted the education field? How has AI improved to meet quality education and student retention? COVID-19 has changed our world. Most of the lectures and other classes, test-taking too went online during the crisis [4]. A 2020 Educause survey found that more than 54% of educational institutions used online platforms or AI remote surveillance services to ensure an unbiased test environment for students.

2 Nature of Artificial Intelligence

The evolution of AI is very interesting and becoming interminable in advanced technology. This is a visually striking performance in upcoming years, raising around \$126 billion by 2025. In various domains, the utilization of AI has been massively profitable. As per the Global AI survey in 2019 substantiates in such a manner 25% increased utilization of AI each year in business areas and 44% say AI benefits in cost reduction. AI can significantly become more obtainable in many organizations [5]. Not only is it effective in reducing costs, but it can also help employees solve and automate common and complex problems. Almost the globe is embracing AI in a variety of probabilistic approaches and diversification scenarios. Noticeably AI enterprise is the preferable trend in many organizations. AI comprehensively

Fig. 1 Application of artificial intelligence



strengthens in many sectors such as Education, Healthcare, Agriculture, Finance, E-Commerce, Gaming, Transport, social media, and the automotive industry. We will see how few sectors involving AI technologies predominantly. Figure 1 is elaborating the application of AI in various sector [6].

AI healthcare is majorly used. The widespread adoption of AI in certain domains will be successfully leveraged in future research. Following AI in Healthcare 2020 Leadership Survey Report, the current state of AI in IDN is they are planning to deploy 33% in the future and proficient currently up to 29%. Furthermore, Academic Medical Centre plans to distribute 44% in upcoming years and currently advanced up to 39% with AI. The recent day space technology is also used in artificial intelligence. There are rapid breakthroughs in AI and Machine Learning in Space. Advanced technologies have been implemented in the development and optimization of complex processes. The distance measures in the space from one planet to another planet will be difficult to calculate. In February 2021, NASA has taken advantage of AI to examine the destination for the satellite before the land on Mars. In addition, NASA is equipped with instruments such as AI smart cameras to identify obstacles or other structures. Based on European Space Agency (ESA), AI helps in handling imaging data that are transmitted by satellites to Earth. Over 150 terabytes of vast data per day are generated. On the other hand, AI will optimally reduce the mission cost, do a detailed observation on huge data, and produce accurate results. Most children and adults play games that help reduce stress, improve problem-solving skills, improve concentration, multitask, and make quick decisions. AI is utilized in gaming to give richness to the environment and predict intelligent behaviours preliminary in NPCs. Beyond NPCs, Researchers also used generative adversarial networks (GAN) to generate new levels in gaming [7]. GAN learned how to create a clone of PacMan (an action game developed in 1980) released by Nvidia in 2020. In addition, GAN gained the ability to regenerate game levels by watching 50,000 games from start to finish.

The emerging of AI has taken agriculture to the topmost levels. AI has been used to forecast the weathers which helps farmers to identify the climatic changes. Based on this analysis, they can cultivate, irrigate, harvest, and store the crops. As mentioned in Indian express dated August 31, 2021, and titled AI for the farmer, Indian food

tech start-ups upraised more than \$1 billion in the years 2019–2020. Following that growth of AI in agriculture is predicted globally to reach \$8.38 billion by 2030, nearly 25% widening growth when compared with 2019 (this reached a calculation of \$852.2 million). As mentioned in the introduction, AI helps the education system to be more interactive. AI continues to develop and new ways of application will be emerged in various sectors. As many are interested and excited to invest in AI in upcoming years. This explains how education and AI correspond to real-world opportunities. Following Times of India survey dated April 13, 2021, and titled as over 75% children had challenges to access education digitally: Survey reported the primary mode of education is WhatsApp which is used by 75% of people. And also 38% of students and teachers interacted by phone calls. During the COVID19 crisis, students face this type of mobile learning barrier [8]. To overcome challenges faced by students, they are focusing on the learning spiral which leverages AI to deliver better solutions and utility to institutions, schools, and colleges. The quality of education is to make sure every student must have an opportunity to learn on regular basis, supportive environment from institutions, teachers, and family. The outcome of it will be the better knowledgeable person to establish himself/herself into the society with ease. This might possible to achieve with the help of AI where complex things are made easier to handle in tough situations. AI will give the flexibility in such a way to explore multiple concepts, do multitasking, automation for regular scrap works in the administrations, and online tutoring where the students need more learning other than regular classes. Cloud computing is also playing a major role in online education [9].

3 Technological Aspects of AI in Education

AI is the evolution of computers in different forms. As the field of study, AI aims to solve cognitive problems, continuously analyse students to increase the performance of individuals, and provide targeted training based on their interested domain. There will be the generation of huge data to be captured, analysed, and processed by the computer. This survey aims to discuss how techniques of AI are used to improve the education system. AI Education can be classified based on learning strategies used for students, automation tools used in the administration process, and intelligence support systems for virtual labs.

3.1 AI in Administration Automation Tools

AI uses machines with modern algorithms to analyse and process data from the real world. In simple terms, collection of datasets, analysis of data using algorithms, data cleaning to eliminate non-value data, providing model training for the machines based on the analysed data, testing the results, and deploying the successful

outcomes of previous steps in the real-world scenarios [10]. One of the major reduction processes in administration will be calculating the student grades. This may be deployed through the AI where the tools of AI help to grade the student performance not only based on exams but also the behavioural performance of each student. AI uses machine learning (ML) for grading student performance which combines human understanding and AI statistical approach. ML can learn from the previous grades of students to analyse the performance automatically [11]. Training the ML by providing a huge amount of data from the real world, construct the process using classification algorithms such as Random Forest classifier, Logistic Regression classifier, Linear Regression, Naïve Bays algorithm, and processed data can be further classified based on a neural network which results in better solutions. The student's performance can be recorded in various formats like documents, records, papers (photo scanned pictures), and other formats. Weka AI is the software that can be used to perform the analysis using different classification and clustering. Pre-processor in Weka import the data into the tool and do the pre-processing on it which gives us insights on several records, the relation between the records, and help us how to classify the students based on it. Now, the classification can be applied to the previous outcome of the pre-processing tool. In this paper, we will discuss Random Forest Classification (decision tree) based on previous and current semester grades. Random Forest Classification is based on the decision trees linked with various previous subsets. The algorithm performs calculations based on previously available grades and also current grades to provide the average results of the individual students. The advantage of using this algorithm is to perform effectively on both Regression and Classification tasks. The obtained result from this algorithm will be more accurate because this could handle a large set of student data. Another tool like Scikit-learn can be used to apply the Classification, Regression, and Clustering algorithms. This is the python programming and open-source software. One way to predict the student's grades is using artificial neural networks (ANN) [12]. ANN is based on the human neurons where billions of neurons are connected in the human brain. In such a way, ANN is designed with nodes (like billion cells interconnected in the human brain), then input (like dendrites), weight (like synapse), and output (like Axon). This is represented as a weighted directed graph where the hidden layers between the neuron input and neuron output can be classified using weights provided for each input. This way students' grades can be calculated not only based on exams but also on other hidden terms. The student grades can be classified based on the decision tree algorithm and neural networks to achieve more accurate results. The automation of grading systems could be more advantageous in time savings, ability to obtain accuracy on grades of an individual student using automated tools in AI. This may help teachers could actively participate in other admin works in which human decision preliminary necessity.

3.2 AI in Student Education Learning System

Adaptive learning is an approach to provide the students with customized learning achieved with the help of AI and different kinds of computer algorithms. The personalized learning approach is based on each student's interests and skills. When the students cannot understand the concepts, they need to go through them twice and more to get clear insights on the concepts. This adaptive learning provides the flexibility for the students to recall at any time from any location. This also encourages tutors on targeted topics and visualizing in a different perspective for better understanding. The real-time application for Adaptive learning is the Dreambox learning platform [13]. DreamBox is designed to implement the K-8 math program digitally which adapts to each student and the ultimate aim to provide a personalized learning experience. Rocketship public school is a charter school in California that uses the Dreambox platform for mathematical concepts. This helps the students with personalized material for building foundations. This adaptive learning increases the student's grade up to 5.5% for spending only 21 h on the Dreambox platform. Carlton Innovation School, a school from Massachusetts had gained the benefits of customized learning. This increased by 17% in math proficiency for students. The cognitive model and data analysis are used in the Dreambox which helps the agent to validate the student performance by taking past six-week data sets and validating using a statistical model. This feedback helps the student to track their learning, results to develop valuable self-monitoring skills.

Machine learning (ML) comes into the picture when we talk about the predictive modeling approach [14]. ML is the subset of AL. The machine learns from the environment data sets and predicts the future data sets based on the predictive approach. ML takes the input data sets (huge amount of data) from experience, applies the ML algorithm to train and learn from data, builds the logical model to predict the new data which the output results. ML involves solving complex problems in a short amount of time, finding the pattern to get useful particulars from the given data sets. Cross-validation is a technique in the ML used to validate and compare the performance. ML cross-validation tests the training data sets which involve the basics steps as follows reverse the dataset of a particular input to get the validated set, use the training dataset to provide the training to the model, and cross-check the performance of the trained model using the validated set. The validation test can be performed on either one or more datasets. This provides the accuracy of the input datasets based on predictions. As discussed, the adaptive learning plan prioritizes the student goals to increase personalized strength, skills, and positive attitude towards society with the support of AI predictive models.

3.3 AI in Intelligent Support System

The intelligent support with the help of AI assists the student to proficient in the language and also significant growth in academic performance. Natural Language Processing (NLP) gives the space to improve the writing skills of each student. NLP is the technology used by machines to analyse and interpret human languages. NLP is part of AL, human language, and computer science [15]. From the name itself, we can understand NLU which helps the machines to understand the natural language spoken by humans. NLU extracts the datasets in the form of keywords, relations, and semantic roles. Whereas, NLG is converting the computer language to the natural language. Mainly involves text realization of computerized data to get a natural representation of language. The real-world application of NLP is e-rater. NLP is an effective way to process the linguistic input and convert it to natural texts, sentences that are understandable by humans. E-rater uses the NLP as schema to analyse the student's essays and provide feedback. One best way to evaluate essay writing skills gives holistic grades based on the construction of sentences, and grammar. The results provide constructive feedback on the usage of words, grammatical construction, and sentences composition. NLP is applied in many fields such as laboratory, e-learning, spam detection of unwanted mails, personal assistance, speech recognition, and chatbot. Remind was the most downloaded chatbox in 2018 with 11+ million installs. Remind is the chatbox used in the education community to stay connected with school from anywhere. This keeps the files private and helps to translate them into more than 90 languages. This way student succeeds to communicate lively in the class community.

4 Technological Aspects of AI in Space

Space getting crowded and it is necessary to solve a more complex environment. In space, there are various orbits, planes, and many objects that move around these orbits and planes at different speeds. The AI system helps identify solutions in various aspects, such as identifying the potential target, making decisions, preventing security threats, defending space resources, mission planning, spacecraft maintenance, and many more. AI solutions provide an opportunity for enhancing space missions. This survey aims to discuss how techniques of AI are used to improve the space.

4.1 AI in Space Healthcare

Astronauts play an important role by assisting as commanders on the spacecraft. Astronauts are the trained person who will be deployed to space flight. Monitoring the health of astronauts provides important standards for managing the risks involved

in space exploration. In low Earth orbits, there are opportunities considering medical care in space. In such emergency situations, it is necessary to use the advanced development of AI in Space Healthcare. AI intelligent systems are already being applied to medical fields. The machine needs large datasets to execute various tasks with high accuracy. Weak AI is the type of AI which is used to carry out smart tasks for precision. Weak AI is also known as narrow AI stimulates human cognition and provides the benefits of automatic time-consuming tasks without human interaction. The limitation of weak AI is that it can only be applied only to certain intelligence tasks and not general intelligence tasks. The intelligence system is used to predict the probable health risks by utilizing the learning algorithms of AI. There are many challenges for the development of AI in Space Healthcare that is entirely different from AI in regular Healthcare. The usual quantities such as temperature, oxygen, and others are taken into consideration to determine the medical health risks of a person. In addition to these quantities, the challenge is to verify the variation of gravitational attraction when the astronauts are exposed to the outer space and to review medical history, stimulates including changes in performance. The datasets available to train Weak AI for space healthcare include classified data, unclassified data, In-flight data, environmental data of astronauts. These are the special type of datasets provided to train and analyse. This helps to classify the information of the medical history of astronauts and sometimes may not be appropriate where the datasets are not completely from space environment. The resultant data will be provided to the formal astronauts, commercial astronauts and space tourists. Medical risk model of a mission and space principles to be concerned while providing the space healthcare with propose driven intelligence system.

4.2 AI in Space Mission Planning and Operation

Designing a new system architecture with unstructured data is difficult to fully automate with AI. System design includes a creative and innovative process that must be designed with the help of humans. AI intelligent assistant makes it easier to mission planning and deployment [16]. AI specialized assistance helps to answer complex queries with reliable information which provides the knowledge for early design planning of new space missions. Daphne is the real-time example for intelligent assistant. This helps the system engineer to design the early phases of designs and makes jobs easier by providing the relevant information. AIKO, an Italian start-up company has developed MiRAGE, a software library used to facilitate operations for space missions. Machine learning algorithms that combine the natural language processing (NLP) and speech recognition to enable AI assistant [17]. Speech recognition is the method of converting the audio voices to text. The first step, the intelligent assistance receives the audio voices of the person questioning the intelligent device and transforms them into a signal. These signals transform analog to digital, yielding digital text. This speech recognizer work depends on the Hidden Markov Model (HMM). HMM is used to divide the signals within some microseconds and is only based on

a statistical model which does not change over time. The output of HMM is in the form of vectors and helps in decoding from speech to text. Neural network, another technology of AI can be used here to overcome the time taken to process speech using HMM. Neural network is the deep learning concept which helps to find the hidden states between the input and output data by allocating the weights through an iterative process. Neural networks reduce unwanted audio signals by recognizing unnecessary speech and preventing its processing. This reduces the time for computing the speech signal. The second step, the converted digital text data is being processed through NLP. NLP pipeline helps to analyse the sentences and provides the relevant information based on the analysis. This relevant information answered by AI assistant is used in early design systems which helps to plan for a space mission. In 2018, The German Aerospace Centre (DLR) has successfully launched the AI assistant. This AI assistant performs the daily tasks and is supported as assistance to its astronauts in International Space Station (ISS). Crew Interactive MOBILE companion (CIMON) is developed as a fully voice controlled system. CIMON can act as humans, such as seeing, hearing, speaking, understanding, and even flying. CIMON is replaced with its successor CIMON2. CIMON2 is now onboard the International Space station for 3 years. CIMON was returned after 14 months in ISS. Therefore, AI assistance helps to do daily activities and act as a security system in the absence of humans.

4.3 AI in Space Predictive Approaches

Predictive analytics provides information on what may occur in the future. This information helps to handle situation of failure of any parts in spacecraft or to predict the state of the spacecraft [18]. Livingstone 2(L2) is a satellite which uses an AI system to operate complex environment with minimal human supervision. This also detects the most likely failure and recommends operations to fix failure system. The successor of Livingstone 2 is Livingstone 3 (L3). L3 uses the advancement model approach to predict the state. There are three main components in L3 namely system model, constraint system, and candidate manager. The first component, system model which stores the current system design and is responsible for tracking the constraints that are valid at any given time. The second component, constraint system tracks the overall system based on constraints. This step gets the constraints from previous system model and finds the differences of uncertainty. These observations are propagated to candidate manager which generates the faults information of the particle. These uncertainties are classified based on the Naive Bayesian algorithm. Naive Bayesian algorithm is the type of classification algorithm in AI model [19]. This algorithm is used to track and predict the particles based on the distribution probability. The key benefit of this method is that it does not require the training set, and is faster to predict with the real-time data based on the probability. The utilization of data-driven approaches are being used in space where the flight time series are analysed. This analysis helps the space engineer to take corrective measures for planning and scheduling the spacecraft. For data-driven approaches, the training sets are used to

analyse. Inductive monitoring system (IMS) is the software which utilizes the past data and learns based on machine learning algorithms. IMS automatically analyse the anomalous behaviour for which clustering algorithms are used. The clustering algorithm is a machine learning model. This algorithm classifies the mission data based on the groups with feasible similarities in a group, and with less or no similarities in another group. This algorithm is purely unsupervised learning and not based on any labelled dataset. K-means is the type of clustering algorithm that is successfully tested with the space data from past missions. This clustering algorithm is used in the VLFD05 satellite and resultant data is used for statistical analytics. The resultant data classified to find the relation between the trend of the parameters and fault occurrence by inferring with association rules. Therefore, this way the training dataset is used to predict the fault occurrence that may occur in the future.

4.4 AI in Space Communications

As space data increases, the necessity of using AI techniques increases. Initially, the National Aeronautics and Space Administration used the human controlled radio systems for space communication. The National Aeronautics and Space Administration (NASA) is the independent federal government of the US responsible for the space program and space exploration. As space data increases, NASA changed the human controlled radio system into AI-based cognitive radio systems. This network is more reliable and increases the efficiency in communications. With the use of AI and machine learning, satellites make real-time decisions without waiting for human commands [20]. The cognitive radio model finds the white spaces in the communication bands. These white spaces are already licensed, but currently unused segments of the spectrum in the communication band. This model uses these inactive segments until the primary user becomes active again. With the help of ML algorithms, this cognitive network could suggest the other possible white spaces to communicate. This minimizes the delay time in communications and maximizes the use of limited bands available by decreasing the need for human intervention. Therefore, the ML algorithms work in such a way to analyse and train the machine to suggest the new band or available bands for more effective communications.

5 Conclusion

Artificial intelligence and its educational application provide an efficient solution to various complex problems which results in improvements in the academic performance of every individual student. Education is the zone where intellectual skills are improved based on the teaching culture, academic environment, and extracurricular activities. The ability of AI enables students to develop in all aspects and create a positive learning environment, thus providing a high-quality education for all. The

ability to automate frequent tasks and grading system which saves teacher's valuable time. All these effective tools and techniques can be utilized to implement the framework for the education industry and we may experience an AI-powered education sector if this is completely implemented. The space sector is dealing with space in healthcare, space in communication, space mission planning, and space predictive approach. The future study is mobile remote education using edge computing, real-time analysis, and virtual personalized assistant. The cost factor should be taken into account while performing any task. The comparisons will be made based on the cost for a daily transportation from home to school/ institutions, time-saving while learning remotely, to access library and labs from anywhere with the help of VR, Virtual labs technologies to get the lookalike to feel.

References

1. Roll I, Wylie R (2016) Evolution and revolution in artificial intelligence in education. *Int J Artif Intell Educ* 26(2):582–599
2. Hwang GJ, Xie H, Wah BW, Gašević D (2020) Vision, challenges, roles and research issues of artificial intelligence in education
3. Kaur S, Tandon N, Matharou GS (2020) Contemporary trends in education transformation using artificial intelligence. In: *Transforming management using artificial intelligence techniques*. CRC Press, pp 89–103
4. Manavi SY, Nekkanti V, Choudhary RS, Jayapandian N (2020) Review on emerging internet of things technologies to fight the COVID-19. In: *2020 fifth international conference on research in computational intelligence and communication networks (ICRCICN)*. IEEE, pp 202–208
5. Ikedinachi AP, Misra S, Assibong PA, Olu-Owolabi EF, Maskeliūnas R, Damasevicius R (2019) Artificial intelligence, smart classrooms and online education in the 21st century: implications for human development. *J Cases Inf Technol (JCIT)* 21(3):66–79
6. Lu Y (2019) Artificial intelligence: a survey on evolution, models, applications and future trends. *J Manage Anal* 6(1):1–29
7. Wang K, Gou C, Duan Y, Lin Y, Zheng X, Wang FY (2017) Generative adversarial networks: introduction and outlook. *IEEE/CAA J Automatica Sinica* 4(4):588–598
8. Alhumaid K, Habes M, Salloum SA (2021) Examining the factors influencing the mobile learning usage during COVID-19 pandemic: an integrated SEM-ANN method. *IEEE Access* 9:102567–102578
9. Jayapandian N, Pavithra S, Revathi B (2017) Effective usage of online cloud computing in different scenario of education sector. In: *2017 international conference on innovations in information, embedded and communication systems (ICIIECS)*. IEEE, pp 202–208
10. Filgueiras F (2021) New Pythias of public administration: ambiguity and choice in AI systems as challenges for governance. *AI & SOCIETY*, 1–14
11. Sree SR, Vyshnavi SB, Jayapandian N (2019) Real-world application of machine learning and deep learning. In: *2019 international conference on smart systems and inventive technology (ICSSIT)*. IEEE, pp 1069–1073
12. Çetinkaya A, Baykan ÖK (2020) Prediction of middle school students' programming talent using artificial neural networks. *Eng Sci Technol Int J* 23(6):1301–1307
13. Ball SJ, Grimaldi E (2021) Neoliberal education and the neoliberal digital classroom. *Learn Media Technol* 1–15
14. Huang HY, Broughton M, Mohseni M, Babbush R, Boixo S, Neven H, McClean JR (2021) Power of data in quantum machine learning. *Nat Commun* 12(1):1–9

15. Batunova E, Popovich T, Smirnova O, Truhachev S (2020) Concept of intelligent decision-making support system for city environment management. In: Information fusion and intelligent geographic information systems. Springer, Cham, pp 167–178
16. Garcia-Piquer A, Ribas I, Colomé J (2015) Artificial intelligence for the EChO mission planning tool. *Exp Astron* 40(2–3):671–694
17. Kaddari Z, Mellah Y, Berrich J, Belkasmi MG, Bouchentouf T (2020) Natural language processing: challenges and future directions. In: International conference on artificial intelligence & industrial applications. Springer, Cham, pp 236–246
18. Zhang X, Nguyen H, Bui XN, Le HA, Nguyen-Thoi T, Moayedi H, Mahesh V (2020) Evaluating and predicting the stability of roadways in tunnelling and underground space using artificial neural network-based particle swarm optimization. *Tunn Undergr Space Technol* 103:103517
19. Bhavana K, Nekkanti V, Jayapandian N (2019) Internet of things enabled device fault prediction system using machine learning. In: International conference on inventive computation technologies. Springer, Cham, pp 920–927
20. Guo J, Li X, Chen M, Jiang P, Yang T, Duan W, Yu Q (2020) AI enabled wireless communications with real channel measurements: channel feedback. *J Commun Inf Netw* 5(3):310–317

Frame Duplication Detection Using CNN-Based Features with PCA and Agglomerative Clustering



Neetu Singla, Sushama Nagpal, and Jyotsna Singh

Abstract In a court of law, surveillance videos and recordings are the major sources of evidence for any incident or crime. With a simple video editor, the reality could be easily manipulated. This introduces the challenge of verifying the authenticity of contents before they may be used in any critical application domains. In this paper, a video forensic technique is proposed to detect frame duplication forgery in surveillance videos. The proposed technique utilizes convolutional neural network for integrity embedding generation and matching, for video frames duplication detection and localization. Through this work, we compare the efficacy of various deep learning models in generating true embeddings of surveillance footages. To perform experiments a dataset of over 100 authentic and 300 forged, high-resolution HEVC-coded video clips is prepared from surveillance clips. Experimental results indicate that the proposed technique is a good replacement of traditional hand-crafted and compression domain feature-based approaches.

Keywords Video forensics · Frame duplication · CNN · PCA · Agglomerative clustering

1 Introduction

With the advancement of technology smartphones, video recorders, and surveillance cameras are increasingly commonplace in many facets of daily life. In a city, offices, houses, roads thousands of cameras are collecting a huge amount of video data for monitoring purposes. These videos are crucial in providing convincing evidence in a court of law [1]. Video editing applications, such as Windows Movie Maker, Adobe Photoshop, Avidemux and Adobe After Effects, can effortlessly manipulate the reality. As a result, video forensics studies are being conducted to detect fake media from authentic content.

N. Singla (✉) · S. Nagpal · J. Singh
Netaji Subhas University of Technology, New Delhi, India
e-mail: neetu.co19@nsut.ac.in

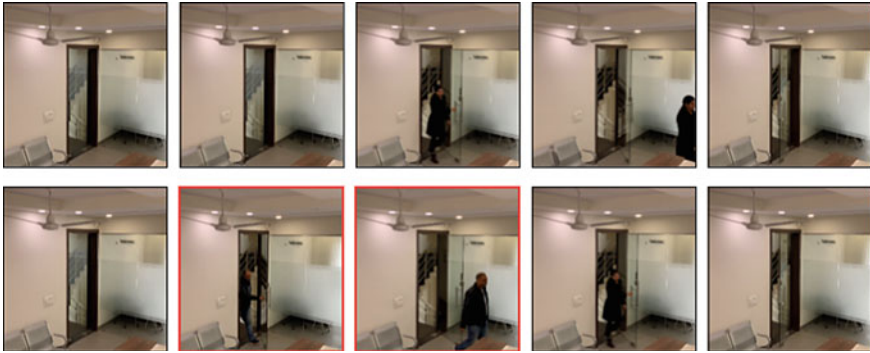


Fig. 1 Top row shows selected frames of an authentic video. Bottom row shows a frame duplication forged video. Frames depicting a person’s arrival have been copied from same surveillance video and incorporated at incorrect position

Frame duplication is considered to be the most undetectable type of forgery. It is a manipulation in which a sequence of frames is copied from a video and placed into the same video at a different timeline. In a real-case scenario, frame duplication forgery could be used to fabricate an individual’s presence at a location in an incorrect time. This scenario is depicted in Fig. 1 using an office surveillance footage. If such an altered video was used in a criminal investigation, then authorities could be easily misled.

In recent years, deep learning methods have experienced enormous success due to its strong potential to automatically learn features for large-scale visual data. Existing forensic methods rely on extracting hand-crafted features from video data. However, when dealing with large amounts of surveillance data, these methods become impractical due to low accuracy rates and significant computational complexity. The goal of this research is to look into the performance of various deep learning models for detecting frame duplication on huge surveillance dataset.

The rest of the paper is organized as follows. In Sect. 2, we discuss the related work for frame duplication detection. The proposed approach has been described in Sect. 2.4, and in Sect. 3 dataset and experimental results are presented. Finally, discussion and conclusion is provided in Sect. 4.

2 Related Work

Many approaches to frame duplication detection has been proposed in literature. These can be classified into three categories: discontinuity in correlations, similarity in statistical features and deep learning-based approaches.

2.1 *Discontinuity in Correlation-Based Approach*

In videos, a strong temporal correlation exists between adjacent frames. This correlation is disrupted when a frame sequence is duplicated between two frames. Authors in [2–4] exploited inconsistencies in correlation distribution to detect a tampering. Some authors extracted texture-based features such as, Zernike opponent chromaticity moments (ZOCM) [2], local binary pattern (LBP) [3] and Haralick features [4] to find the correlation between adjacent video frames. Huang et al. [2] utilized the difference in ZOCM between neighbouring frames to extract abnormal locations. However, this strategy does not work well for videos with a dynamic background. Fadl et al. in [5] extracted features from the compression domain. Authors analysed anomalies in spatial energy and temporal energy computed from the residues of MPEG-coded video stream. When the entire GOP or a multiple of a GOP is duplicated, the compression-based features cannot anticipate the discontinuity points.

2.2 *Similarity in Statistical Feature-Based Approach*

This category of algorithms extracts hand-crafted features such as SVD, SIFT, HOG from each frame of video and assesses them using a distance metric. Authors in [6] extracted singular value decomposition (SVD) features from each frame and computed Euclidian distance between a reference frame and other frames in a sub sequence. SIFT characteristics were used by several authors [7–9] to detect duplication. Ulutas et al. [8] presented a Bag of Word (BoW)-based technique in which each frame of the test video is analysed for SIFT key-points and descriptors. To make visual words, these descriptors are quantized. BoW features are formed by combining the frequency of visual words, and these features are then utilized to detect frame duplication attacks. The approach, however, does not function well in static scenes. To detect frame duplication forgeries in MPEG 4 video, Kharat et al. [9] suggested a two-stage approach. In the first stage, the motion vectors for all of the frames are obtained in order to categorize suspicious frames. The SIFT features of each frame are calculated in the next stage, and the ultimate decision to detect duplication forgery is made. In [10], Fadl et al. utilized an edge change ratio (ERC) algorithm to partition a video into variable length shots. Texture feature vectors are computed for each frame using Grey-Level Co-occurrence Matrix. Finally, the average feature vector of each shot is examined in order to identify duplicated shots. Mohiuddin et al. [11] employed the structural similarity index measure (SSIM) to evaluate the similarity between consecutive frames. Further, a searching technique was utilized to detect and locate duplicated frames. However, similarity-based techniques become inefficient when the duplicated frames go under basic image manipulations.

2.3 Deep Learning-Based Approach

Long et al. [12] employed I3D network to extract features from digital videos comprising both temporal and spatial information of video segments. A ResNet-based Siamese network was used for fine level correspondences between individual duplicated frames. On the VIRAT dataset, they were 84.05% accurate, while on the MFC dataset, they were 85.88% accurate. Recently, Munawar et al. [13] proposed a deep learning approach that integrates the functionalities of 3D-I3D model with Siamese-based RNN (recurrent neural network). The proposed method has an accuracy of 86.6% with the VIRAT dataset and 93% with the MFC dataset. Bakas et al. [14] introduced a motion vector analysis-based deep learning solution for detecting inter-frame forgeries in videos. However, they have not taken the advantage of CNN's ability to extract features from visual input.

Different from the existing methods for video forgery detection which mainly used hand-crafted features or correlations under compression domain, we take advantage of convolutional neural networks to extract global features as frame integrity embeddings, for frame duplication detection and localization. Further we analyse and compare some pretrained deep learning models, to test their feature learning ability for the proposed task.

2.4 Proposed Methodology

Embeddings are the representation of the visual data. Convolutional neural networks (CNNs) have been capable of learning resilient feature representations, allowing images with the same original content to appear close to each other in the embedding space, despite being modified using a set of image manipulations. Various computer vision tasks are performed using CNN-based feature representation learning such as semantic segmentation, object detection, style transfer and medical image similarity work. Surveillance videos generate huge amount of data every day. As per our knowledge, very few work has been done to visualize this data using CNN-based feature descriptions for duplication forgery detection. The proposed methodology is based on three fundamental components: CNN-based feature extractor, feature dimensionality reduction and agglomerative clustering for detecting duplicate frames. The flowchart of our proposed approach is shown in Fig. 2.

2.5 CNN-Based Feature Extractor

An input video is partitioned into frames. These frames go through a pretrained CNN model which outputs an integrity embedding for each frame. In proposed work, a

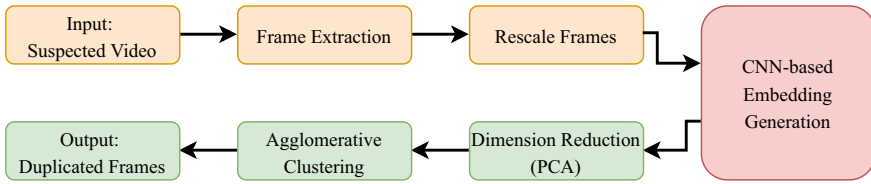


Fig. 2 A flowchart of proposed frame duplication detection approach

variety of baseline networks are examined namely, VGG16 [15], ResNet50 [16], Inception V3 [17] and Xception [18].

2.6 Dimensionality Reduction

The process of dimensionality reduction is critical for reducing computing time and storage. PCA converts image data from high dimensionality to low dimensionality based on the most significant aspects. Here we make use of PCA to reduce the dimensions from range of 10^4 to merely 2. This preserves the linearity of features between various points, allowing us to use distance as a metric in the following steps.

2.7 Agglomerative Clustering

After feature extraction and dimensionality reduction, the next step is similarity analysis. Agglomerative clustering is applied on principal components to obtain a relationship between frames. Small clusters are easily identified using agglomerative clustering. We select clusters with minimum variance or variance below a decided threshold and classify them as duplicates. It also allows us to work in cases where same frame is duplicated multiple times. To facilitate the interpretation of frame relationships, plots depicted in Fig. 2 were obtained for each baseline network.

3 Experiments and Analysis

3.1 Dataset and System Requirements

To conduct experiments, a self-collected video dataset is utilized. Videos were shot at various surveillance sites, using iPhone XS smartphone. The videos were high resolution (1920×1080), HEVC coded with a frame rate of 30 fps. These videos were divided into non-overlapping sequences of 200 frames each. In order to generate

forged videos with frame duplication tampering, we randomly select a sequence of 25 frames, and then re-inserted them into the same source video. X265 video codec libraries were used to generate the manipulated video corpus. Experiments are performed using Google Colab with GPU support. Following Python libraries are used in the implementation: OpenCV, Scikit-learn, TensorFlow, Numpy and Matplotlib.

3.2 Comparison of Pretrained CNN Models

Table 1 presents the comparison of features for various pretrained models: VGG16, ResNet50, Inception V3 and Xception. The comparison is performed in terms of input frame size, layer at which features are extracted, output embedding vector dimension and feature extraction time per frame.

3.3 Frame Duplication Detection Results

A performance of various baseline CNN models in terms of detection accuracy and model execution time is depicted in Fig. 3. Figure 4 graphically depicts the frame duplication results on a test video using all four proposed models. The x and y -axis of plot shows the two principal components generated after dimension reduction. Each frame is represented by red colour on plot. The blue coloured line connects the consecutive frames. The frames which are duplicated will be overlapped or nearly located in space. Duplicated frames are marked by light blue colour. With an accuracy of 94.9%, the Xception Model was the most accurate, while the Resnet50 network architecture took the least amount of time to detect forgeries. Table 2 compares the performance of proposed approach with other feature extraction architectures including compression based and hand-crafted features based.

Table 1 Comparison of the various baseline CNN models

Model	Input frame size	Output layer	Feature dimension	Extraction time
VGG16	270 × 480	Block5_pool (MaxPooling2D)	61,440	10.6
ResNet50	270 × 480	Conv5_block3_out (Activation)	276,480	101.5
Inception V3	270 × 480	Mixed10 (Concatenate)	186,368	18.8
Xception	270 × 480	Block_14_sepconv2_act (Activation)	276,480	20.1

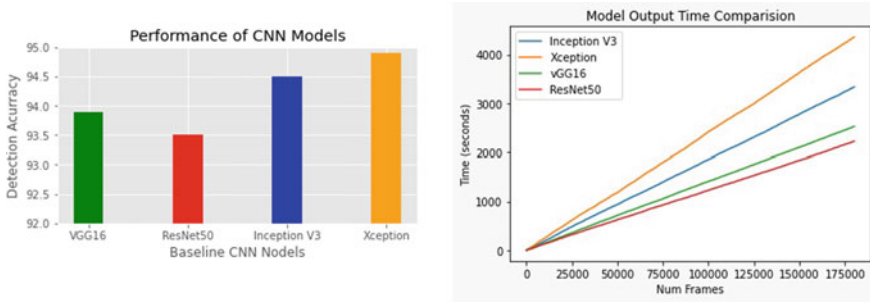


Fig. 3 Comparison of baseline CNN models for frame duplication detection in terms of accuracy and time consumption

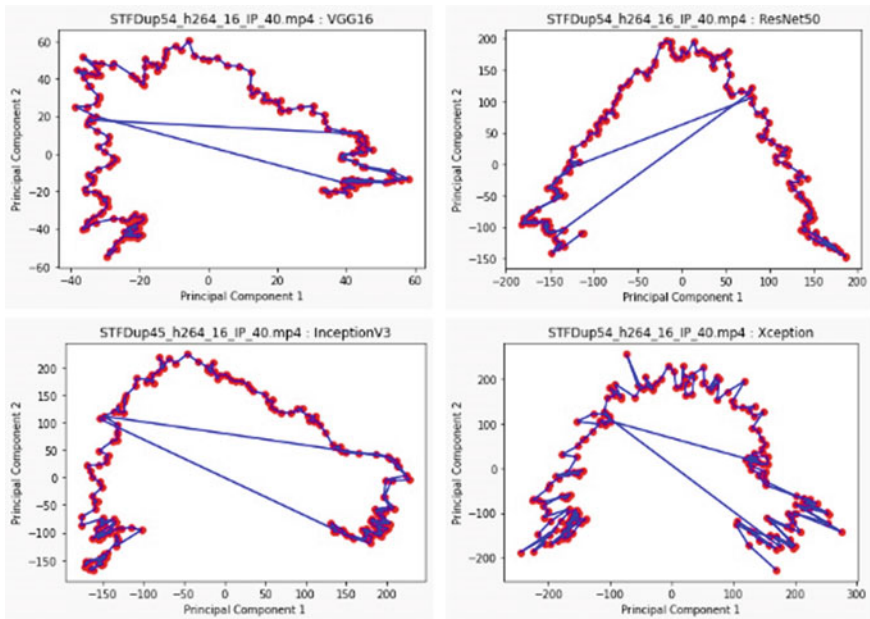


Fig. 4 Frame duplication detection

Table 2 Comparison of frame duplication detection approach with others

Method	Execution time	Detection accuracy
Wang and Farid [19]	294.67	0.658
Lin [20]	140.6	0.940
Yang [6]	0.58	0.91
Proposed	0.175	0.94

4 Conclusion and Future Work

We presented a novel approach for frame duplication detection in videos using CNN-based integrity embeddings. We analysed the effectiveness of four well-known CNN models for image-representation learning on surveillance data. We have shown how these characteristic embeddings may be tailored to effectively detect duplication in a video. We prove that CNN-extracted features are robust and can be successfully applied to duplication detection. In future work, we will investigate whether our system can be further improved by utilizing video embedding generation techniques or a substitute distance metric.

References

1. Shelke N, Kasana SS (2021) A comprehensive survey on passive techniques for digital video forgery detection. *Multimedia Tools Appl* 80:6247–6310
2. Liu Y, Huang T (2015) Exposing video inter-frame forgery by Zernike opponent chromaticity moments and coarseness analysis. *Multimedia Syst* 23:08
3. Zhang Z, Hou J, Ma Q, Li Z (2015) Efficient video frame insertion and deletion detection based on inconsistency of correlations between local binary pattern coded frames. *Secur Commun Netw* 8:01
4. Bakas J, Naskar R, Dixit R (2018) Detection and localization of inter-frame video forgeries based on inconsistency in correlation distribution between haralick coded frames. *Multimedia Tools Appl* 78:4905–4935
5. Fadl SM, Han Q, Li Q (2019) Inter-frame forgery detection based on differential energy of residue. *IET Image Process* 13:522–528
6. Yang J, Huang T, Su L (2014) Using similarity analysis to detect frame duplication forgery in videos. *Multimedia Tools Appl* 75:1793–1811
7. Pandey R, Singh S, Shukla KK (2014) Passive copy-move forgery detection in videos. In: 2014 International conference on computer and communication technology (ICCCCT), pp 301–306
8. Ulutas G, Ustubioglu B, Ulutas M, Nabiye V (2017) Frame duplication detection based on bow model. *Multimedia Syst* 24:549–567
9. Kharat J, Chougule S (2020) A passive blind forgery detection technique to identify frame duplication attack. *Multimedia Tools Appl* 79:8107–8123
10. Fadl SM, Megahed A, Han Q, Li Q (2020) Frame duplication and shuffling forgery detection technique in surveillance videos based on temporal average and gray level co-occurrence matrix. *Multimedia Tools Appl* 79:17619–17643
11. Mohiuddin S, Malakar S, Sarkar R (2021) Duplicate frame detection in forged videos using sequence matching. In: CICBA
12. Long C, Basharat A, Hoogs A (2019) A coarse-to-fine deep convolutional neural network framework for frame duplication detection and localization in forged videos. In: CVPR workshops
13. Maryam Munawar IN (2021) Duplicate frame video forgery detection using siamese-based RNN. *Intell Autom Soft Comput* 29(3):927–937
14. Bakas J, Naskar R, Bakshi S (2021) Detection and localization of inter-frame forgeries in videos based on macroblock variation and motion vector analysis. *Comput Electr Eng* 89:106929
15. Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition
16. He K, Zhang X, Ren S, Sun J (2015) Deep residual learning for image recognition

17. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z (2015) Rethinking the inception architecture for computer vision
18. Chollet F (2016) Xception: deep learning with depthwise separable convolutions. CoRR, vol. abs/1610.02357
19. Wang W, Farid H (2007) Exposing digital forgeries in video by detecting duplication. In: MM&Sec
20. Lin G-S, Chang J-F (2012) Detection of frame duplication forgery in videos based on spatial and temporal analysis. Int J Pattern Recognit Artif Intell 26

Detection of MA Based on Iris Blood Vessel Segmentation and Classification Using Convolutional Neural Networks (ConvNets)



S. Karthika and M. Durgadevi

Abstract Deep learning and artificial intelligence (AI) play a vital role in the biomedical field for segmenting and classifying various diseases. By the use of AI, highly precise and efficient systems can be developed with which doctors can identify and diagnose diseases at an early stage and without the extensive resources available in specialist clinics. The detection of MA in fundus images remains an open problem in the medical imaging process due to the poor reliability (with existing detection or deduction methods). Detection of diabetic retinopathy in earlier stages is essential for preventing blindness. Detection of microaneurysms (MA) is the initial stage in DR, which is present in the retina with a slight swelling on both sides of the blood vessel. The detection of MA in fundus images remains an open problem in the medical imaging process due to poor reliability. Convolutional neural networks (ConvNets) can achieve an accuracy of 98.358% in the detection of the microaneurysms using publicly available Kaggle datasets. This paper tends to list the various strategies and methods used to detect microaneurysms using ConvNets.

Keywords Deep learning · Retinal images · Microaneurysms · Diabetic retinopathy · ConvNets

1 Introduction

When diabetic blood vessels in the retina are destroyed, diabetic retinopathy (DR) occurs. DR is a major cause of vision loss worldwide if not properly treated [1]. The fundus disease is diagnosed, and then, the patient's condition is evaluated. DR detection is slow and tedious and may cause more errors. More than 80% of new cases can be reduced with correct eye care and treatments. Diabetic retinopathy is

S. Karthika (✉) · M. Durgadevi
Department of Computer Science and Engineering, College of Engineering and Technology, SRM Institute of Science and Technology, Vadapalani Campus, No.1, Jawaharlal Nehru Road, Vadapalani, Chennai, Tamil Nadu, India
e-mail: ks6928@srmist.edu.in

M. Durgadevi
e-mail: durgadem@srmist.edu.in

classified according to the degree into five classes (0–4), namely 0—normal, mild—1, moderate—2, severe—3, and proliferative or disease-free—4. Identifying DR is difficult due to delaying results leading to misunderstanding and processing delays. Doctors can detect DR by the presence of vascular abnormalities associated with lesions caused by the disease. Different types of symptoms include MA, vascular blood loss, abnormal growth in new blood vessels, and damage of nerves [2]. Due to the lack of medical resources and a high number of diabetics in some areas, many patients affected by DR are unable to diagnose and treat patients on time. This lag of diagnosis extends into a patient's treatment, thereby causing loss of irreversible vision and even consequences of blindness. For a long time, people have recognized the need for a comprehensive and automated DR detection method, and great efforts have been made in the past to classify images and models. This study aims to provide a model for diagnosing diabetic retinopathy that automatically learns the features critical to diagnosing the disease stage without manually extracting the features. In the first phase of diabetic retinopathy, there will be virtually no differentiation in the tissue layer to be determined, but the delay in DR will be severe and result in visual loss. Diabetic retinopathy is classified by its features such as microaneurysm, hemorrhages, and bleeding in red lesions such as hard exudates and soft exudates. When a blood clot forms in retinal vessels, a patient's first sign of DR is a microaneurysm. Hemorrhages can occur if microaneurysms are not addressed. If the condition is not treated promptly, exudates may form on the retina, resulting in permanent vision loss or impairment. Consequently, the microaneurysm (MA) is the initial symptom of diabetic retinopathy; here, non-contrast images of the retina are used for screening and periodic examination.

Early detection through regular screening is recommended in patients with diabetes to prevent visual disturbances and loss of vision. It must be checked annually, which is an important task for ophthalmologists. It is therefore important to develop an appropriate automated DR screening quality system that not only reduces the workload of the ophthalmologists but together improves detection accuracy. Many analytics teams develop methods of detecting MA using machine-driven techniques. This paper compares the many ways to automatically detect diabetic retinopathy based on parameters such as sensitivity, specificity, accuracy, and mean square root. The purpose of this study was to review the literature used to detect diabetic retinopathy and provide researchers with a comprehensive resource on commercially available methods to detect diabetic retinopathy. Figure 1 shows the different stages of lesions' images.

Because automatic DR imaging may detect DR at an early stage, it is a useful tool to delay or prevent vision damage, and early detection and treatment are essential. These methodologies can be summarized by saying that deep learning technology has transformed the field of many academics who are particularly interested in the use of convolutional neural networks (CNNs) for image classification. These features and blood vessels are segmented and classified in this field of research. There has been tremendous improvement in the classification of DR fundus images using natural image classification techniques and new research.

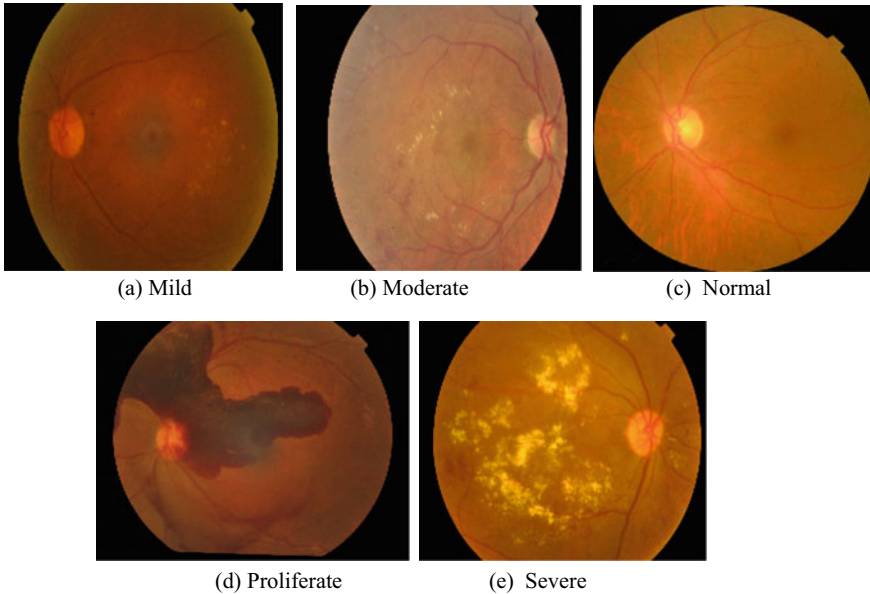


Fig. 1 Sample images of lesions in the fundus

The purpose of this paper was to categorize the fundus image to do an automatic DR diagnosis, so that real-time classification of the patient condition may be accomplished. Rather than relying on the doctor’s skill and manual operation to diagnose and treat DR, this system does it in an automated and highly accurate manner [3]. With the help of several image preprocessing methods, it extracts many significant features and classifies them into five classes of datasets that are analyzed using CNN. For each model, its sensitivity, specificity, accuracy, and mean square root (MSR) are determined. For example, augmented data will increase the training examples and normalized data will reduce noise to accurately predict the finest image dataset to train for our model. This paper is divided into the following sections. Section II discusses the related work of ConvNets. Section III shows how to preprocess the fundus image and introduces the methodologies used in CNN and Kaggle datasets. Section IV contains the experimental data and analyses. Lastly, Section V addresses the conclusion.

2 Relative Work

There have been numerous studies that have attempted to detect and segment lesions in fundus pictures automatically or to create numerical indices. According to Shahin et al. [4], a technique was established to characterize iris fundus images as proliferative diabetic retinopathy present or not present. Entropy and homogeneity were

extracted via morphological processing in addition to pathological characteristics including blood vessel area and exudates. Incorporating the variables into a shallow network, a sensitivity of 88.01% and a specificity of 100% are obtained.

Jaafar et al. [5] have developed an algorithm that uses a top-down approach to identify and evaluate hard exudates and a polar coordinate system that is focused on foveal location. On a limited sample of 236 fundus images, it is 93.21% sensitive. More than 90% accuracy is achieved by using the random forest method [6] developed by R. Casanova et al. In addition to the number of microaneurysms, human retinal images graded by specialists can influence the outcome. On three benchmark datasets [7], the trained support vector machine (SVM) attained a 93% accuracy rate. On the other hand, in a study [8], texture features from a local binary pattern were used to detect exudates and the classification accuracy was reported at 96.7%. A bootstrapped decision tree [9] was used to classify fundus images in a second dual classifier model. Creating two binary vessel maps involves reducing the number of feature vectors.

For vessel segmentation, Wang et al. [10] extract image features using a CNN model (LeNet-5). There are certain drawbacks to these methods. The accuracy of the data collection cannot be guaranteed because the characteristics are extracted manually and empirically. As a second issue, the dataset is short and of poor quality, with only a few hundred or possibly a few dozen fundus images, relative to the collection environment, making it difficult to compare how well an algorithm performs in an experiment. Krizhevsky [11] in 2012, the AlexNet architecture was introduced to improve performance, and to increase the use of DCNN in computer vision. After proposing several excellent CNN architectures, such as VggNet [12, 13] and GoogleNet [14], ResNet [15] was proposed in 2015 as one of the key network models to further improve the performance of CNN image classification. Finally, it provides automatic and accurate identification to minimize visual loss by utilizing learning transfer to shorten the time and compares the performances with GoogleNet, AlexNet, VggNet, and ResNet. There are several improvements in our method's convergence time and classification performance for big experimental datasets compared with earlier methods.

3 Model Description and Methodologies

This section explains the flow of work, steps, and methodologies used in detail in the following.

3.1 *Workflow Model*

The DCNN architecture uses enlarged images of the input taken and performing data argumentation is the first step. Preprocessing is the next stage that is used to enhance and resize the input image. RGB images are compressed into three channels

called red, green, and blue. RGB image can be achieved by dividing input images into three channels, and mainly, green channel is used to remove noisy contents in the image. The blue channel contains low contrast and does not include much data information. Blood vessels are visible only in the red channel. The extraction of the green channel is segmented according to region growth, and statistical and geometric features are extracted. The growing area approach is a bottom-up method, and it is the opposite of the partitioning and combining approach. Then, a machine learning classification called ConvNets is used to capture microaneurysm (MA) with different DR stages, namely normal—0, mild—1, moderate—2, severe—3, and proliferative or disease-free—4.

This study aims to detect stage I diabetes in the retina and is based on MATLAB graphical user interface (GUI). Each of them consists of an m-file and a figure file. These are programmable files that contain image information. Figure 2 shows the overall workflow of the method.

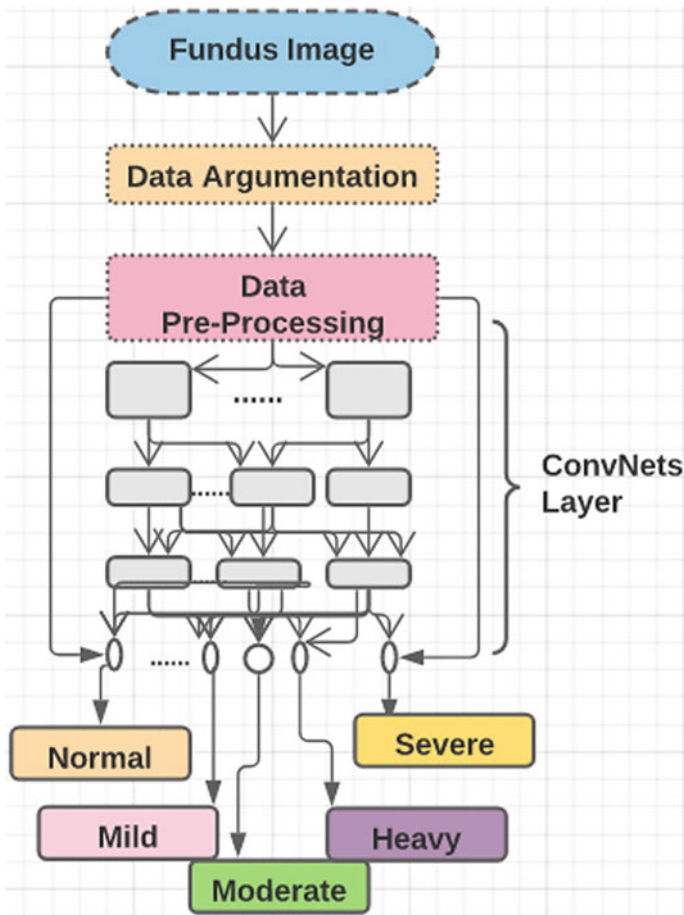


Fig. 2 Overall workflow

3.2 Steps to Detect the Stage-I Microaneurysm in DR

```

Input: Retinal Dataset
Output: Microaneurysm present or not
Step 1: Let Number of datasets=D, Number of trained sets =T
Step 2: For each dataset
    For 1 to n do
        Randomly select data from the training dataset
    Let O =Input image, R1=Resized image
Step 3: Perform green channel extraction for the input image using RGB function
Step 4: Convert green channel to Grayscale image
Step 5: For blood vessel detection
    do
        Perform image approximation in fundus image using canny detection
    else
Repeat the steps until the stopping criterion
end
Step 5: For image segmentation
    do
        Perform region growing function in R1 and display
        Fundus image is visible with a dot of microaneurysm detection
    else
Repeat the procedure until the stopping criterion
Step 6: For hog feature extraction
    Calculate the mean, standard deviation, entropy, and mean square root
Step 7: For classification using CNN
if O == 1 display
Mild Microaneurysm is present
else
Microaneurysm is not present
Step 8: end-all

```

3.3 Data Preprocessing and Methodology

Datasets and Preprocessing. Our datasets were used from the publicly available Kaggle [16] website. These datasets are rated by professionals according to five classes (0–4) which include high-resolution retinal images. The split ratio used for training and testing data is 80:20 using MATLAB. The datasets contain more than 2537 high-resolution RGB images with multiple varieties of imaging with a resolution of about 3500×3000 . Table 1 shows the distribution of input datasets which consist of four labels in a range from L0 to L4 which denote normal, mild, moderate,

Table 1 Distribution of original data

Label	Class	No. of images
L0	Normal	512
L1	Mild	698
L2	Moderate	530
L3	Severe	497
L4	No	300

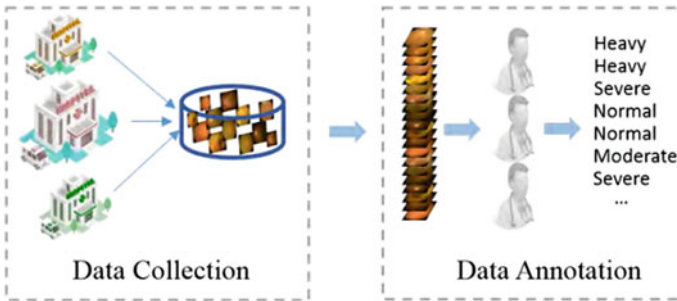


Fig. 3 DR data collection and annotation

severe, and no DR, respectively. Figure 3 shows the normal DR data collection and annotations using the classes from Table 1.

Preprocessing is a key process on the way to progress image resolution, and the outcomes can be used as training data with five classes. Due to noisy data and a restricted amount of datasets, stabilization techniques and data augmentation are used for preprocessing. In addition, the fundus image is cropped to a lower size to minimize the irrelevant parts. Preprocessing is to improve the quality of the images to confirm the dataset is consistent and has relevant characteristics. High brightness, low contrast, and noise are invariably present in fundus images of the retina, which has serious implications for the DR diagnostic process as well as the automatic detection of lesions, especially in MA [17]. The calculations were selected based on written suggestions for medical image processing. In digital graphics and images, resizing an image refers to changing the size of a digital image. In video technology, enlarging digital material is thought of as increasing or improving the resolution. To ensure that machines learn actual DR properties rather than device-specific information, the fundus images should be processed from various bases and converted into a standard setup using the below procedures:

Standardization of Size. Reducing or resizing various images to a similar measure is the primary step, and the fundus areas in different images have the same diameter. As soon as the images are added horizontally and vertically, the black borders on each side of the fundus image are removed. The images are then resized according to a set of parameters.

Regularization of Shape. Some images are complete circles; however, others may lack the top and bottom margins. The circle's edge is also detected by numerous systems. To unify the structures of these photographs, it uses a mask that covers the biggest common region of all images from multiple sources [18]. Figure 4 shows the original input image used for detection of DR, and Fig. 5 shows the resized image. Whenever you resize a graphic image, you must generate a replacement image with the closest or lower range of pixels.

RCB to Grayscale Extraction. After each image is preprocessed, its color needs to be converted into an RGB image. This is achieved by dividing the retinal image into three channels and using one channel called the green channel, to provide accurate information. The blue channel is difficult to distinguish and contains less information. The blood vessels are only visible when the red channel is extracted. The original image (RGB) is converted into the corresponding color space for further

Fig. 4 Original image of iris

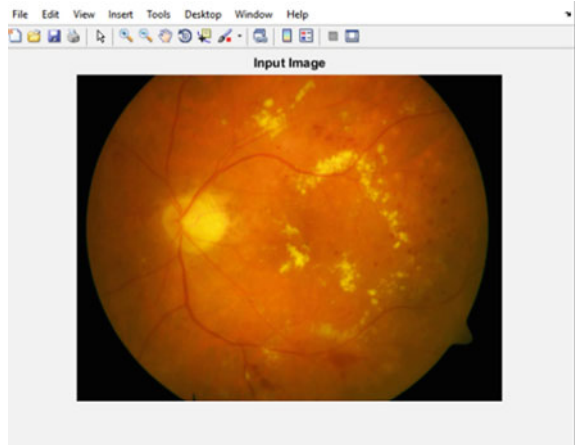
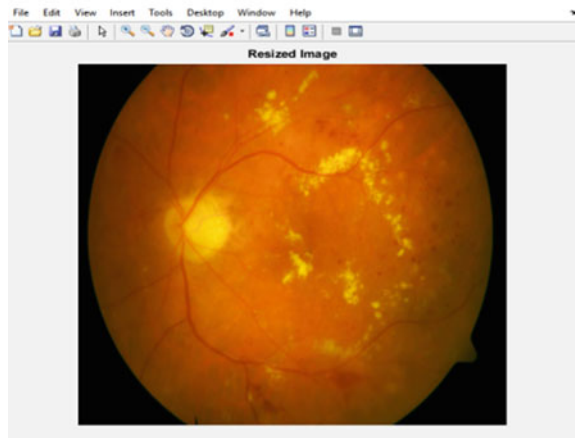


Fig. 5 Resized image from original image



processing. The color change is done after standardizing each image here, and the color modification method is simple: It converts each of an iris image’s RGB channels to a pre-calculated mean and deletes standards greater than 255 as shown in the following [19]:

$$Rx = \text{minimum} \{ Rx / \text{mean}(RC) \cdot r1, 255 \} \tag{1}$$

$$Gx = \text{minimum} \{ Gx / \text{mean}(GC) \cdot g1, 255 \} \tag{2}$$

$$Bx = \text{minimum} \{ Bx / \text{mean}(BC) \cdot b1, 255 \} \tag{3}$$

where $r1$, $g1$, and $b1$ from Eqs. (1), (2), and (3), respectively, are mean standards estimated across 2000 fundus images taken well with illuminations: RC, GC, BC. Rx , Gx , and Bx represent individual pixels in each channel. For each channel, the total input values are almost similar for various iris images, providing different learning models such as neural networks that have a more constant range of input. As shown in Fig. 6, firstly the image is converted to a green image to enable the blood vessel segmentation. In visual inspection, the optic disk and the background in the green channel form a high contrast, and in Fig. 7, the green image is translated into grayscale for normalization purposes.

Image segmentation and feature selection. Image segmentation is a technique that divides an image into one or more groups of segmented pixels (also called superpixels) [20]. The main purpose of segmentation is to make the image meaningful and simpler. In particular, image segmentation is a process of labeling every pixel in the image so that pixels with the same label have common attributes. Image segmentation is usually done to define objects and boundaries such as straight lines

Fig. 6 Green channel

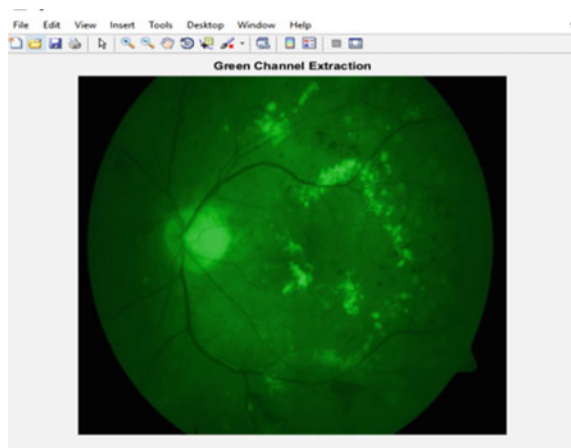
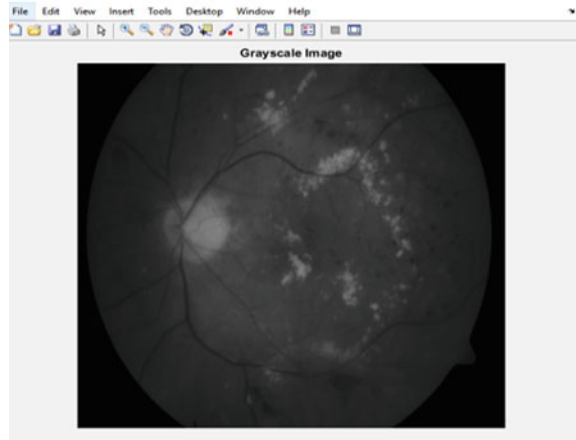


Fig. 7 Grayscale image



and curves in an image. To do this, Antal et al. [21] used contrast-limited adaptive histogram equalization (CLAHE).

In CLAHE, each neighborhood pixel is exposed to a contrast limit, which prevents the noise from over-amplification. Two-level DR images were segmented using histogram-based segmentation [22]. An image is segmented into regions where each pixel has a color-to-one relationship with the principal colors in the image at the first level. Histograms (H) are formed in phase one, and there is no comparison between the little H peaks and that of the variable. Second, for each segmenting region based on size merging is done to reduce the segments in images. Figure 8 shows the approximation in blood vessels with a black and white display, and Fig. 9 shows the approximation in the fundus image.

The selection of features starts from the first measurement datasets, and the values obtained are information rich rather than redundant, which is conducive to subsequent training and generalization stages. The feature extraction is called reduction. The

Fig. 8 Approximation in the region growing

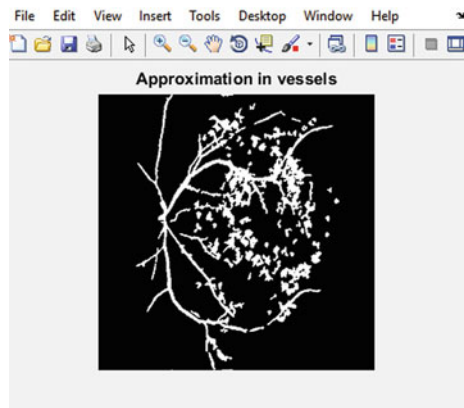
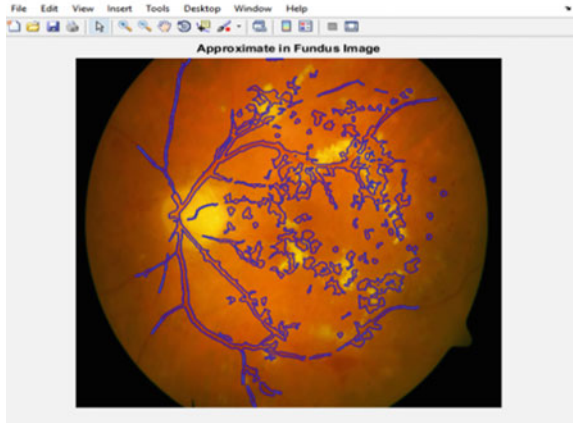


Fig. 9 Approximation in the fundus image



oriented gradient histogram [23] is used in image and image processing to recognize objects. The technique counts the occurrence of an orientation gradient in localized picture elements.

This technique is similar to histogram edge orientation, which calculates the invariant scale features and shape contexts that differs. It is used to improve accuracy in overlapping [24] of local contrast normalisation. Figure 10 shows the microaneurysm detection with dot representation. Figure 11 shows the approximation in the fundus image, and Figs. 12 and 13 show the results in the dialog box using the MATLAB tool.

Fig. 10 Microaneurysm detection

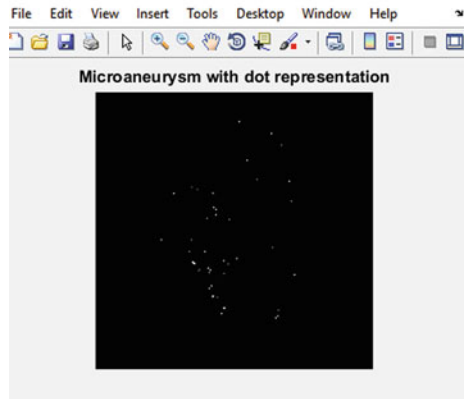


Fig. 11 Microaneurysm approximation in fundus image

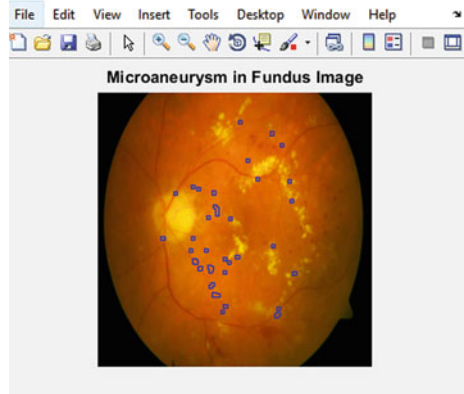


Fig. 12 Result of MA present in dialog box

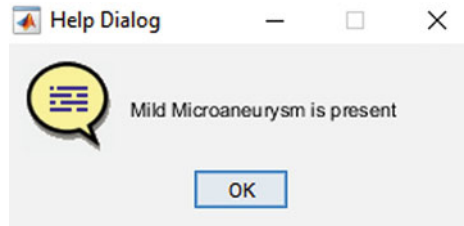
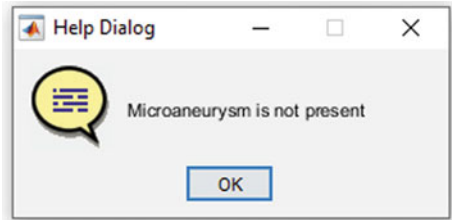


Fig. 13 Results of MA not present in dialog box



4 Experimental Analysis and Result

4.1 Indicators of Performance

To evaluate the performance of the provided ConvNets model, three assessment metrics were used: sensitivity, specificity, and accuracy. The performance is evaluated using the following parameters:

- A- Correctly identified, i.e., DR found as DR.
- B- Correctly rejected, i.e., non-affected found as non-affected.
- C- Wrongly identified, i.e., non-affected found as affected.
- D- Incorrectly rejected, i.e., affected found as non-affected.

Table 2 Details of hyperparameters

Hyperparameter	Description	Range
Learning rates	learning_rates	0.001
Batch size	batch_sizes	32
Optimizer	Optimizer	Adam

These parameters are used to calculate the sensitivity, specificity, accuracy, and mean square using (4), (5), (6), and (7), respectively. These indicators are given in the following.

Sensitivity: the likelihood of a patient being tested positive when they have a disease, and it is given by

$$\text{Sensitivity} = \frac{A}{A + D} \tag{4}$$

Specificity: the likelihood of a patient being tested negative when the patient is not sick and it is given by

$$\text{Specificity} = \frac{B}{C + B} \tag{5}$$

Accuracy: samples classified correctly according to size, determined by

$$\text{Accuracy} = \frac{A + B}{A + B + C + D} \tag{6}$$

Mean square root (MSR) is a network performance functionality. The standard error of the performance is calculated.

$$\text{Performance} = \text{mse}(\text{net}, t, y, \text{ew}) \tag{7}$$

where the net is the neural network, t is the target cell–matrix or array, y is the result cell array, and ew is the error weight.

In Table 2, the trained hyperparameters are listed alongside their descriptions.

4.2 Results and Discussion

Although the AlexNet [25–28] model performed well in describing the techniques when compared with ConvNets, it was unable to exceed it. In terms of classification, the VggNet [26–28] model performed significantly well than other methods. However, it performed worse with the presented model and AlexNet models in comparison. When it came to classifying the data, VggNet and GoogleNet [26, 27]

Table 3 Comparison of different models

Model used	ACC	Sensitivity	Specificity
ConvNets	98.38	97.5	96.5
AlexNet	96	92	97
GoogleNet	93.07	78	92
ResNet	90.53	89	96
VggNet-S	96.9	86	97
VggNet-16	93.9	91	94
VggNet-19	94.05	89	96

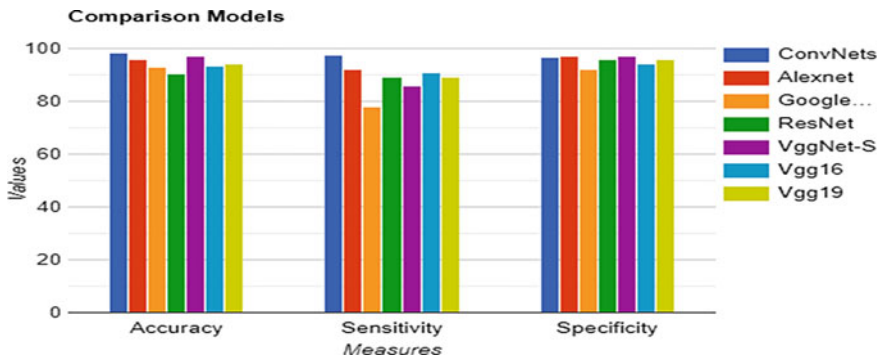


Fig. 14 Comparison of various classifier models

all produced the same results. On the other hand, they are not quite as efficient as the ConvNets. The AlexNet model has the lowest classification accuracy. The results of a detailed comparison with other recently created deep learning models are given in Table 3. On the DR image dataset, Fig. 14 displays the comparison results of the proposed model with different techniques. It appears from the table values that the proposed technique will have exceptional classification performance.

To further guarantee the goodness of the ConvNets model, an analysis was also performed as given in Table 3. The table results demonstrate that the ConvNets models gained high analysis over other approaches. Next to it, the AlexNet and GoogleNet models delivered moderate data. The entire experiment confirms the suitability of the provided ConvNets model for efficient diagnosis of DR due to its promising characteristics such as employment of histogram-based segmentation and the newest deep learning model. Figure 15 explains the training state and validation checks at epoch 25. Figure 16 shows the validation performance at epoch 19. According to the testing results, the proposed approach had a high accuracy of 98.38, a sensitivity of 97.5, and a specificity of 96.5. Figure 17 shows the overall experimental results of MA detection with a validation loss of 0.4.

Fig. 15 Training state at epoch 25

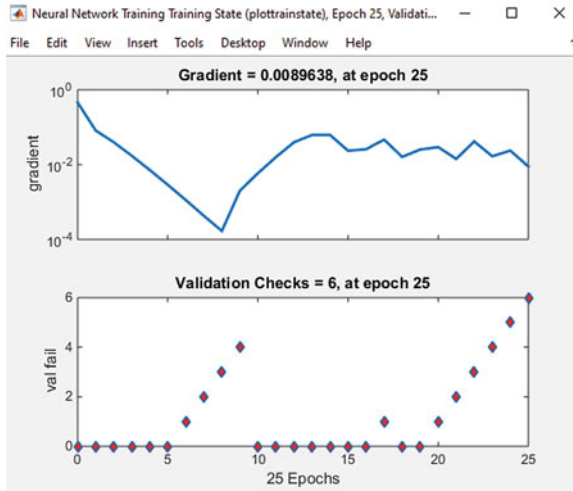
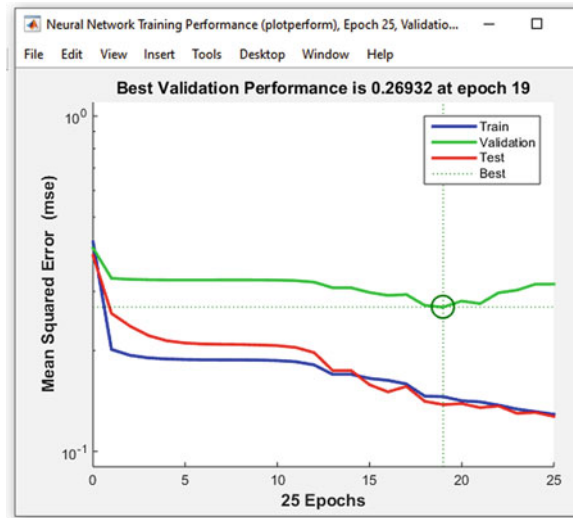


Fig. 16 Performance of the MA at epoch 19



5 Conclusion

As the diabetic population grows, so does the demand for automatic diabetic retinopathy diagnosis systems. Many subproblems, such as vessel segmentation and lesion identification, have yielded satisfactory results so far. Although these findings were attained on relatively modest datasets, real-world applications are still a way off. This paper presents the design, architecture, and implementation of ConvNets for automatic detection and classification of DR using MATLAB. DR poses several issues. Because of low contrast, it is difficult to distinguish the affected areas of

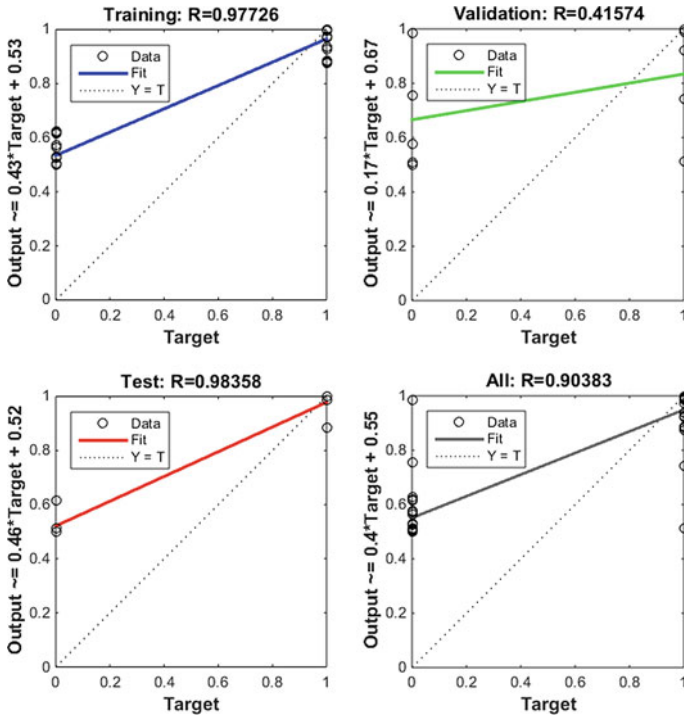


Fig. 17 Overall test results of MA detection with validation loss

DR. So, this paper uses a denoising technique to remove the noise and to achieve high accuracy compared with different pre-trained models. The secondary feature selection methods must be efficient so that features without useful information are irrelevant and redundant. There are 98.38 accuracy, 97.5 sensitivity, and 96.5 specificity for the proposed approach according to the testing results. Filtering techniques can be used to improve image quality before processing in future. More AlexNet and inception techniques can be explored to improve the performance of the suggested method by changing the hyperparameters. All known approaches are reviewed in this study to provide a thorough overview of the area. For machine learning and data mining applications, the learning machine’s processing speed and accuracy must also be considered. This model is used in all biomedical image applications for the prediction and detection of diseases. All the existing models use CNN as a supervised technique, but the proposed model uses an unsupervised technique with a good prediction of MA using retinal images. The challenge faced here is only small datasets are supported than the large datasets while using MATLAB. So, this can be overcome by using pre-trained models. As a result of this work, researchers will get a greater understanding of the issue and may be able to build more effective and efficient algorithms.

References

1. Cree MJ, Olsoni JA, McHardy KC, Forrester JV, Sharp PF (1996) Automated microaneurysms detection. IEEE conference, pp 699–702
2. Fleming AD, Philip S, Goatman KA (2006) Automated microaneurysm detection using local contrast normalization and local vessel detection. *IEEE Trans Med Imag* 25(9):1223–1232
3. Ahmad A, Mansoor AB, Mumtaz R, Khan M, Mirza SH (2015) Image processing and classification in diabetic retinopathy: a review. In: *Proceedings of European workshop on visual information processing*, pp 1–6
4. Shahin EM, Taha TE, Al-Nuaimy W, El Rabaie S, Zahran OF, El-Samie FEA (2013) Automated detection of diabetic retinopathy in blurred digital fundus images. In: *Proceedings of 8th international computer engineering conference*, pp 20–25
5. Jaafar HF, Nandi AK, Al-Nuaimy W (2011) ‘Automated detection and grading of hard exudates from retinal fundus images. In: *Proceedings of 19th European signal processing conference*, pp 66–70
6. Casanova R, Saldana S, Chew EY, Danis RP, Greven CM, Ambrosius WT (2014) Application of random forests methods to diabetic retinopathy classification analyses. *PLoS ONE* 9(6):e98587
7. Cunha-Vaz JG (2002) Measurement and mapping of retinal leakage and retinal thickness—surrogate outcomes for the initial stages of diabetic retinopathy. *Curr Med Chem-Immunol EndocrMetab Agents* 2:91–108
8. Anandakumar H, Umamaheswari K (2017) Supervised machine learning techniques in cognitive radio networks during cooperative spectrum handovers. *Cluster* 20:1–11
9. Omar M, Khelifi F, Tahir MA (2016) Detection and classification of retinal fundus images exudates using region based multiscale LBP texture approach. In: *2016 International conference on control, decision and information technologies (CoDIT)*
10. Wang S, Yin Y, Cao G, Wei B, Zheng Y, Yang G (2015) Hierarchical retinal blood vessel segmentation based on feature and ensemble learning. *Neurocomputing* 149:708–717
11. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: *Advances in neural information processing systems*, pp 1097–1105
12. Parkhi OM, Vedaldi A, Zisserman A et al (2015) Deep face recognition. In: *BMVC*, vol 1, p 6
13. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:14091556*
14. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A et al (2015) Going deeper with convolutions. In: *Cvpr*
15. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 770–778
16. Kaggle Diabetic Retinopathy Detection Competition. Accessed 18 Oct 2018. <https://www.kaggle.com/c/diabetic-retinopathy-detection>
17. Karthika S, Johnson S (2019) Survey on convolutional neural network based efficient automated detection of micro aneurysm in diabetic retinopathy. *Int J Sci Res Comput Sci Eng Inf Technol (IJSRCSEIT)* 5(3):361–368. ISSN: 2456-3307 Available at DOI: <https://doi.org/10.32628/CSEIT195333>, Journal URL: <http://ijsrcseit.com/CSEIT195333>
18. Wan S, Liang Y, Zhang Y (2018) Deep convolutional neural networks for diabetic retinopathy detection by image classification. *Comput Electr Eng* 72:274–282. <https://doi.org/10.1016/j.compeleceng.2018.07>
19. Gao Z, Li J, Guo J, Chen Y, Yi Z, Zhong J (2018) Diagnosis of diabetic retinopathy using deep neural networks. *IEEE Access*, 1–1. <https://doi.org/10.1109/access.2018.2888639>
20. Antal B, Hajdu A (2012) An ensemble-based system for microaneurysm detection and diabetic retinopathy grading. *IEEE Trans Biomed Eng* 59(6):1720–1726
21. Shankar K, Sait AR, Gupta D, Lakshmanaprabu SK, Khanna A, Pandey HM (2020) Automated detection and classification of fundus diabetic retinopathy images using synergic deep learning model. *Pattern Recogn Lett* 133:210–216
22. Chen H, Zeng X, Luo Y, Ye W (2018) Detection of diabetic retinopathy using deep neural network. In: *2018 IEEE 23rd international conference on digital signal processing (DSP)*

23. Ahmad A, Mansoor AB, Mumtaz R, Khan M, Mirza SH (2014) Image processing and classification in diabetic retinopathy: a review. In: 2014 5th European workshop on visual information processing (EUVIP). IEEE, pp 1–6
24. Zhang W, Zhong J, Yang S, Gao Z, Hu J, Chen Y, Zhang Y (2019) Automated identification and grading system of diabetic retinopathy using deep neural networks. *Knowl Base Syst* 175:12–25
25. Zhao Z, Zhang K, Hao X, Tian J, Heng Chua MC, Chen L, Xu X (2019) Bira-net: bilinear attention net for diabetic retinopathy grading. In: 2019 IEEE international conference on image processing, ICIP, pp 1385–1389
26. Lam C, Yi D, Guo M, Lindsey T (2018) Automated detection of diabetic retinopathy using deep learning. In: AMIA Joint summits on translational science proceedings, pp 147–155. ISSN 2153-4063
27. Wang X, Lu Y, Wang Y, Chen W-B (2018) Diabetic retinopathy stage classification using convolutional neural networks. In: 2018 IEEE international conference on information reuse and integration (IRI), IEEE, pp 465–471. ISBN 978-1-5386-2659-7
28. Li T, Gao Y, Wang K, Guo S, Liu H, Kang H (2019) Diagnostic assessment of deep learning algorithms for diabetic retinopathy screening. *Inf Sci* 501:511–522

Implementation of Laboratory Information Management to Medical Analyzer Data Integration



Devashri Raich , Yashpal Singh, and Asha Ambhaikar

Abstract Pathology laboratory playing major role in healthcare diagnostics and with development in automated testing methods machine or the analyzer taking place of manual testing process. These machines perform accurate and consistent testing. The result generated or populated by the analyzers are used for final report generation using manual method or through report generation software like laboratory information management system (LIMS). Different advance analyzers support direct data exchange with computer or supporting connected system for automated data exchange between analyzer and system software to avoid the manual human errors in reporting. Development of these types of integration software needs different levels of skillsets and understanding of analyzer integration methodologies apart from programming skill. Different machine works on different physical connection modes and the different communication protocols. Different subcomponent or the software tools required for successful implementation of analyzer integration. This paper deals with the methodology study and covering the skillsets required for LIS integration, and it also covers the different protocols may be used for data exchange.

Keywords LIS · Analyzer · HL7 · ASTM · Host · Control characters

1 Introduction

Medical device integration technologies are evolving at a rapid pace, and medical data interoperability is an important step in improving patient care. There is a wealth of streaming data from a variety of devices, including testing at the point of care, such as glucometers, blood analyzers, heart rate monitors, and more. Various NLP tools are also used for data analysis [1, 2]. When these data sources are combined

D. Raich (✉) · Y. Singh · A. Ambhaikar
Department of CSE, Kalinga University, Raipur, India
e-mail: devashriraich@gmail.com

Y. Singh
e-mail: yashpal.singh@kalingauniversity.ac.in

with powerful tools and external systems such as electronic medical records (EMR), they can save you real time.

Integrated medical devices, with outcome data properly fed into hospital information systems (HIS) and other downstream systems, create new opportunities for healthcare providers, medical device manufacturers, and healthcare IT companies to plan, billing accuracy, and improve care for patients [3, 4]. Medical device integration is primarily intended to automatically transfer recorded data to supported systems, which may include hardware or software modules. This will improve the quality of medical information about the patient and the system for obtaining pathology results. By physically and logically connecting to medical devices, record data can be entered directly into HIS, RIS, or LIS.

Since the medical device supports standard communications, not every medical device supports or works with standard communication protocols. Throughout the world, communication protocols such as ASTM and HL7 are gaining popularity among many medical instruments and suppliers, and many local or small businesses still operate with proprietary communication protocols due to cost efficiency and ease of communication, while meeting basic requirements for data exchange [5]. To develop a universal medical device, or as far as possible, support a medical device integration system, it is necessary to understand the standard interfaces and communication protocols. This document is intended to provide a brief study needed to develop a medical device integration system.

2 Proposed System

The proposed system is to develop a multi-functionality and multi-connectivity LIS integration software to cover different aspects of the integration. This research study is to compile and to give brief idea about different component from physical connectivity to logical configuration in LIS to machine integration. It covers a standard connection methodology and standard communication protocol involved in interface. This research paper is intentionally written to guide on-site support engineer, system integrator, and LIS software developer. Finally, it covers the required software tools and process flow, which should be followed while adding any new machine in integration set. The following are list of instruments at different location and type of physical connectivity, which were studied and successfully configured in order to complete this research.

3 Physical Connections

Medical instrument allows host connectivity through different communication medium like RS232, RJ45, or USB. These three primary media can be extended

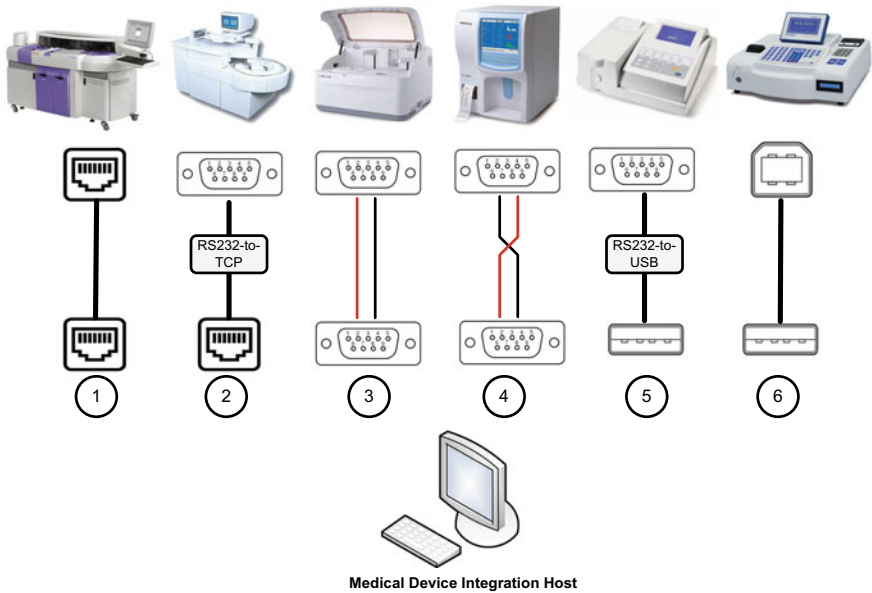


Fig. 1 Physical connectivity model for medical analyzers

to multiple combination using different converters like RS232-to-TCP or RS232-to-USB. There is no specification or restriction over using any of the combination, and this totally depend on the developer’s deployment model and supported software connection modes. If host software supports, it may communicate with instruments connected over RS232, USB, or LAN simultaneously.

Figure 1 illustrated the different possible combination of physical connectivity established between medical instrument and medical device integration host. Further, every combination explained in brief in numbered those are as follows:

- (1) RJ45 LAN connector on both instrument and medical device integration host end. Its usage TCP/IP connection for communication. Its usage network communication protocol and may connect multiple machines over LAN connectivity and it has higher wiring distance range and through right network equipment instruments can be connected to host machine over Wi-Fi network.
- (2) Medical instrument with serial port connection media can be connected over LAN network, and instrument can be converted in to TCP/IP server device using RS232-to-TCP server convertor device. This will allow connecting serial device over network wide avoiding distance limitations.

RS232 is widely used in medical instruments for communication. Using RS232 serial cable, instruments can be directly connected to medical device integration host computer if host supports RS232 port. This cable normally comes in two combinations with or without handshake, support and those are

- (3) Straight/parallel cable with Rx1-to-Rx2 and Tx1-to-Tx2 connection.

- (4) Crossover cable with Rx1-to-Tx2 and Tx1-to-Rx2 connection
- (5) Since still RS232 connector available in wide medical instruments, with innovations and up-gradations in computer system, modern computer delivers with USB port and RS232 is not available. This makes difficult to connect instruments with the host system. Using RS232-to-USB convertor cable, RS232-based instrument can be connected with host over USB connector. This convertor will allow same functionality as RS232 port only difference in connector.
- (6) Like RS232 connector, USB type B port provides similar serial communication over 4-wire USB cable. Both RS232 and USB work on similar principle of logical communication port with configuration like COM port number, baud rate, bit rate, and parity bit. Using simple USB, A-B type printer cable instruments can be connected to medical device integration host computer.

4 Software Integration

LIS host machine or integration software is primarily responsible to capture or demand records from analyzer and push these records to hospital information system (HIS) or laboratory information management system (LIMS) for reporting and storage purpose. LIS integration could be a part of the HIS or LIMS. Considering the LIS integration as a separate software or the module, Fig. 1 illustrates the possible way of deployment to create interfacing between LIS integration and LIMS or HIS. As illustrated in LIS integration with HIS or LIMS having six primarily interfacing deployment methodology (see Fig. 2). Brief description for every type would be as follows.

LIS host could directly push the captured record from analyzer to HIS/LIMS database so that HIS/LIMS can access it [6]. In this scenario, LIS integrator needs access to HIS/LIMS database through direct database connection. HIS develop and LIS integrator need to mutually finalize the middleware table structure to store records. This model would be best solution in local network but feasible only if same vendor is working on either end module as it may expose the database.

Some LIS integration vendors provide shared DLL, library classes, or SDK to have access over LIS local record storage database and other functionality [7]. This method is safest way to share the records with HIS/LIMS, but this integration could be implemented at the development time only and post development this is not feasible option; secondly this option is more feasible only if the HIS/LIMS is desktop-based application and needs multiple Web permission while using in Web interface model.

Like LIS integrator can have an access to HIS/LIMS database, the same way HIS/LIMS could have access to LIS local record storage database to get the desired test record [8, 9]. In this scenario, LIS integrator needs to share the local database access to HIS/LIMS developers to have an access over the local database in local intranet. This method is quite safe as LIS local storage do not hold confidential information and only host result records. This same access can be given to remote

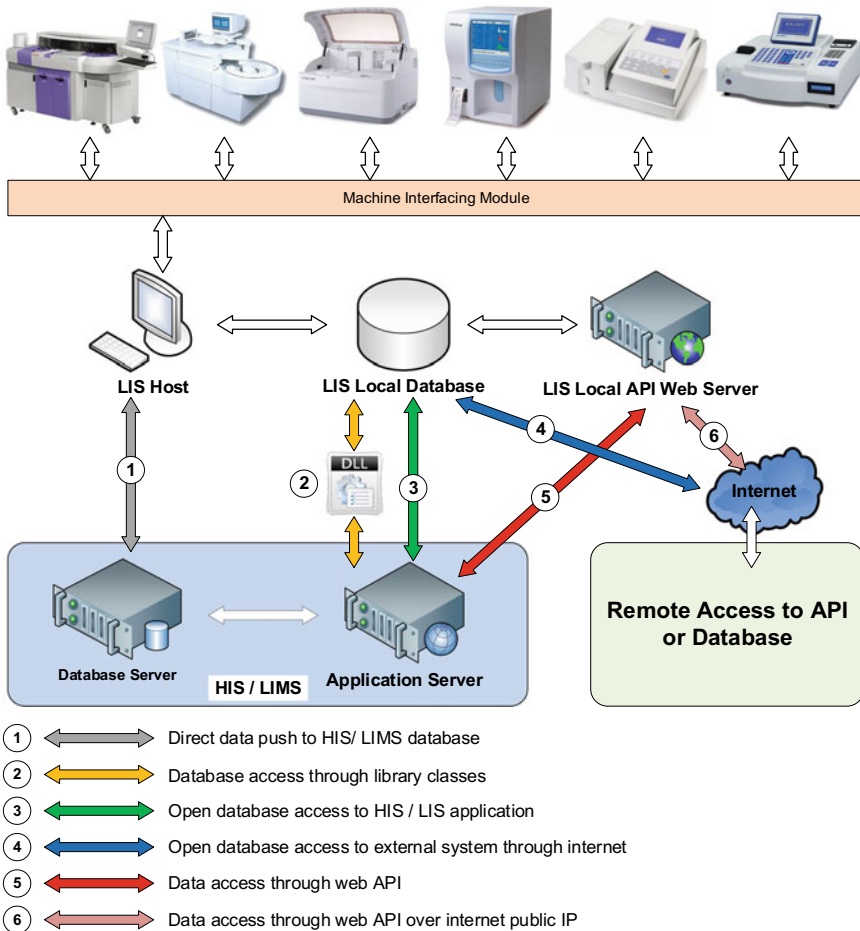


Fig. 2 LIS to HIS possible communication models

application through Internet public IP and TCP/IP-based socket connection with DBMS [9].

Like SDK or class libraries, database access could be given to external HIS/LIMS like application through Web-based API [10, 11]. This is another safe and vendor independent deployment methodology where HIS/LIMS developers can have the access through LIS integration provided Web API. The best part of this type of implementation is to give access of local LIS database and functionality to multiple application concurrently through Web service. These Web API can be served over Internet public IP for remote application [11].

5 Software Modules

LIS host machine or integration software is primarily responsible to capture or demand records from analyzer and push these records to hospital information system) or laboratory information management system (LIMS) for reporting and storage purpose. LIS integration could be a part of the HIS or LIMS. This section primarily discusses the programmatic functionality required for implementation of the LIS integration. This section could directly help LIS integration developer in software implementation. LIS implementation may consist following modules discussed in Table 1.

Primarily, medical analyzer supports two communication protocols where first is ASTM and second is HL7. Both are specifically designing keeping different medical information formation. This information is mainly related to test performed of specific parameters. Both protocols having large scope of explanation and not covered in this paper. Both protocols need specialized data parser to encode and decode data exchange format information packet.

6 Implementation

As an implementation, we selected a TCP-IP-based analyzer machines as it gives better speed and long-distance communication with the machine. Proposed software works as a server, and the analyzer plays client role. Server is always in listen mode, and analyzer connects to the server software as and when available. Normally, after completion of any event like sample received, sample processed, sample failed, calibration error, etc., analyzer generates a record and transmit it to connected server. It is a server software responsibility to received, decode, and as defined on specific record. In case of receiving successfully received result, server software push it in storage of defined format, and primarily, this storage is in the form of database table. We used SQL server as a database for storing the result records of the test parameters. For decoding process, developers need to refer the analyzer LIS manual as every machine may have different placement for different data field in the records. As per the standards, the separators and the construction format support standard and globally accepted field related to medical test. Server software GUI illustrated (see Fig. 3) the running mode of the software with server parameters information like connection status, server IP and port number, analyzer name, and the setting options button. Typically, these software runs in background and having very less user interaction controls. Another GUI screen illustrated the record fields parsed and stored in database tables in predefined format (see Fig. 4).

Table 1 LIS integration software modules

Module	Description
Connectivity and communication tester	In order to add any new machine interfacing in LIS, it is important to ensure the connectivity and communication with analyzer. Secondly, this tool could be used for capturing sample records or data from analyzer and to understand the record format
Analyzer configurator	Laboratory with multiple analyzer needs configurator and management utility
Database connectivity manager	LIS integration may support multiple type of DBMS and record storage. This utility helps integrator to manage DBMS connectivity parameters and configuration settings
Data communicator	Most important module in LIS integration to establish and control data exchange between the analyzer and the LIS host. It is suggested to develop this with minimum real-time refreshing GUI component, as it requires highest CPU load in order to manage multiple analyzer data exchange
Data parser	Every record received from analyzer is in defined format and separated with special characters. These records need to decode or parse in different fields or token, hence parsing is the most important module in LIS integration
API manager	If LIS integration supports the API access, this utility may assist in configuring and sharing the API functions and the parameter along with API documentations
Record reporting	All the records captured and stored in LIS local database needs a reporting tool in order to understand or check the desired records from different patients, samples, test, or the machines. This could also help to find out if HIS/LIMS fails to get the desired record is really got captured or not
Log monitor	This tool could help developer for understanding the different application logs like performance, error, or the activity log. Log can be viewed from log files but preparing the log file viewer may help developer in better way

7 Conclusion

Proposed study has wide range of applications and covered wide range of medical analyzer communication methodology. Understanding the LIS integration needs detailed study of individual communication protocols, and this software needs large set of trials and errors. Right storage and analytics of these laboratory data may

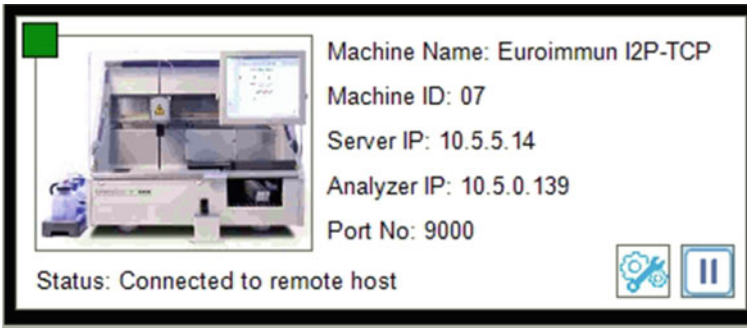


Fig. 3 Server software GUI

The screenshot shows a window titled 'SearchData' with search filters for 'Sysmex XS 500-01', 'LIS Machine', and '11-Jun-2021'. Below the filters is a table with columns: Sample ID, Param ID, Value, Unit, Date Time, and Machine. The table contains 14 rows of data.

Sample ID	Param ID	Value	Unit	Date Time	Machine
QC-10850812	WBC	16.97	10 ³ /uL	2021/06/10 17:09:44	Sysmex XS 500-01
QC-10850812	RBC	5.27	10 ⁶ /uL	2021/06/10 17:09:44	Sysmex XS 500-01
QC-10850812	HGB	16.2	g/dL	2021/06/10 17:09:44	Sysmex XS 500-01
QC-10850812	HCT	47.8	%	2021/06/10 17:09:44	Sysmex XS 500-01
QC-10850812	MCV	90.7	fL	2021/06/10 17:09:44	Sysmex XS 500-01
QC-10850812	MCH	30.7	pg	2021/06/10 17:09:44	Sysmex XS 500-01
QC-10850812	MCHC	33.9	g/dL	2021/06/10 17:09:44	Sysmex XS 500-01
QC-10850812	PLT	497	10 ³ /uL	2021/06/10 17:09:44	Sysmex XS 500-01
QC-10850812	NEUT%	45.2	%	2021/06/10 17:09:44	Sysmex XS 500-01
QC-10850812	LYMPH%	27.1	%	2021/06/10 17:09:44	Sysmex XS 500-01
QC-10850812	MONO%	8.8	%	2021/06/10 17:09:44	Sysmex XS 500-01
QC-10850812	EO%	11.3	%	2021/06/10 17:09:44	Sysmex XS 500-01
QC-10850812	BASO%	7.6	%	2021/06/10 17:09:44	Sysmex XS 500-01

Fig. 4 Database stored records field structure

help medical researcher understanding the levels of test performed at particular location for particular dieses for particular age group and particular gender. This study has large scope of improvement and level of knowledge acquisition since every machine works differently and respond differently. Every test has different method of processing, and every parameter has different way of interpretation and references ranges.

References

1. Baghal A, Al-Shukri S, Kumari A (2019) Agile natural language processing model for pathology knowledge extraction and integration with clinical enterprise data warehouse. In: 2019 Sixth international conference on social networks analysis, management and security (SNAMS). <https://doi.org/10.1109/snams.2019.8931828>
2. Bender D, Sartipi K (2013) HL7 FHIR: An Agile and RESTful approach to healthcare information exchange. In: Proceedings of the 26th IEEE international symposium on computer-based medical systems. <https://doi.org/10.1109/cbms.2013.6627810>
3. Chen M, Hao Y, Hwang K, Wang L, Wang L (2017) Disease prediction by machine learning over big data from healthcare communities. *IEEE Access* 5(1):8869–8879

4. Qian B, Wang X, Cao N, Li H, Jiang Y-G (2015) A relative similarity-based method for interactive patient risk prediction. *Data Mining Knowl Discov* 29(4):1070–1093
5. Saripalle R, Runyan C, Russell M (2019) Using HL7 FHIR to achieve interoperability in patient health record. *J Biomed Inform* 94:103188. <https://doi.org/10.1016/j.jbi.2019.103188>
6. Kilintzis V, Kosvyra A, Beredimas N, Natsiavas P, Maglaveras N, Chouvarda I (2019) A sustainable HL7 FHIR based ontology for PHR data*. In: 2019 41st Annual international conference of the IEEE engineering in medicine and biology society (EMBC). <https://doi.org/10.1109/embc.2019.8856415>
7. Houta S, Ameler T, Surges R (2019) Use of HL7 FHIR to structure data in epilepsy self-management applications. In: 2019 International conference on wireless and mobile computing, networking and communications (WiMob). <https://doi.org/10.1109/wimob.2019.8923179>
8. Gomes F, Freitas R, Ribeiro M, Moura C, Andrade O, Oliveira M (2019) GIRLS, a Gateway for Interoperability of electronic health record in low-cost system: * interoperability between FHIR and OpenEHR standards. In: 2019 IEEE international conference on e-health networking, application & services (HealthCom). <https://doi.org/10.1109/healthcom46333.2019.9009602>
9. Mukhiya SK, Rabbi F, Pun KI, Lamo Y (2019) An architectural design for self-reporting e-health systems. In: 2019 IEEE/ACM 1st international workshop on software engineering for healthcare (SEH). <https://doi.org/10.1109/seh.2019.00008>
10. Corici AA, Olaf R, Kraufmann B, Billig A, Caumanns J, Deglmann M, Walter V, Rexin J, Nolte G (2020) Interoperable and discrete eHealth data exchange between hospital and patient. In: 2020 23rd Conference on innovation in clouds, internet and networks and workshops (ICIN). <https://doi.org/10.1109/icin48450.2020.9059335>
11. Khalique F, Khan SA, Nosheen I (2019) PHF-A framework for public health monitorin, analytics and research. *IEEE Access*, pp 1–1. <https://doi.org/10.1109/access.2019.2930730>

Framework for the Integration of Transmission Optimization Components into LoRaWAN Stack



Bruno Mendes, Shani du Plessis, Dário Passos, and Noélia Correia

Abstract The Internet of things has grown in recent years, and new applications are now emerging, many requiring long-range coverage and low energy consumption while operating on low data rates. These requirements have driven the emergence of new technologies, like low power wide area networks. LoRa wide area network is one of these technologies that operate on the unlicensed frequency band and is extremely customizable. More specifically, its parameters can be set to increase the quality of packet transmission, by increasing the time on air, at the expense of bandwidth. This also leads to an increase in power consumption. Therefore, in order to increase the data rate and save energy, optimization procedures, which seek to dynamically adjust the airtime, should be used. The goal of this chapter is to architect an optimization agnostic framework for ChirpStack, which is an open-source LoRa wide area network server stack, for the incorporation of any optimizer, e.g., learning agent, aiming to adapt LoRa transmission parameters to the current network scenario. The framework includes a Handler that waits for frame information from the devices, filters relevant data and places it in a broker, and includes a subscriber that gets optimization procedure results from a subscribed topic at the broker and converts them to an acceptable downlink format.

Keywords Internet of things · LPWAN · LoRaWAN · ChirpStack · Open source

This work is supported by Fundação para a ciência e Tecnologia within CEOT (Center for Electronic, Optoelectronic and Telecommunications) and the UID/MULTI/00631/2020 project.

B. Mendes (✉) · S. du Plessis · D. Passos · N. Correia
CEOT, University of Algarve, Faro, Portugal
e-mail: bemendes@ualg.pt

S. du Plessis
e-mail: sduplessis@ualg.pt

D. Passos
e-mail: dmpassos@ualg.pt

N. Correia
e-mail: ncorreia@ualg.pt

N. Correia
Faculty of Science and Technology, University of Algarve, Faro, Portugal

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
H. Sharma et al. (eds.), *Communication and Intelligent Systems*, Lecture Notes
in Networks and Systems 461, https://doi.org/10.1007/978-981-19-2130-8_34

1 Introduction

Low power wide area network (LPWAN) technologies enable massive low bit rate wireless connections, covering long distances with minimum power consumption and maintenance [1]. One of these LPWAN technologies is the narrowband Internet of things (NB-IoT), which is inherited from cellular communication, working in licensed frequency bands [2]. Another one is LoRaWAN technology that operates in the unlicensed frequency band, so that end users are free to build up LoRa-based architectures similar to house-owned WiFi routers [3]. This freedom will contribute to the emergence of new IoT applications.

The large-scale deployment of LPWAN for IoT brings along many challenges because radio resources are scarce and their management becomes very challenging in practical networks. Therefore, procedures for the efficient and dynamic management of resources become necessary. In the case of LoRaWAN networks, an adaptive data rate (ADR) mechanism that is able to dynamically assign transmission parameters to the end nodes is available. However, this approach has shortcomings and open challenges, as stated in [4], which can be tackled by using other techniques. An example of such shortcomings is the convergence of ADR, which is slow in certain scenarios and, moreover, these adjustments are done based on past information, which ends up being inefficient when the environment is dynamic. This has led to the emergence of more efficient ways to optimize transmission parameters, and intelligent ways to disseminate configurations. The scarcity of radio resources and short transmission duty cycle (TDC), of both gateways and end nodes, are the main obstacles that these approaches face. However, these works limit themselves to the optimization techniques and do not discuss their integration in an open-source LoRaWAN stack, for real deployment.

In this chapter, an optimization agnostic framework for ChirpStack, an open-source LoRaWAN Network Server stack, is architected, so that mechanisms for the optimization of transmission parameters can easily be incorporated. The proposed framework is analyzed and evaluated in every workflow step, and its feasibility to plugin optimization decisions into LoRaWAN networks is shown.

The remainder of this chapter is organized as follows. In Sect. 2, recent approaches for LoRaWAN resource management are presented. Section 3 discusses the main components of ChirpStack, the open-source LoRaWAN Network Server stack, including its available integration facility. Section 4 presents the optimization agnostic framework, while Sect. 5 discusses deployment tests performed. Section 6 concludes the chapter and points to work ahead.

2 Related Work

Radio resource management (RRM) is a fundamental problem in all wireless communications. In the case of LoRaWAN standard, a simple rate adaptation mechanism is specified, but its implementation is left for manufacturers [5]. However, its current

de facto implementation has the shortcomings of having a slow convergence in certain scenarios and making adjustments based on the past, which may not be adequate for the future (no learning exists) [4, 6]. Proposals have emerged recently to better address the optimization of transmission parameters. In [7], the authors optimize the communication parameters using reinforcement learning (RL), so that the accumulated per-node throughput increases. In [8, 9], the authors investigate LoRaWAN frame collisions and consider different scenarios to gain better insight into collisions within real networks, while in [10] a RL approach is proposed for agents to learn how to assign orthogonal channels, thereby addressing collisions. RL is now a hot research topic in the development of self-driving networks where the idea is to learn rather than to design the network management solutions [11].

3 ChirpStack: The Open-Source LoRaWAN Network Server Stack

3.1 Main Components

ChirpStack is an open-source LoRaWAN stack made of three main components that ensure LoRa packet delivery from the device till the end user: (i) Gateway Bridge; (ii) Network Server; and (iii) Application Server. These are illustrated in Fig. 1. APIs are provided for communication with the device and other purposes, as discussed below.

3.1.1 Gateway Bridge

The main job of this component is to convert the packet data received from the gateway, having a Semtech UDP packet-forwarder format, to a data serialization method like protobuf or json. The resulting structure is then published on an MQTT broker under a topic that defines the type of message that was sent from the device: an `uplink` message or a `join` request from the device, or a `stat` update from the gateway. The Network Server subscribes to such topics.

3.1.2 Network Server

This component works as an intermediate between the gateway and the Application Server. It provides a packet verification mechanism to ensure packet uniqueness. That is, only non-duplicated packets are sent to the Application Server.

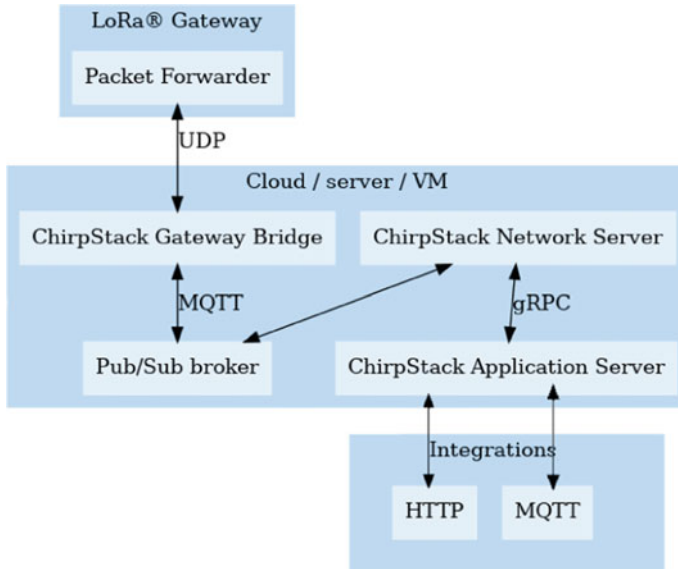


Fig. 1 ChirpStack components. *Source* [12]

3.1.3 Application Server

This component handles the `join` requests and payload encryption. Additionally, it offers a user interface (UI) to easily configure and manage the whole LoRaWAN stack. An API is provided that allows access to the whole LoRaWAN inventory.

3.2 Available Integration Facility

The ChirpStack Application Server offers a mechanism for integration with user applications. More specifically, events can be forwarded to user application endpoints. The type of event and URL endpoint are configurable. When operating this way is ensured that the Application Server will send all the `uplink` messages to the user application. This mechanism will be used to collect all the information that is relevant for the optimization of communications, as discussed in the following section.

4 Optimization Agnostic Framework

4.1 Motivation and Overall Architecture

The available integration facility of ChirpStack allows sandbox software development, or software agent training with subsequent deployment of learned configurations. In fact, an optimization agnostic framework can be integrated into ChirpStack, as long as relevant data from the LoRaWAN is made available, and configuration activation channels are provided. Any kind of intelligent adaptation can be incorporated for the optimization of communications, ultimately determining the feasibility of applications in the face of scarce radio resources.

The optimization framework, to be integrated into ChirpStack, has two main parts: (i) the Handler, a server that waits for frame information from the devices, via the ChirpStack API, filters relevant data and acts as an MQTT client to place such relevant data in an MQTT broker; (ii) the Subscriber, a MQTT client that subscribes to topics from the same MQTT broker, topics resulting from optimization procedures, converting them to an acceptable downlink format.

The above-mentioned optimization framework is shown in Fig. 2. When starting, the Handler process starts listening to incoming packets, from ChirpStack API, while simultaneously, the Subscriber process subscribes to an optimizer related topic in the MQTT broker. More specifically, the workflow can be summarized as follows:

1. The Handler listens on a predefined port for packets arriving from the ChirpStack API. The Handler processes and filters each packet, extracting just the fields that are valuable for time on air (ToA) optimization.
2. A message with the extracted information is published to the MQTT broker under Topic A.
3. The optimizer subscribing to Topic A collects data from the MQTT broker.
4. Optimizer determines the best communication parameters and publishes results under Topic B.
5. The MQTT Subscriber gets Topic B notifications.
6. Topic B notifications are converted to downlink message format and sent to ChirpStack API. Such messages are placed in the downlink queue and will be sent to the end node within the defined time frame, e.g., after receiving the next uplink in case of Class A devices.

4.2 The Handler

This component is responsible for receiving the information from the ChirpStack API, filtering the relevant data and sending it to the MQTT broker. The pseudocode of the Handler is shown in Algorithm 1. This has as a basis the code provided in [13].

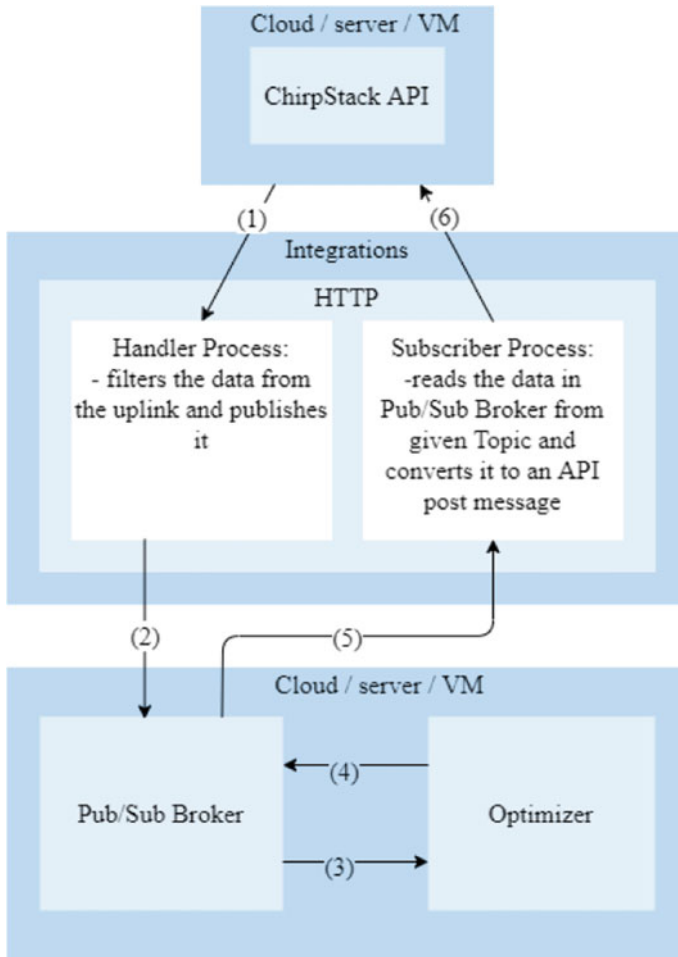


Fig. 2 Proposed optimization framework for integration into ChirpStack

The Handler acts by listening to POST requests from the Application Server, and by unmarshalling the data. In order to extract data relevant for optimization purposes, and information required to create a downlink in the API, a filter is implemented. For the downlink, `dev_eui` and the `f_port` are required, and for this reason, these are included in the data structure. Data has to be encoded in Base64 format for publishing to a MQTT broker under a specific topic, to which the optimization algorithm will subscribe.

It is important to note that the alternative would be to subscribe at the MQTT broker ensuring Gateway Bridge to Network Server communication, but this requires implementing all Network Server functionalities, like checking for duplicated packets, which is not feasible in practice.

```

1 Input: post_msg, dst_ip, dst_port, topic
2 Output: b64_msg


---


3 while TRUE do
4   if post_msg & post_msg is uplink then
5     unmarshalled_msg = UNMARSHALL(post_msg)
6     /* converting bytes to json object */
7     mqtt_msg = {
8       "dev_eui" : unmarshalled_msg.dev_eui,
9       "crc" : unmarshalled_msg.code_rate,
10      "sf" : unmarshalled_msg.spreading_factor,
11      "bw" : unmarshalled_msg.bandwidth,
12      "f_port" : unmarshalled_msg.f_port,
13      "lora_snr" : unmarshalled_msg.lora_snr
14    }
15    b64_msg = B64_ENCODE(mqtt_msg)
16    client = MQTT_CLIENT(dst_ip,dst_port)
17    client.PUBLISH(topic,b64_msg)
18    client.CLOSE()
19  end
20 end

```

Algorithm 1: Pseudocode of the Handler.

4.3 The Subscriber

This component is responsible for subscribing to notifications placed at the MQTT server by the communications optimizer. Besides information regarding parameters to be configured at the end nodes, such notifications include the `device_eui` and `f_port` required when building the downlink. Such notifications must be decoded from the Base64 format to the API downlink structure, for further POST to the ChirpStack’s Application Server API. The pseudocode of the Subscriber is shown in Algorithm 2.

4.4 The Optimizer

LoRa is a spread spectrum modulation technique derived from chirp spread spectrum (CSS) technology. LoRa stands out as an LPWAN mainly due to the use of unlicensed frequency bands, allowing complex LPWANs to be built at affordable prices. Taking advantage of LoRa and LPWAN architecture, the LoRaWAN concept appeared, allowing long-range (~15 kms) and low power reliable networks to be built. The lifespan of the battery mainly depends on the time on air (ToA), which defines the elapsed time for a LoRaWAN packet to travel between an end device and the gateway, and duty cycle, which is the fraction of one period in which a signal or system is active, required for answers from the Network Server to arrive. For a specific payload

```

1 Input: b64_msg, dst_ip, dst_port, topic, chirpstack_api_server
2 Output: api_msg


---


3 client = MQTT_CLIENT(dst_ip, dst_port)
4 client.SUBSCRIBE(topic)
5 while client is subscribed do
6   msg_rcv = client.RECEIVE_MSG()
7   msg_decoded = B64_DECODE(msg_rcv)
8   downlink_data = {
9     "crc": msg_decoded.crc,
10    "sf": msg_decoded.sf,
11    "bw": msg_decoded.bw
12  }
13  b64_downlink_data = B64_ENCODE(downlink_data)
14  api_msg = {
15    "deviceQueueItem": {
16      "dev_eui": msg_decoded.dev_eui,
17      "f_port": msg_decoded.f_port,
18      "data": b64_downlink_data
19    }
20  }
21  POST(chirpstack_api_server, api_msg)
22 end

```

Algorithm 2: Pseudocode of the Subscriber.

size, the ToA will be determined by: (i) spreading factor (SF), bandwidth (BW) and coding rate (CR). More specifically, the ToA will be:

$$\text{ToA} = \frac{2^{\text{SF}}}{\text{BW}} \times \left[8 + \max \left(\left\lceil \frac{8\text{PL} - 4\text{SF} + 28 + 16 - 20\text{H}}{4(\text{SF} - 2\text{DE})} \right\rceil \times (\text{CR} + 4), 0 \right) \right] \quad (1)$$

where PL is the packet payload, H is the header implicit flag and DE a low data rate optimization flag. The signal to noise ratio at the receiver depends not only on these parameters, but also on the environment in which devices are placed and communication distance. Thus, the best SF/BW/CR combination has to be dynamically optimized according to changes in the environment and distance.

5 Deployment Tests

5.1 The IoT System

The developed IoT system used to perform tests includes the following key devices: (i) A machine running Ubuntu 20.04LTS containing all the ChirpStack-related software, such as PostgreSQL, MQTT, ChirpStack modules and the proposed framework; (ii) a Raspberry Pi 3 using Raspbian (Buster) with a LoRa Hat where Semtech

Packet Forwarder is installed; (iii) an Arduino MKRWAN 1310 (with DHT11 digital temperature/humidity sensor) as the end device that sends sensor data and listens to downlinks, in order to update the LoRa communication parameters. The end device is implemented as a Class A device supporting bidirectional communication with the gateway, where device to gateway communication can be done at any time while gateway to device communication can only occur in a small window (after the uplink).

5.2 Framework in Operation

To confirm the successful operation of the proposed framework, all workflow from the ChirpStack Application Server until the Optimizer, and vice-versa, was tested. Message integrity is ensured at every intermediate step of the communication between these endpoints.

Implementation and tests were carried out using Python, also for MQTT client and broker implementations. A configuration JSON file is adopted to store MQTT broker, Topics and ChirpStack API-related information. JSON Web Token (JWT) is used to securely transfer JSON objects to the ChirpStack API, those including configuration parameters resulting from the Optimizer. The tested workflow is as follows:

- The framework starts by listening to the IP address and port, specified in the configuration file, and subscribes at the MQTT Broker the Optimizer related Topic. See Fig. 3a.
- A script was developed to build LoRa packets, having a Semtech Packet Forwarder format, to be sent to ChirpStack Network Server. The Network Server verifies the packet and sends in a POST request to the Handler. See Fig. 3b.
- The Handler filters packet payloads and publishes the relevant data at the MQTT Broker, properly encoded. This is shown in Fig. 3c. Figure 3d confirms the successful publishing at the Broker.
- The `mosquitto-clients` library was used on a Ubuntu 20.04 machine for the publishing of configurations by the Optimizer. Figure 3e shows a Base64 encoded message being published by the Optimizer.
- Messages published by the Optimizer are received by the Subscriber and converted to an API POST message to be delivery to the ChirpStack API. This is shown in the capture at Fig. 3f.
- It was also possible to confirm the arrival of messages in the downlink queue of the chosen device, as shown in Fig. 3g.

In summary, all the workflow from ChirpStack to the Optimizer, and vice-versa, is working properly, allowing for LoRaWAN network optimization using any algorithm or learning agent.


```

Handler waiting for Chirpstack API Post messages on port 8000
Subscribed: Topic B at 192.168.1.9:1883

```

(a) Framework initiation.

```

Sending Lora pkts to:
Network Server: 192.168.1.9:1700
pkt sent at Thu Apr 15 2021 10:03:37 GMT+0100 (British Summer Time)

```

(b) Packet delivery to the Network Server.

```

Uplink received from a8610a34343c6613 with payload: 0167013f018806
765ff2960a0003e8

MQTT Publish Message:
{'dev_eui': 'a8610a34343c6613', 'crc': '4/6', 'sf': 7, 'bw': 125,
'f_port': 1, 'lora_snr': '5.1'}

Packet published to Topic A in 192.168.1.9:1883 at 2021-04-15 10:03:
37.46225

```

(c) Handler listening to POST requests and publishing at the MQTT Broker.

```

$ mosquitto_sub -t «Topic A»
eyJkZXZfZXVpIjogImE4NjEwYTM0MzQzYzY2MTMlLCAiY3JlIjogIjQvNiIsICJzZiI
6IDcsICJldyI6IDEyNSwgImZfcG9ydCIDEsICJsb3JhX3NuciiI6ICI1LjEifQ==

```

(d) Successful publishing at the MQTT Broker, by the Handler.

```

$ mosquitto_sub -h 127.0.0.1 -m «eyJhZGV2V1aSI6ICJhODYxMGEzNDM0M2M2
NjEzIiwgImZqb3J0IjoxLCAiY3Y3IjogIjQvNSIsImJ3IiA6IDUwMCwic2YiDogIDE
xLjCzbnIiOiAiMjEifQ==» -t «Topic B»

```

(e) Optimizer publishing at the MQTT Broker.

```

Message read from Topic B at 2021-04-15 10:12:30.825746
-----
Post message sent to Chirpstack API at 2021-04-15 10:12:30.826718
Message format:
{'deviceQueueItem': {'confirmed': True, 'data': 'eyJjcmMiOiAiAinc81Ii
wgImJ3IjogNTAwLCAic2YiOiAxMSwgInNuciiI6ICIyMSJ9', 'devEUI': 'a8610a3
4343c6613', 'fCnt': 0, 'fPort': 1}}
<Response [200]>

```

(f) Subscriber receiving notification and corresponding API POST message.

Downlink queue			
Fcnt	Fport	Confirmed	Base64 encoded payload
24	1	yes	eyJjcmMiOiAinc81IiwgImJ3IjogNTAwLCAic2YiOiAxMSwgInNuciiI6ICIyMSJ0I

(g) Downlink queue.

Fig. 3 Screenshots of the tested workflow

6 Conclusions and Future Work

This chapter proposes an optimization agnostic framework for ChirpStack, an open-source LoRaWAN Network Server stack, so that decisions from optimizers can be incorporated. These optimizers decide for the best communication parameters, which leads to better data rates and energy saving. As far as our literature research suggests, previous authors mainly discuss such optimization approaches, e.g., learning agents, but there is no mention to how these would be integrated in a LoRaWAN stack, for a real deployment. This chapter aims to fill this gap. The framework architecture and implementation are discussed, and all framework operations are analyzed and evaluated. Results show that every workflow step operates properly, meaning that such framework is a feasible solution to plugin optimization decisions into LoRaWAN networks. Future work includes testing different optimizers.

References

1. Song Y, Lin J, Tang M, Dong S (2017) An internet of energy things based on wireless LPWAN. *Engineering* 3(4):460–466
2. Ballerini M, Polonelli T, Brunelli D, Magno M, Benini L (2020) Nb-iot versus lorawan: an experimental evaluation for industrial applications. *IEEE Trans Ind Inf* 16(12):7802–7811
3. Bäumker E, Garcia A, Woias P (2019) Minimizing power consumption of lora ® and lorawan for low-power wireless sensor nodes. *J Phys Conf Ser* 1407:012092
4. Kufakunesu R, Hancke GP, Abu-Mahfouz AM (2020) A survey on adaptive data rate optimization in lorawan: Recent solutions and major challenges. *Sensors* 20(18)
5. Semtech Corporation (2016) Lorawan simple rate adaption recommended algorithm. Technical report
6. Li S, Raza U, Khan A (2018) How agile is the adaptive data rate mechanism of lorawan? In: 2018 IEEE global communications conference (GLOBECOM), pp 206–212
7. Sandoval RM, Garcia-Sanchez A-J, Garcia-Haro J (2019) Optimizing and updating lora communication parameters: a machine learning approach. *IEEE Trans Netw Serv Manage* 16(3):884–895
8. Carvalho R, Al-Tam F, Correia N (2021) Q-learning adr agent for lorawan optimization. In: 2021 IEEE International conference on industry 4.0, artificial intelligence, and communications technology (IAICT), pp 104–108
9. Rahmadhani A, Kuipers F (2018) When lorawan frames collide. In: Proceedings of the 12th international workshop on wireless network testbeds, experimental evaluation & characterization, WiNTECH '18, New York, NY, USA. Association for Computing Machinery, pp 89–97
10. Aihara N, Adachi K, Takyu O, Ohta M, Fujii T (2019) Q-learning aided resource allocation and environment recognition in LoRaWAN with CSMA/CA. *IEEE Access* 7:152126–152137
11. Al-Tam F, Correia N, Rodriguez J (2020) Learn to schedule (LEASCH): A deep reinforcement learning approach for radio resource scheduling in the 5G MAC layer
12. Brocaar O, Chirpstack architecture—chirpstack open-source lorawan® network server
13. Brocaar O, Http—chirpstack open-source lorawan network server

Design of Low-Power Parallel Prefix Adder Templates Using Asynchronous Techniques



J. Sudhkar and E. Jagadeeswara Rao

Abstract This paper aims to design low-power circuits like an adder, buffer, AOI, and logic gates using asynchronous quasi delay insensitive (QDI) templates, essential for many arithmetic computations. Adder is a fundamental building block for applications like ALU, microprocessors, and digital signal processors. The present trendy parallel prefix adder KSA is modeled and verified with various asynchronous QDI templates. The prominent templates include pre-charged half buffer, autonomous signal validity half buffer, and sense amplifier half buffer in dual-rail encoding style. Due to clock circuitry, synchronous circuits determine more switching activity, which dissipates more power. This drawback is overcome through clock-less circuits, which dissipate less power by reducing the switching activity without degrading the functionality of a circuit. An asynchronous circuit has various timing approaches such as QDI, delay insensitive, and bundled data. Still, the QDI approach has significant advantages in power dissipation, delay, and energy savings. The major drawback of QDI templates is the completion detector block, which dissipates more power and occupies a large area overhead. To overcome this drawback, an advanced QDI template—sense amplifier half buffer is designed to provide low power, less delay with efficient energy due to the utilization of sub-threshold process and controlled reset transistor in evaluation block and the absence of completion detector circuit. The paper describes the performance aspects of 32-bit KSA using various QDI templates in terms of multiple metrics like power, delay, and energy using the mentor graphics EDA tool.

Keywords Clock-less circuits · QDI templates · Sense amplifier half buffer · Pre-charged half buffer · Autonomous signal validity half buffer · KSA

J. Sudhkar · E. J. Rao (✉)

Department of Electronics and Communication Engineering, Vignan's Institute of Engineering for Women, Kapujagaraju Peta, Visakhapatnam, AP, India

e-mail: emandi.jagadeesh@gmail.com

1 Introduction

Early days, the major problem with designing an integrated circuit (IC) was its size and performance. But nowadays, VLSI designers are more focused on power consumption/dissipation and performance to extend the lifetime of digital devices [1]. The reduction of power dissipation is attained after developing sub-micrometer and nanometer technology devices. Asynchronous approach dominates the industry to design an IC using CMOS logic, but it dissipates more power due to the clocking circuitry. An asynchronous circuit is one of the best approaches that dissipates less power due to the clock-less technique [2]. Self-timed circuits are integrated with some timing assumptions at both input and output sections, which include data insensitive (DI), QDI, and bundled data (BD), as shown in Fig. 1 [3]. These clock-less approaches are designed with dual-rail/quad-rail for data encoding [4].

Among these approaches, QDI technique is the only one delay assumption technique. The other models (DI and BD) are independent of delay assumptions on wires and gates [5, 6]. QDI approach accesses with acknowledgment signal to improve the accuracy of a transition wire. This approach consists of two pipeline stages—data control decomposition (DCD) and integrated latch (IL). An IL structure is more acceptable than the DCD pipeline due to integrating controller and data path in

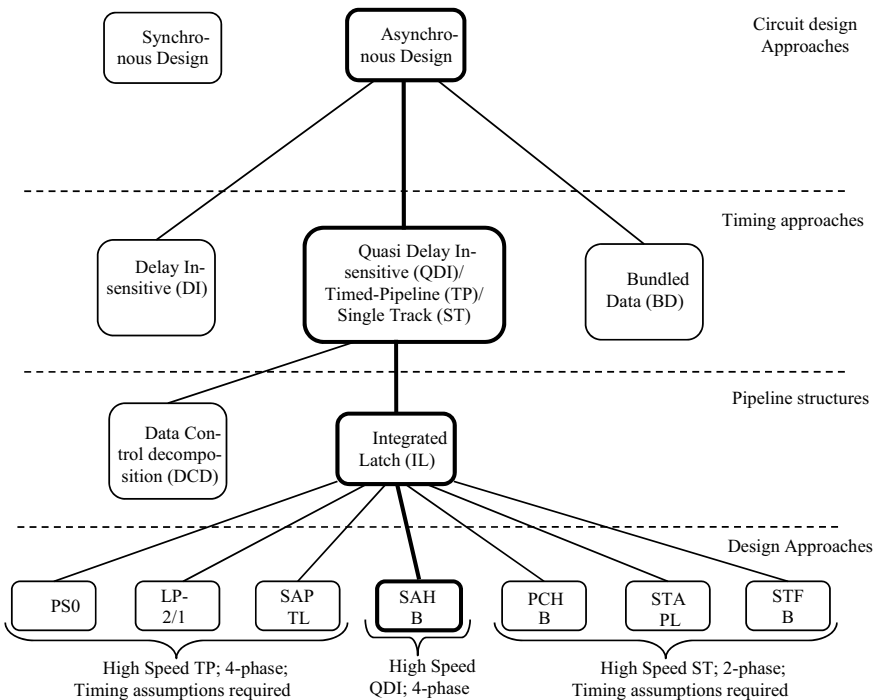


Fig. 1 Classification of digital logic circuits

the mainframe pipeline stage, which attains the shortest critical path [7]. The IL pipeline is classified as cell-based protocols for 2- and 4-phase handshaking protocols. Figure 1 shows the various cell-based design approaches of the QDI template—pre-charged half buffer (PCHB) (moderately), autonomous signal validity half buffer (ASVHB), and sense amplifier half buffer (SAHB). Timed pipeline template—SAPTL [8], PS0, LP2/1, single-track template—STAPL, STFB. The present leading asynchronous QDI techniques are PCHB, ASVHB, and SAHB.

A digital adder is crucial for various arithmetic computations like ALU and MAC units [9]. In VLSI technology, parallel prefix adders are primarily used in arithmetic computations such as microprocessors, digital signal processors (DSPs), and other high-speed applications. These adders reduce the logic complexity, which optimizes the delay, power, and energy of the design. One vital parallel prefix adder is Kogge Stone adder (KSA) [10], which generates carry with high speed compared to other adders. This paper presents the performance evaluation of a 32-bit KSA adder with leading QDI templates such as PCHB, ASVHB, and SAHB.

This paper is organized as follows: Sect. 2 deals with existing QDI templates (PCHB and ASVHB). Section 3 describes the design of KSA with the SAHB cell. Results are depicted in Sect. 4 and conclude the paper with Sect. 5.

2 Preliminary QDI Templates

Synchronous circuits have significant power consumption/dissipation drawbacks due to the clocking approach and complex interconnects. These difficulties are overcome in asynchronous circuits [2], which exploit less power consumption/dissipation and provide better performance due to the absence of a clocking approach. The current leading timing approach of asynchronous logic is QDI [11]. In the QDI approach, isochoric forks are assumed for each wire transition, i.e., change exists with acknowledgment signal [12].

2.1 Half Buffer

The pipeline frame is the half buffer when input and output data rails are influenced by one data token [13]. A frame is a whole buffer when all the input and output lines can hold different data frames simultaneously. An entire buffer cell is more concurrent and complicated to realize than a half buffer cell. Thus, half buffers are primarily used in the QDI approach. There are various cells under half buffer templates, such as PCHB, ASVHB, and SAHB. This paper deals with three common types of half buffer cells: PCHB, ASVHB, and SAHB.

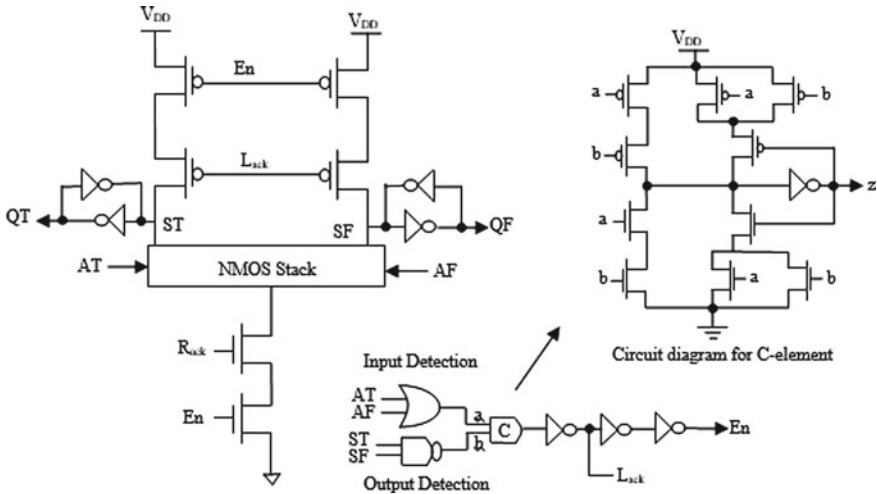


Fig. 2 Gate-level structure of the PCHB buffer cell template

2.1.1 Pre-charge Half Buffer (PCHB)

Generally, PCHB is a 2-phase QDI template that incorporates domino logic [14]. This template operates with completion detectors (input and output) to specify an input as null or data state. The logic circuits are implemented in the evaluation phase and attain output even before all valid inputs. A PCHB template is set to pre-charge mode by lowering (logic ‘0’) the acknowledgment signal [5].

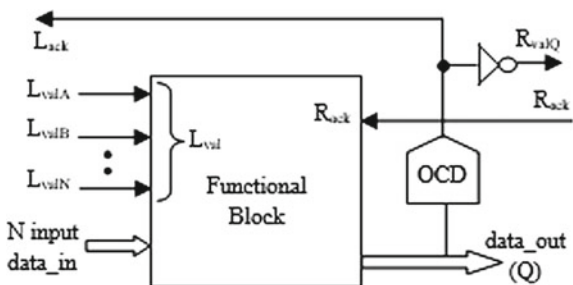
Figure 2 [3] shows the basic gate-level structure of the PCHB buffer cell template, which consists of the pull-up, pull-down, and two weak feedback inverters at the output side. PCHB is an isochoric fork, accessing it with the acknowledgment signals (rack for input and lack for output). E_n is an enable signal integrated with a micro-cell. The lack and E_n signals are generated via completion detectors of input and output signals. These completion detectors are used for validating the availability of input/output signals. But the drawback in this template is dissipating more power due to usage of additional logic gates, including C-element in input completion detector (ICD) and output completion detector (OCD). Thus, another leading QDI approach is introduced, i.e., ASVHB cell, which dissipates less power and better performs by reducing the transistor count than the PCHB template [15].

2.1.2 Autonomous Signal Validity Half Buffer

ASVHB is another leading QDI template for low-power sub-threshold operation. This template requires 3 steps:

1. Utilize an autonomous validity signal to operate the QDI circuit efficiently.
2. The fine-grained gate-level method is used to improve the throughput rate.

Fig. 3 Block diagram of AVSHB cell



- 3. ASVHB design is realized with static logic implementation for sub-threshold operation.

Figure 3 shows a block diagram of the ASVHB cell. It consists of input data_in and output data_out (Q) [15].

The handshaking signals are single-ended—lack and rack. The validity signals for both input and output data signals are (L_valA, L_valB, ..., L_valN, and R_valQ). An output completion detector (OCD) is used to validate the availability of the output signal [15]. The presence of OCD block causes higher power dissipation even in ASVHB. QDI template sense amplifier half buffer (SAHB) addresses all these issues and provides optimized performance.

3 SAHB Design of 32-Bit Kogge Stone Adder

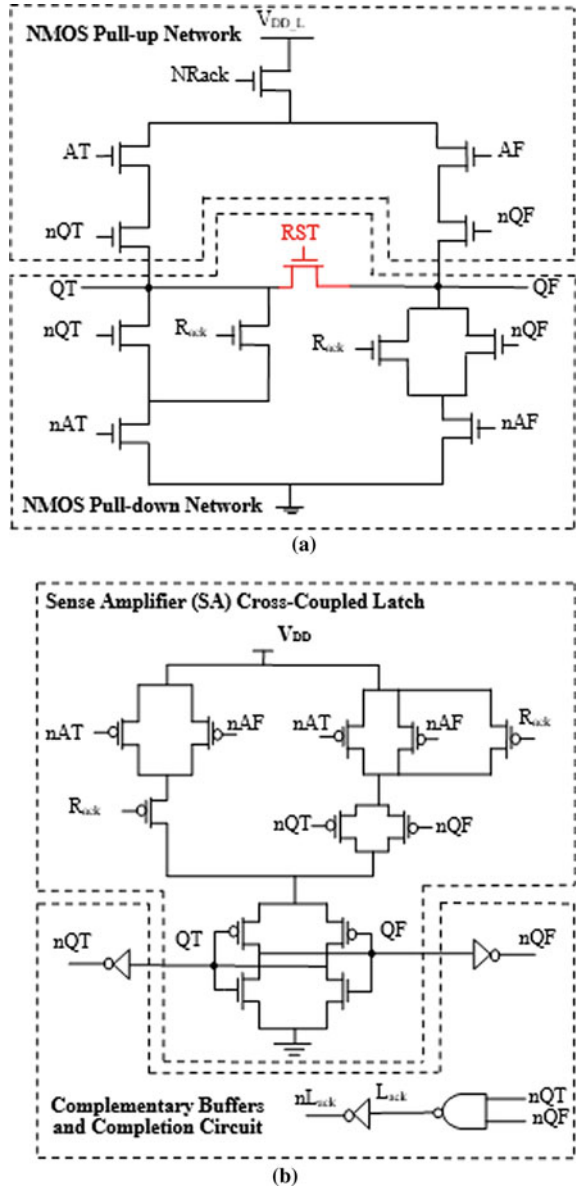
3.1 Sense Amplifier Half Buffer

SAHB cell is a leading QDI template, which operates with a 4-phase handshake protocol. The operation of this cell is based on two modes—evaluation and pre-charge mode. Figure 4 shows the gate-level design of the SAHB buffer cell, which consists of dual-rail data input/output signals (input-AT/AF, output-QT/QF, and complement outputs-nQT/nQF). The handshake signals L_ack/nL_ack are validated for output completeness, and R_ack/nR_ack signals are validated for input completeness. VDD_L feeds the evaluation block, and SA block is with VDD, respectively. The evaluation and SA block are interconnected, which reduces the count of switching nodes.

To minimize the power dissipation and internal leakages, the evaluation block of SAHB is designed with NMOS network. The buffer logic function of Fig. 4 is expressed in Eq. (1) [3]

$$QT = AT; QF = AF \tag{1}$$

Fig. 4 Architecture of SAHB buffer cell. **a** Evaluation block of SAHB cell, **b** sense amplifier block of SAHB cell



When all actual inputs (AT, R_{ack}) are asserted with logic ‘0’, the data AF is assigned with logic ‘1’ ($nAF = 0$) in the evaluation phase. Then, the NMOS pull-up network partially charges the output QF to $V_{DD,L}$ and QT set to logic ‘0’.

By applying the valid data to sense amplifier cross-coupled latch from the evaluation block, the final result QF is set to logic ‘1’. Therefore, QF is latched ($nQF = 0$) and generate L_{ack} as logic ‘1’ ($nL_{ack} = 0$). During the reset phase, the dual-rail

output attains logic '0' even when all inputs are valid/invalid and generates logic '0' at L_{ack} ; [3].

The critical path depends on the forward and backward pipeline latency as expressed in Eq. (2) [14].

Total cycle time for SAHB

$$CT_{SAHB} = FL_{SAHB} + BL_{SAHB} \quad (2)$$

Forward latency (FL) of SAHB

$$FL_{SAHB} = 3 \times t_{eval} + 3 \times t_{L_ON} \quad (3)$$

Backward latency (BL) of SAHB

$$BL_{SAHB} = 2 \times t_{CD} + t_{rst} + t_{L_RST} \quad (4)$$

where t_{eval} is the time evaluation, t_{L_ON} is ON time across cross-coupled latch, t_{CD} is time across completion detectors, t_{rst} is reset time, and t_{L_RST} is reset time of latch.

This template dissipates less power with the high-speed operation and reduces the leakage current by controlling the circuit with a single transistor (RST). In this paper, the performance of the 32-bit KSA adder using the SAHB cell is analyzed and compared with existing QDI templates (PCHB and ASVHB).

3.2 Kogge Stone Adder

Adder is a vital subsystem for performing arithmetic computations in DSP processors, ALU, and microprocessors. The present trendy parallel prefix adder is Kogge Stone adder (KSA), first introduced by Peter M Kogge and Harold S Stone in 1973 [10]. The KSA adder is a parallel prefix form of carry look ahead adder (CLA), suitable for high-speed applications is illustrated in Fig. 5. Parallel prefix computations are segregated into the pre-processing stage, carry generation network, and post-processing stage, as shown in Fig. 5 [10].

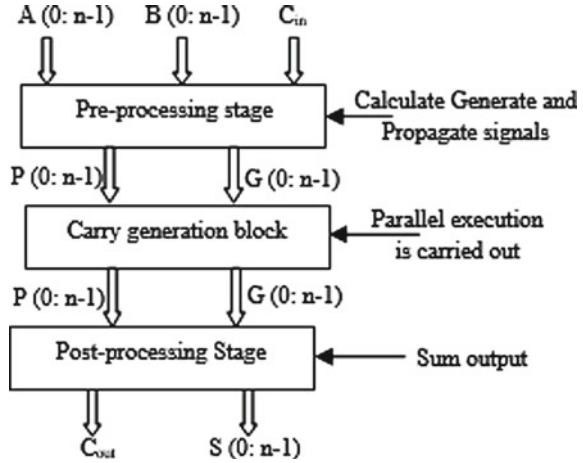
3.2.1 Pre-processing Stage

This stage computes, generates, and propagates signal for each input bit. The logic for generating these signals is expressed in Eqs. (5) and (6) [16].

$$P_i = A_i \oplus B_i \quad (5)$$

$$G_i = A_i.B_i \quad (6)$$

Fig. 5 Algorithm steps for Kogge Stone adder



3.2.2 Carry Generation Block

This stage calculates the carry corresponding to each bit, progressing execution in parallel format. Carry-generate and propagate signals are used as an intermediate signals. This stage considers two pairs of inputs, i.e., generate and propagate signals $[(G_i, P_i)$ and $(G_j, P_j)]$, and it computes a couple of generating, propagating signals (G_{ij}, P_{ij}) as output. These signals are expressed in Eqs. (7) and (8).

$$P_i : j = P_i : k.P_k - 1 : j \tag{7}$$

$$G_i : j = G_i : k + (P_i : k.G_k - 1 : j) \tag{8}$$

The carry generates and propagates signals are developed by black/gray cells. A gray cell generates a (G_i) signal, whereas a black cell propagates a (P_i) signal. The internal structures of these cells are explained in the following section.

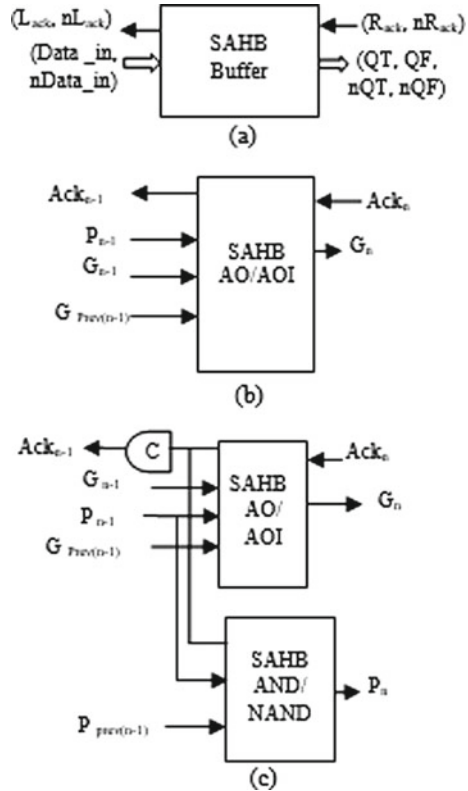
3.2.3 Post-processing Stage

This stage is to compute sum bits for the given input bits, and their logical equations are given in (9) and (10) [16–21].

$$C_i = (P_i.C_{in}) + G_i \tag{9}$$

$$S_i = P_i \oplus C_i - 1 \tag{10}$$

Fig. 6 Internal blocks of the KSA pipeline using SAHB, **a** SAHB buffer, **b** SAHB gray cell, and **c** SAHB black cell



Black and gray cells are designed with XOR gate, AO/AOI cell, AND/NAND cell, and buffer template using various low-power asynchronous QDI templates (PCHB, ASVHB, and proposed SAHB) [3, 16]. Figure 6 shows the block diagram of a gray cell, black cell, and buffer using the SAHB template [19].

Figure 7 shows the pipeline structure of 32-bit KSA, which is designed with buffers, XOR gate, AND gate, black cells, and gray cells [16, 20]. The input operands for 32-bit KSA design are represented with A (A_1 – A_{32}), B (B_1 – B_{32}), and output operands are sum (S_1 – S_{32}) and carryout (C_{32}), as shown in Fig. 6 [20]. Total, twelve pipeline stages are required to design 32-bit KSA, which results in forwarding latency of 12 pipeline delays, and the throughput rate is equal to one pipeline stage. A 32-bit KSA is designed and implemented with various QDI templates such as PCHB, ASVHB, and leading template SAHB along with traditional CMOS. 32-bit KSA using SAHB QDI cell is illustrated in Fig. 8.

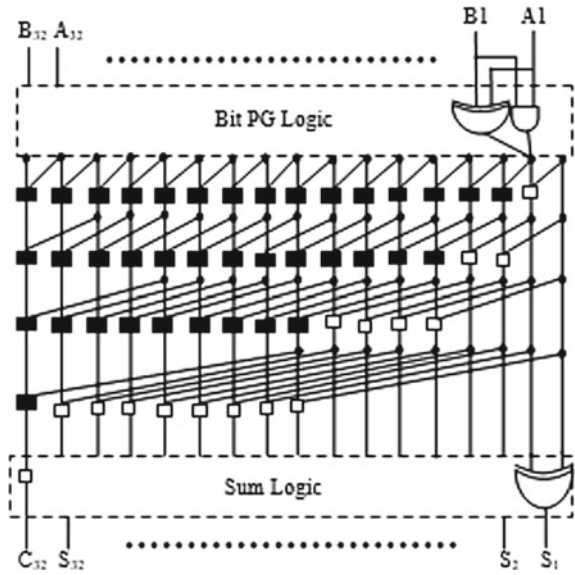


Fig. 7 Architecture of 32-bit KSA

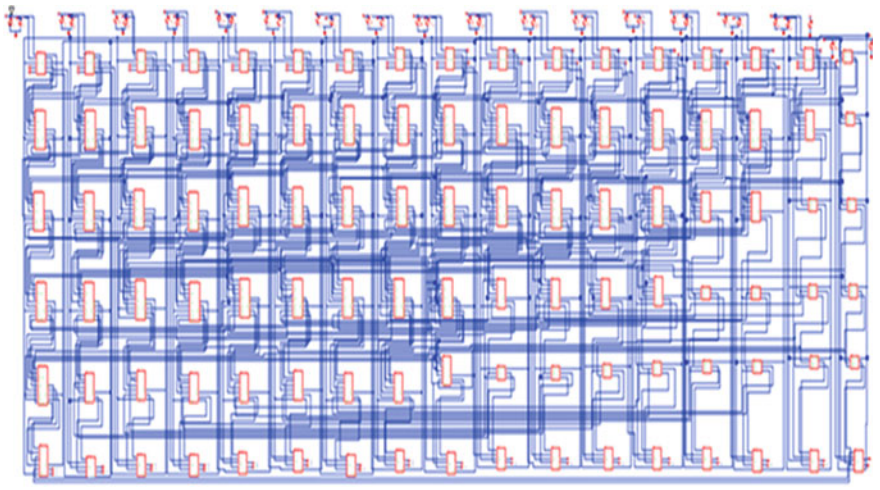


Fig. 8 Design of 32-bit KSA using SAHB QDI cell

4 Results and Discussion

This paper designs 32-bit KSA using asynchronous QDI templates such as PCHB, ASVHB, and leading SAHB cell templates. Table 1 shows the comparison table of 32-bit KSA design using various QDI half buffer cells in terms of power, delay, and

Table 1 Performance evaluation of various arithmetic templates

Circuit	Power dissipation (nW)						Delay (pS)						Energy (aJ)						
	CMOS	PCHB	ASVHB	SAHB	SAHB	CMOS	PCHB	ASVHB	SAHB	SAHB	CMOS	PCHB	ASVHB	SAHB	SAHB	CMOS	PCHB	ASVHB	SAHB
AO/AOI	84	73	67	56	56	46	40	39	34	34	3.86	2.92	2.61	1.9	1.9	3.86	2.92	2.61	1.9
AND/NAND	96	83	77	64	64	53	46	45	39	39	5.08	3.81	3.46	2.49	2.49	5.08	3.81	3.46	2.49
XOR	99	86	80	66	66	55	48	47	41	41	5.44	4.12	3.76	2.7	2.7	5.44	4.12	3.76	2.7
4-bit KSA	261	228	213	174	174	146	128	124	110	110	38.1	29.18	26.41	19.14	19.14	38.1	29.18	26.41	19.14
8-bit KSA	493	436	395	312	312	271	228	226	197	197	133.6	99.4	89.27	61.46	61.46	133.6	99.4	89.27	61.46
16-bit KSA	1095	948	826	684	684	582	511	497	407	407	637.29	484.42	410.52	278.38	278.38	637.29	484.42	410.52	278.38
32-bit KSA	1824	1582	1397	1135	1135	968	847	824	681	681	1765.63	1339.95	1151.12	772.93	772.93	1765.63	1339.95	1151.12	772.93

energy constraints. Power dissipation produces heat as an unwanted by-product. This unwanted heat leads to a reduction in the lifespan of an IC. Delay is another considerable parameter, which occurs between two timing operations. If delay increases while processing with multiple sequences, the circuit results may overlap and produce glitches. To improve the lifespan, reliability, other timing operations of an IC—power dissipation and delay should be reduced. The existing QDI cells (PCHB and ASVHB) dissipate more power because the pre-charge process has to be done at the initial condition of each stage/cycle. KSA design using leading SAHB QDI cell achieves low-power dissipation and provides less delay due to RST transistor in evaluation block and cross-coupled latch in sense amplifier block. Thus, KSA design using the SAHB template is better and can be utilized for various arithmetic computations.

5 Conclusion

This paper designs and implements a 32-bit KSA adder using traditional CMOS and various asynchronous QDI cells such as PCHB, ASVHB, and SAHB. A low power, less delay, and efficient energy are attained with the SAHB template due to the reset transistor in the evaluation block, which controls the leakages. From Table 1, it is observed that SAHB reduces power dissipation with 28%, 19% reduction in delay, and 42% of energy compared to existing QDI templates. This design is primarily helpful for low-power arithmetic computations in digital signal processors. Further reduce the power consumption of 32-bit KSA adder using traditional CMOS and various asynchronous QDI cells such as PCHB, ASVHB, and SAHB using low-power techniques and design mentioned above QDI cells using advanced technology files available in the market.

References

1. Kai W, Ahmad N, Jabbar MH (2017) Variable body biasing based VLSI design approach to reduce static power. *Int J Electr Comput Eng* 7(6):3010–3019
2. Stevens KS, Rotem S, Ginosar R, Beerel P, Myers CJ, Yun KY, Koi R, Dike C, Roncken M (2001) An asynchronous instruction length decoder. *IEEE J Solid-State* 36(2):217–228
3. Chong KS, Ho W-G, Lin T, Gwee BH, Chang JS (2017) Sense amplifier half-buffer (SAHB): a low-power high-performance asynchronous logic QDI cell template. *IEEE Trans VLSI Syst* 25(2):1–14
4. Balasubramanian P, Maskell D, Mastorakis N (2018) Low power robust early output asynchronous block carry-lookahead adder with redundant carry logic. *J Electron (MDPI)* 7(10):1–21
5. Sudhakar J, Sushma K (2017) Energy-efficient IEEE 754 floating-point multiplier dual spacer delay insensitive logic. *Circ World* 43(2):72–79
6. Martin AJ, Nystrom M (2006) Asynchronous techniques for the system-on-chip design. *Proc IEEE* 94(6):1089–1120
7. Nowick SM, Singh M (2015) Asynchronous design—part1: overview and recent advances. *IEEE Des Test* 32(3):5–18

8. Liu TT, Alarcón LP, Pierson MD, Rabaey JM (2009) Asynchronous computing in a sense amplifier-based pass transistor logic. *IEEE Trans Very Large Scale Integr (VLSI) Syst* 17(7):883–892
9. Manjunatha Naik V, Poornima N (2015) Performance analysis of parallel prefix adder. *Int J Electr Electron Data Commun* 3(7):74–77
10. Galphat V, Lonbale N (2015) Design the high-speed Kogge-Stone Adder by using MUX. *Int J Eng Res Appl* 5(8):58–60
11. Zhou R, Chong KS, Gwee BH, Chang JS (2014) A low overhead quasi-delay-insensitive (QDI) asynchronous data path synthesis based on a microcell-interleaving genetic algorithm (MIGA). *IEEE Trans Comput-Aided Des Integr Circ Syst* 33(7):989–1002
12. Sudhakar J, Prasad AM, Panda AK (2015) Multi-objective analysis of NCL threshold gates with a return to zero protocols. *IOSR J Electron Commun Eng* 10(3):12–17
13. Chuang C-C, Lai Y-H, Jiang J-H (2014) Synthesis of PCHB-WCHB hybrid quasi-delay insensitive circuits. In: *Proceedings of 51st ACM/EDAC/IEEE design automation conference (DAC)*, San Francisco, CA, USA, pp 1–6
14. Guazzelli RA (2017) QDI asynchronous design and voltage scaling. Book, Porto Alegre, UK
15. Ho WG, Chong KS, Gwee BH, Chang JS (2015) Low power sub-threshold asynchronous quasi-delay-insensitive 32-bit arithmetic and logic unit based on autonomous signal-validity half-buffer. *IET Circ Dev Syst* 9(4):309–318
16. Athira TS, Divya R, Karthik M, Manikandan A (2017) Design of Kogge-Stone for fast addition. *Int J Ind Electron Electr Eng* 5(4):48–50
17. Mei Xiang L, Mun'im Ahmad Zabidi M, Haziyah Awab A, Ab Rahman AA (2017) VLSI Implementation of a fast Kogge-Stone parallel-prefix adder. *J Phys* 1049(1):1–10
18. Madhavi K, Chandrasekar KN (2016) Design and comparative analysis of conventional adders and parallel prefix adders. *Int J Eng Trends Technol (IJETT)* 35(9):435–439
19. Sun S, Sechen C (2007) Post-layout comparison of high performance 64b static adders in energy delay space. In: *Proceedings of the IEEE 25th international conference on computer design: VLSI in computers and processors (ICCD)*, Lake Tahoe, CA, USA, pp 401–408
20. Zeydel BR, Baran D, Oklobdzija VG (2010) Energy-efficient design methodologies: high-performance VLSI adders. *IEEE J Solid-State Circ* 45(6):1220–1233
21. Chaitanya P, Nagendra R (2013) Design of 32-bit parallel prefix adders. *IOSR J Electron Commun Eng* 6(1):1–6

Intellectualization of Lean Production Logistic Technology Based on Fuzzy Expert System and Multi-agent Metaheuristics



Eugene Fedorov , Svitlana Smerichevska , Olga Nechyporenko , Tetyana Utkina , and Yuliia Remyha 

Abstract The paper discusses the intellectualization of lean production in order to minimize the costs main groups that do not develop customer value for the end user in supply chain management industrial enterprises by creating optimization methods based on multi-agent metaheuristics and an adaptive fuzzy expert system for evaluating equipment load efficiency, the use of which is aimed at creating “perfect,” and, consequently, competitive supply chains for economy globalization and intellectualization. The possibility of minimizing costs associated with unnecessary (unjustified) movements of goods and personnel has been substantiated, and it is also proposed to minimize losses associated with expectations, with managing stocks and determining the shortest path of movement of goods through multi-agent metaheuristic methods based on particle swarm optimization and simulated annealing. The proposed methods provide control over the convergence rate of the method, providing global (at initial iterations) and local (at final iterations) searches by simulating annealing, the possibility of discrete and conditional optimization through the random key technique and the penalty function. An adaptive fuzzy expert system for assessing the efficiency of equipment load has been developed which simplifies the interaction between the operator and the computer system through the use of quality indicators and also allows the identification of its parameters using the proposed multi-agent metaheuristics. The proposed optimization methods based on

E. Fedorov (✉) · O. Nechyporenko · T. Utkina
Cherkasy State Technological University, Shevchenko blvd., 460, Cherkasy, Ukraine
e-mail: fedorovee75@ukr.net

O. Nechyporenko
e-mail: olne@ukr.net

T. Utkina
e-mail: t.utkina@chdtu.edu.ua

S. Smerichevska
National Aviation University, L. Gyzara avenue, 1, Kyiv, Ukraine
e-mail: smerichevska.s@gmail.com

Y. Remyha
International European University, Acad. Glushkov avenue, 42V, Kyiv, Ukraine
e-mail: remyga_julia@ukr.net

multi-agent metaheuristics and an adaptive fuzzy expert system allow intellectualizing Lean Production logistic technology for industrial enterprises.

Keywords Lean production · Logistic technology · Optimization methods · Multi-agent metaheuristics · Adaptive fuzzy expert system · Particle swarm optimization · Simulated annealing · Efficient equipment load · Minimization of production losses

1 Introduction

Currently, most companies are striving to improve their production based on the introduction of lean production technology, which is effective for companies in various industries at all stages of the supply chain of products to the consumer. Lean production technology is especially important for industrial enterprises. In the context of digitalization and globalization and the economy based on communication and information technologies, lean production technology is dominant in creating “perfect” supply chains, which ensures competitiveness [20]. As a result, the development of intellectualization methods for lean production technology, which is based on the solution of optimization problems, is relevant.

Methods that find an exact solution to optimization problems are slow. Random search methods do not guarantee convergence. Directed search methods have a high probability of hitting a local minimum or maximum. Thus, the problem of low efficiency of optimization methods is relevant. Metaheuristics are used to improve the accuracy and speed of solving optimization problems [9, 16, 22, 23, 25]. Metaheuristics combining heuristic methods based on a high-level strategy [3, 15, 19, 21, 24].

The disadvantages of modern metaheuristics:

- the method is focused on solving only a specific problem or has an abstract description [22];
- the method convergence is not guaranteed [11];
- the iteration number is not taken into account for finding the solution [9];
- the non-boolean potential solutions are not use [7];
- insufficient accuracy of the method [18];
- the determining parameter is not automated [4];
- conditional optimization problems are not solve [6].

Thereby, the problem of constructing effective metaheuristic optimization methods comes up [10, 12].

One of the popular metaheuristics is the method of particle swarm optimization [1, 5, 8].

Another urgent task is the development of expert systems that are used for lean production logistic technology and are aimed at assessing production losses that do

not develop additional consumer value for the product. Such losses include overproduction, excess stocks, transportation, moving, waiting, excessive processing, defects and rework.

The creation of such systems is based on a knowledge base (most often in the form of production rules) and a logical inference mechanism.

The disadvantages of such systems include the fact that they require accurate quantitative estimates, while it is more convenient for the operator to work with qualitative estimates.

To simplify the interaction between the operator and the computer system, fuzzy expert systems are currently used, which usually use the fuzzy logic Mamdani, Larsen, Sugeno and Tsukamoto inference mechanism [2, 17].

The disadvantages of such systems include the fact that their parametric identification is not automated [13, 14].

Thereby, the task of constructing fuzzy expert systems that use the parametric identification method for adaptation and tuning arises.

The purpose of the work is the intellectualization of lean production logistic technology through the creation of optimization methods based on multi-agent metaheuristics and an adaptive fuzzy expert system for evaluating equipment load efficiency.

To reach the target, the following tasks were intended and solved:

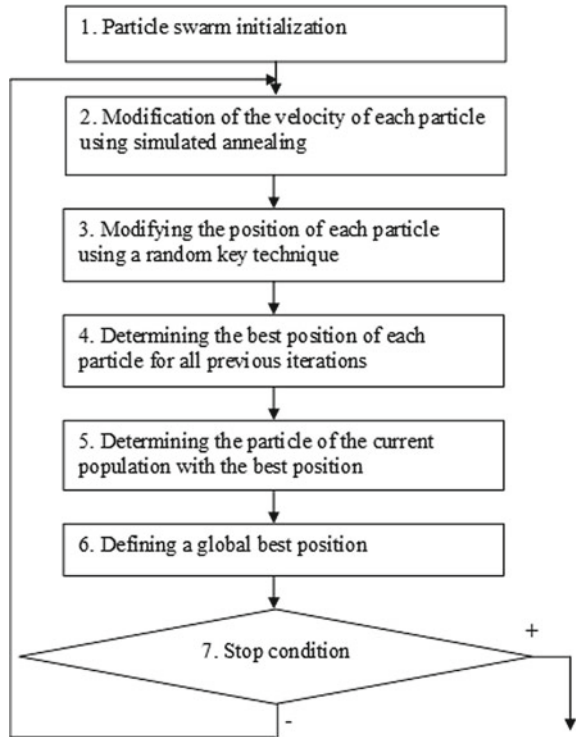
1. To develop multi-agent metaheuristic methods based on particle swarm optimization and simulated annealing to solve the traveling salesman problem, the shortest path problem, the assignment problem, the inventory control problem.
2. To develop an adaptive fuzzy expert system for assessing the equipment load efficiency.
3. Conduct a numerical study.

2 Materials and Methods

2.1 Minimizing the Costs Associated with Unnecessary (Unjustified) Movements, Which Used the Multi-agent Metaheuristic for Solving the Traveling Salesman Problem

Optimization of costs associated with the movement of, for example, personnel in the process of collecting goods in a warehouse for loading, may be reduced to the traveling salesman problem (TSP). To solve TSP, a multi-agent metaheuristic method SAPSORK is proposed—particle swarm optimization (PSO), which uses simulated annealing (SA) and a random key (RK).

Fig. 1 Structure of the multi-agent metaheuristic SAPSORK method for optimizing the collection of goods in the warehouse



The random key technique is that each component of an unordered real vector is associated with its position in the ascending real vector. The multi-agent metaheuristic SAPSORK structure is shown in Fig. 1.

It is proposed to use the length of the route as a target function F

$$F(x) = d_{x_M, x_1} + \sum_{i=1}^{M-1} d_{x_i, x_{i+1}} \rightarrow \min_x,$$

where d_{x_k, x_1} —distance between points x_i and x_{i+1} , $x_i, x_{i+1} \in V$, $V = \{1, \dots, M\}$ —set of items numbers, x —items numbers vector (route), M —items numbers vector length (route length).

The SAPSORK method allows one to find a quasi-optimal route between all points of transportation of goods and consists of the following blocks.

Block 1—Initialization:

- set the current iteration number $n = 1$;
- set the iterations maximum number N ;

- set the swarm size K ;
- set the dimension of the particle position M (length of the route);
- setting the distance between two points $d_{i,j}$, $i, j \in \overline{1, M}$;
- initialization of position x_k (corresponds to the route) using the random key technique

$$\tilde{x}_k = (\tilde{x}_{k1}, \dots, \tilde{x}_{kM}), \tilde{x}_{kj} = U(0, 1),$$

$$x_k = \text{random_key}(\tilde{x}_k),$$

where $\text{random_key}(\cdot)$ —function mapping a real vector in a route based on a random key technique, $U(0, 1)$ —a uniformly distributed random number for interval $[0, 1]$;

- perform personal best position initialization x_k^{best}

$$\tilde{x}_k^{best} = \tilde{x}_k,$$

$$x_k^{best} = x_k;$$

- perform speed initialization v_k

$$v_k = (v_{k1}, \dots, v_{kM}), v_{kj} = 0;$$

- perform an initial particles swarm

$$Q = \{(x_k, \tilde{x}_k, x_k^{best}, \tilde{x}_k^{best}, v_k)\};$$

- calculation of a particle swarm from the population with the best position (which according to the target function)

$$k^* = \arg \min_{k \in \overline{1, K}} F(x_k),$$

$$x^* = x_{k^*},$$

$$\tilde{x}^* = \tilde{x}_{k^*}.$$

Block 2—Calculation of the particle velocity on based simulated annealing

$$r1_k = (r1_{k1}, \dots, r1_{kM}), r1_{kj} \in \{U(0, 1), C(0, 1), N(0, 1)\}, k \in \overline{1, K}, j \in \overline{1, M},$$

$$r2_k = (r2_{k1}, \dots, r2_{kM}), r2_{kj} \in \{U(0, 1), C(0, 1), N(0, 1)\}, k \in \overline{1, K}, j \in \overline{1, M},$$

$$v_k = w(n)v_k + \alpha_1(n)(x_k^{best} - x_k)(r1_k)^T + \alpha_2(n)(x^* - x_k)(r2_k)^T, k \in \overline{1, K},$$

$$\alpha_1(n) = \alpha_2(n) = \alpha(0) \exp(-1/T(n)), w(n) = w(0) \exp(-1/T(n)),$$

$$\alpha(0) = \alpha_0 = 0.5 + \ln 2, w(0) = w_0 = \frac{1}{2 \ln 2},$$

$$T(n) = \beta T(n - 1), T(0) = T_0,$$

$$\beta = N^{-\frac{1}{N-1}}, T_0 = N^{\frac{N}{N-1}},$$

where $N(0, 1)$ —a standard normally distributed random number, $C(0, 1)$ —a standard Cauchy distributed random number, $\alpha_1(n)$ —parameter for component $(x_k^{best} - x_k)(r1_k)^T$, $\alpha_2(n)$ —parameter for component $(x^* - x_k)(r2_k)^T$, $w(n)$ —parameter for particle velocity, α_0 —parameters initial value $\alpha_1(n)$ and $\alpha_2(n)$, w_0 —parameter initial value $w(n)$, $T(n)$ —annealing temperature at iteration n , T_0 —temperature for initial annealing, β —parameter controlling the annealing temperature.

The simulated annealing introduced in this work allows the establishing of inverse relationship between the parameters $\alpha_1(n)$, $\alpha_2(n)$, $w(n)$ and the iteration number. In this paper, a relationship is determined between the iteration number and the parameters T_0 , β , which allows to automate the set of parameters T_0 , β .

The choice of initial values $\alpha_0 = 0.5 + \ln 2$ and $w_0 = \frac{1}{2 \ln 2}$ is standard and satisfies the conditions of particle swarm convergence $w < 1$ and $w_0 > \frac{1}{2}(\alpha_1 + \alpha_2) - 1$.

Block 3—Modifying the position of each particle using a random key technique

$$\tilde{x}_k = \tilde{x}_k + v_k, k \in \overline{1, K},$$

$$x_k = random_key(\tilde{x}_k), k \in \overline{1, K}.$$

Block 4—Determining the each particle personal best position

If $F(x_k) \leq F(x_k^{best})$, then $x_k^{best} = x_k, \tilde{x}_k^{best} = \tilde{x}_k, k \in \overline{1, K}$.

Block 5—Determining the particle of the current population with the best position

$$k^* = \arg \min_{k \in \overline{1, K}} F(x_k).$$

Block 6—Calculating the global best position

If $F(x_{k^*}) < F(x^*)$, then $x^* = x_{k^*}, \tilde{x}^* = \tilde{x}_{k^*}$.

Block 7—Termination condition.

If $n < N$, then increment the iteration number n and go to Block 2.

The SAPSORK method allows one to find a quasi-optimal route between the indicated points.

2.2 *Minimization of Costs Associated with Unnecessary Movements and Irrational Transportations, Based on the Multi-agent Metaheuristic Method for Solving the Shortest Path Problem*

Elimination, in accordance with lean production logistic technology, of such types of costs that do not create consumer value, such as unreasonable movement of personnel, or irrational ways of transporting materials at the factory in accordance with the technological cycle, can in fact be reduced to solving the shortest path problem.

To solve this problem, a multi-agent metaheuristic method SAPSORK can be proposed—particle swarm optimization—which uses simulated annealing and a random key.

It is proposed to use as a target function F the length of the shortest path from point 1 to point M containing L points out of M possible, $2 \leq L \leq M$

$$F(x, L) = \sum_{i=1}^{L-1} d_{x_i, x_{i+1}} \rightarrow \min_x,$$

$$x_1 = 1, x_L = M,$$

where d_{x_k, x_l} —distance between points x_i and x_{i+1} , $x_i, x_{i+1} \in V$, $V = \{1, \dots, M\}$ —set of items numbers, x —items numbers vector, M —items numbers vector length.

The SAPSORK method allows one to find a quasi-optimal route between the source and destination.

2.3 *Minimization of Losses Associated with Waiting Based on a Multi-agent Metaheuristic Method for Solving the Assignment Problem*

Optimization of loading the collected goods that is carried out within the framework of lean production logistic technology consists in the distribution of goods between movers and comes down to the assignment problem. To solve this problem, a multi-agent metaheuristic method SAPSO is proposed—particle swarm optimization—which uses simulated annealing.

It is proposed to use the reverse costs as a target function F

$$F(x) = \left(\sum_{i=1}^M w_{i, x_i} \right)^{-1} \rightarrow \min_x,$$

where w_{i,x_i} – the cost of the i th agent performing the x_i th task, is set, x —vector of numbers of agents, M —number of agents/tasks.

The SAPSO method allows one to perform quasi-optimal assignment of agents to task and is similar to the SAPSORK method, but does not use a random key.

2.4 Multi-agent Metaheuristic Method for Solving the Inventory Control Problem

Optimization of the volume of stored stocks in the warehouse is carried out in the framework of lean production logistic technology and is reduced to the inventory control problem. To solve this problem, a multi-agent metaheuristic method SAPSO is also proposed—particle swarm optimization—which uses simulated annealing. It is proposed to use as a target function F the partial derivative of cost function $F1$ and the penalty function $F2$ based on simulated annealing

$$F(x, n) = F1(x)/F2(z, n) \rightarrow \min_x,$$

$$F1(x) = \sum_{j=1}^M (w1_j x_j + w2_j z_j),$$

$$F2(z, n) = \exp\left(-\frac{\varphi(z)}{T(n)}\right),$$

$$\varphi(z) = \sum_{j=1}^M \max\{0, z_j^{\min} - z_j\} + \sum_{j=1}^M \max\{0, z_j - z_j^{\max}\},$$

$$z_j = x_j + z_{j-1} - D_j,$$

$$T(n) = \beta T(n - 1), T(0) = T_0,$$

$$\beta = N^{-\frac{1}{N-1}}, T_0 = N^{\frac{N}{N-1}},$$

where n —iteration number, T_0 —initial annealing temperature, β —parameter controlling the annealing temperature, $w1_j$ —purchase costs from the supplier of one unit of goods, is set, $w2_j$ —storage costs of one unit of goods, is set, x_j —the amount of purchased goods from the supplier during the j th stage, z_j —the number of stocks of goods at the end of the j th stage, z_0 —initial quantity of stocks of goods, is set, z_j^{\min}, z_j^{\max} —the minimum and maximum quantity of stocks of goods at the end of the j th stage, is set, D_j —the amount of goods sold during the j th stage, is set, M —number of stages, $\varphi(\cdot)$ —restrictions violation measure.

The SAPSO method allows one to perform quasi-optimal assignment of executors to work and is similar to the SAPSORK method, but does not use a random key.

2.5 Adaptive Fuzzy Expert System for Evaluating the Efficiency of Equipment Load

Optimization, in the framework of lean production logistic technology, of the waiting time, for example, of equipment, can be reduced to an analysis of equipment load based on information on the number of cases and duration of equipment inactivity, as well as the distance between equipment and the duration of its readjustment. To evaluate the equipment load efficiency, an adaptive fuzzy expert system is proposed, which involves the following steps:

1. The linguistic variables creation.
2. The fuzzy knowledge base creation.
3. The formation of the fuzzy logic Mamdani inference mechanism:
 - fuzzification;
 - sub-condition aggregation;
 - conclusions activation;
 - conclusions aggregation;
 - defuzzification.
4. Identification of parameters based on multi-agent metaheuristics.

2.5.1 The Formation of Linguistic Variables

As clear input variables were selected:

- number of equipment inactivity cases x_1 ;
- equipment inactivity time x_2 in units of time;
- distance between equipment x_3 ;
- equipment readjustment time x_4 .

As linguistic input variables were selected:

- the number of equipment inactivity cases \tilde{x}_1 with its values $\tilde{\alpha}_{11} = \textit{little}$, $\tilde{\alpha}_{12} = \textit{average}$, $\tilde{\alpha}_{13} = \textit{a lot}$, whose values intervals are fuzzy sets $\tilde{A}_{11} = \{x_1 | \mu_{\tilde{A}_{11}}(x_1)\}$, $\tilde{A}_{12} = \{x_1 | \mu_{\tilde{A}_{12}}(x_1)\}$, $\tilde{A}_{13} = \{x_1 | \mu_{\tilde{A}_{13}}(x_1)\}$;
- the duration of equipment inactivity \tilde{x}_2 with its values $\tilde{\alpha}_{21} = \textit{short}$, $\tilde{\alpha}_{22} = \textit{average}$, $\tilde{\alpha}_{23} = \textit{long}$, whose values intervals are fuzzy sets $\tilde{A}_{21} = \{x_2 | \mu_{\tilde{A}_{21}}(x_2)\}$, $\tilde{A}_{22} = \{x_2 | \mu_{\tilde{A}_{22}}(x_2)\}$, $\tilde{A}_{23} = \{x_2 | \mu_{\tilde{A}_{23}}(x_2)\}$;

- the distance between equipment \tilde{x}_3 with its values $\tilde{\alpha}_{31} = \textit{close}$, $\tilde{\alpha}_{32} = \textit{average}$, $\tilde{\alpha}_{33} = \textit{far away}$, whose values intervals are fuzzy sets $\tilde{A}_{31} = \{x_3 | \mu_{\tilde{A}_{31}}(x_3)\}$, $\tilde{A}_{32} = \{x_3 | \mu_{\tilde{A}_{32}}(x_3)\}$, $\tilde{A}_{33} = \{x_3 | \mu_{\tilde{A}_{33}}(x_3)\}$;
- the duration of equipment readjustment \tilde{x}_4 with its own values $\tilde{\alpha}_{41} = \textit{short}$, $\tilde{\alpha}_{42} = \textit{average}$, $\tilde{\alpha}_{43} = \textit{long}$, whose values intervals are fuzzy sets $\tilde{A}_{41} = \{x_4 | \mu_{\tilde{A}_{41}}(x_4)\}$, $\tilde{A}_{42} = \{x_4 | \mu_{\tilde{A}_{42}}(x_4)\}$, $\tilde{A}_{43} = \{x_4 | \mu_{\tilde{A}_{43}}(x_4)\}$.

As clear output variables were selected:

- action number for changing equipment distance \tilde{y}_1 ;
- action number for changing the equipment readjustment time \tilde{y}_2 .

As linguistic output variables were selected:

- action for changing the distance between equipment \tilde{y}_1 with its values $\tilde{\beta}_{11} = \textit{do not bring closer}$, $\tilde{\beta}_{12} = \textit{bring closer}$, whose values intervals are fuzzy sets $\tilde{B}_{11} = \{y_1 | \mu_{\tilde{B}_{11}}(y_1)\}$, $\tilde{B}_{12} = \{y_1 | \mu_{\tilde{B}_{12}}(y_1)\}$;
- action for changing the duration of equipment readjustment \tilde{y}_2 with its values $\tilde{\beta}_{21} = \textit{don't cut back}$, $\tilde{\beta}_{22} = \textit{to cut back}$, whose values intervals are fuzzy sets $\tilde{B}_{21} = \{y_2 | \mu_{\tilde{B}_{21}}(y_2)\}$, $\tilde{B}_{22} = \{y_2 | \mu_{\tilde{B}_{22}}(y_2)\}$.

2.5.2 Formation of a Fuzzy Knowledge Base

Fuzzy knowledge is presented in the form of the following fuzzy rules that contain:

- first linguistic output variable R^n :IF \tilde{x}_1 is $\tilde{\alpha}_{1i}$ AND \tilde{x}_2 is $\tilde{\alpha}_{2j}$ AND \tilde{x}_3 is $\tilde{\alpha}_{3k}$ THEN \tilde{y}_1 is $\tilde{\beta}_{1m}$,

$$n = i + l_1(j - 1) + l_1l_2(k - 1), m = \begin{cases} 1, & k = 1 \\ 2, & k > 1 \end{cases}, i \in \overline{1, l_1}, j \in \overline{1, l_2}, k \in \overline{1, l_3};$$

- second linguistic output variable R^n : IF \tilde{x}_1 is $\tilde{\alpha}_{1i}$ AND \tilde{x}_2 is $\tilde{\alpha}_{2j}$ AND \tilde{x}_4 is $\tilde{\alpha}_{4p}$ THEN \tilde{y}_2 is $\tilde{\beta}_{2m}$,

$$n = l_1l_2l_3 + i + l_1(j - 1) + l_1l_2(p - 1),$$

$$m = \begin{cases} 1, & p = 1 \\ 2, & p > 1 \end{cases}, i \in \overline{1, l_1}, j \in \overline{1, l_2}, p \in \overline{1, l_4},$$

where l_s —the number of values of the s th linguistic input variable.

2.5.3 The Formation of the Fuzzy Logic Mamdani Inference Mechanism

Fuzzification. Let us calculate the power of truth of each sub-condition of each

production, using the membership function $\mu_{\tilde{A}_{ij}}(x_i)$. As membership functions of the sub-conditions were selected:

- logistic function $a_{i1} < 0$, i.e.,

$$\mu_{\tilde{A}_{i1}}(x_i) = \exp[-a_{i1}(x_i - b_{i1})], \quad i \in \overline{1, 4};$$

- Gauss function, i.e.,

$$\mu_{\tilde{A}_{i2}}(x_i) = \exp\left[-\frac{1}{2}\left(\frac{x_i - b_{i2}}{a_{i2}}\right)^2\right], \quad i \in \overline{1, 4};$$

- logistic function with $a_{i3} > 0$, i.e.,

$$\mu_{\tilde{A}_{i3}}(x_i) = \exp[-a_{i3}(x_i - b_{i3})], \quad i \in \overline{1, 4},$$

where a_{ij}, b_{ij} —membership function parameters.

Aggregation of Sub-conditions. Based on the minimum value method, the membership functions of the condition are determined for each rule that contains:

- first linguistic output variable

$$\mu_{\tilde{A}_{i1} \cup \tilde{A}_{2j} \cup \tilde{A}_{3k}}(x_1, x_2, x_3) = \min\{\mu_{\tilde{A}_{i1}}(x_1), \mu_{\tilde{A}_{2j}}(x_2), \mu_{\tilde{A}_{3k}}(x_3)\},$$

$$i \in \overline{1, l_1}, j \in \overline{1, l_2}, k \in \overline{1, l_3};$$

- second linguistic output variable

$$\mu_{\tilde{A}_{i1} \cup \tilde{A}_{2j} \cup \tilde{A}_{4p}}(x_1, x_2, x_4) = \min\{\mu_{\tilde{A}_{i1}}(x_1), \mu_{\tilde{A}_{2j}}(x_2), \mu_{\tilde{A}_{4p}}(x_4)\},$$

$$i \in \overline{1, l_1}, j \in \overline{1, l_2}, p \in \overline{1, l_4}.$$

Activation of Conclusions. Based on the minimum value method, membership functions are determined for each rule that contains:

- first linguistic output variable

$$\mu_{\tilde{C}^n}(x_1, x_2, x_3, y_1) = \min\{\mu_{\tilde{A}_{i1} \cup \tilde{A}_{2j} \cup \tilde{A}_{3k}}(x_1, x_2, x_3), \mu_{\tilde{B}_{1m}}(y_1)\},$$

$$n = i + l_1(j - 1) + l_1 l_2(k - 1), m = \begin{cases} 1, & k = 1 \\ 2, & k > 1 \end{cases}, i \in \overline{1, l_1}, j \in \overline{1, l_2}, k \in \overline{1, l_3};$$

- second linguistic output variable

$$\mu_{\tilde{C}^n}(x_1, x_2, x_4, y_2) = \min\{\mu_{\tilde{A}_{i1} \cup \tilde{A}_{2j} \cup \tilde{A}_{4p}}(x_1, x_2, x_4), \mu_{\tilde{B}_{2m}}(y_2)\},$$

$$n = l_1 l_2 l_3 + i + l_1(j - 1) + l_1 l_2(p - 1),$$

$$m = \begin{cases} 1, & p = 1 \\ 2, & p > 1 \end{cases}, i \in \overline{1, l_1}, j \in \overline{1, l_2}, p \in \overline{1, l_4}.$$

In this paper, membership functions $\mu_{B_{im}}(y_i)$ are defined as indicator functions in the form of

$$\mu_{B_{im}}(y_i) = \chi_{\{m\}}(y_i) = \begin{cases} 1, & y_i \in \{m\} \\ 0, & y_i \notin \{m\} \end{cases}.$$

Aggregation of Conclusions. Based on the maximum value method, the membership functions of the final conclusion are determined, which contains:

- first linguistic output variable

$$\mu_{\tilde{C}}(x_1, x_2, x_3, y_1) = \max_n \{\mu_{\tilde{C}^n}(x_1, x_2, x_3, y_1)\}, n \in \overline{1, l_1 l_2 l_3};$$

- second linguistic output variable

$$\mu_{\tilde{C}}(x_1, x_2, x_4, y_2) = \max_n \{\mu_{\tilde{C}^n}(x_1, x_2, x_4, y_2)\}, n \in \overline{1, l_1 l_2 l_3 + 1, l_1 l_2 l_3 + l_1 l_2 l_4}.$$

Defuzzification. Based on the maximum method, the numbers of action types are determined for changing:

- distance between equipment

$$y_1 = \arg \max_m \mu_{\tilde{C}}(x_1, x_2, x_3, m), \quad m \in \overline{1, 2};$$

- equipment changeovers

$$y_2 = \arg \max_m \mu_{\tilde{C}}(x_1, x_2, x_4, m), \quad m \in \overline{1, 2}.$$

2.5.4 Identification of Parameters Based on Multi-agent Metaheuristics

To identify the parameters of membership functions, a multi-agent metaheuristic method SAPSO is proposed—particle swarm optimization,—which uses simulated annealing.

As a target function F , it is proposed to use the probability of the correct action

$$F = \frac{1}{P} \sum_{p=1}^P I(y_p - d_p) \rightarrow \max_x,$$

$$I(a) = \begin{cases} 1, & a = 0 \\ 0, & \text{else} \end{cases},$$

where d_p —response received from the controlled object, y_p —response received as a result from fuzzy inference, P —number of test implementations, $x = (a_{11}, \dots, a_{43}, b_{11}, \dots, b_{43})$ —membership function parameter vector.

The SAPSO method allows for quasi-optimal identification of the parameters of membership functions and is similar to the SAPSORK method, but does not use a random key.

3 Results and Discussion

For the traveling salesman problem, a search for the solution was conducted on the standard berlin52 database.

For the shortest path problem, a search for the solution was conducted on the standard rcspl database.

For the assignment problem, a search for the solution was conducted on the tai50a standard database.

For the inventory control problem, the search for the solution was conducted on the information of the logistics firm “Ekol Ukraine.” The results of comparison of the proposed method with the traditional method of particle swarm optimization are presented in Table 1.

To assess the efficiency of equipment loading, an adaptive fuzzy expert system was also studied on the data of “Ekol Ukraine” logistics company.

The comparison results of the proposed adaptive fuzzy expert system with the operator are presented in Table 2.

The identification of parameters $\alpha_1(n), \alpha_2(n), w(n)$ based on the simulated annealing of the proposed optimization method, provides strong changes in the particle velocity and, accordingly, the potential solution at the initial iterations and weak

Table 1 Comparison of the proposed and traditional methods of particle swarm optimization

No. p/p	Problem	RMS error method	
		Proposed	Existing
1	Traveling salesman	0.02	0.08
2	Shortest path	0.02	0.07
3	Assignment	0.03	0.09
4	Inventory control	0.04	0.1

Table 2 Comparison of an adaptive fuzzy expert system with an operator

The probability of a correct assessment in the case of	
Adaptive fuzzy expert system	The operator
0.98	0.9

changes in the particle velocity and, accordingly, the potential solution at the final iterations.

Traditional particle swarm optimization method:

- ignores the iteration number, which reduces the accuracy of finding a solution (Table 1);
- does not allow for integer potential solutions, which makes discrete optimization impossible;
- does not give an opportunity us to find the conditional minimum or maximum.

The suggested method allows you to fix these drawbacks.

The traditional non-automated approach to assessing the efficiency of equipment loading reduces the likelihood of correct assessment (Table 2).

The proposed method allows us to eliminate this drawback.

4 Conclusions

1. In the framework of lean production technology, to solve the problem of optimizing the business processes of the industrial enterprises based on information and communication technologies, as well as reducing losses that do not create additional consumer value for the product, the corresponding optimization methods and expert systems were studied. These researches have shown that present day the most effective is the apply of metaheuristic methods and fuzzy expert systems.
2. To minimize losses that do not create consumer value and which underlie lean production technology, multi-agent metaheuristic methods have been developed based on particle swarm optimization and simulated annealing to solve the traveling salesman problem, the shortest path problem, the assignment problem and the inventory control problem. The use of these methods is aimed at minimizing the costs associated with unnecessary movements, irrational transportations, waiting and stocks.

The proposed metaheuristic methods provide: control of the convergence rate of the method, as well as providing at the initial iterations of the global search, and at the final iterations of the local search through the simulated annealing; the possibility of discrete and conditional optimization through the random key technique and the penalty function.

3. An adaptive fuzzy expert system for evaluating the efficiency of equipment loading has been developed. The proposed system simplifies the interaction between the operator and the computer system through the use of quality indicators and also allows for identifying its parameters using the proposed multi-agent metaheuristics.

The proposed optimization methods based on multi-agent metaheuristics and an adaptive fuzzy expert system allow intellectualizing lean production logistic technology for industrial enterprises.

Prospects for further research are to test the proposed methods on a wider set of test databases.



References

1. Alba E, Nakib A, Siarry P (2013) *Metaheuristics for dynamic optimization*. Springer, Berlin Heidelberg, p 400
2. Aliev RA, Fazlollahi B, Aliev RR, Guirimov B (2008) Linguistic time series forecasting using fuzzy recurrent neural network. *Int Soft Comput J* 12:183–190
3. Blum C, Raidl GR (2016) *Hybrid metaheuristics. Powerful tools for optimization*. Springer International Publishing, p 157
4. Bozorg-Haddad O, Solgi M, Loáiciga HA (2017) *Meta-heuristic and evolutionary algorithms for engineering optimization*. Wiley & Sons, Hoboken, New Jersey, p 304
5. Brownlee J (2011) *Clever algorithms: nature-inspired programming recipes*. Brownlee, Melbourne, p 436
6. Chopard B, Tomassini M (2018) *An introduction to metaheuristics for optimization*. Springer International Publishing, p 226
7. Doerner KF, Gendreau M, Greistorfer P, Gutjahr W, Hartl RF, Reimann M (2007) *Metaheuristics. Progress in complex systems optimization*. Springer US, p 410
8. Du K-L, Swamy MNS (2016) *Search and optimization by metaheuristics. Techniques and algorithms inspired by nature*. Birkhäuser Basel, p 434
9. Engelbrecht AP (2007) *Computational intelligence: an introduction*. Wiley & Sons, Chichester, West Sussex, p 630
10. Fedorov E, Lukashenko V, Utkina T, Lukashenko A, Rudakov K (2019) Method for parametric identification of gaussian mixture model based on clonal selection algorithm. In: *CEUR workshop proceedings*, vol 2353, pp 41–55
11. Gendreau M, Potvin J-Y (2010) *Handbook of metaheuristics*. Springer International Publishing, p 604
12. Grygor O, Fedorov E, Utkina T, Lukashenko A, Rudakov K, Harder D, Lukashenko V (2019) Optimization method based on the synthesis of clonal selection and annealing simulation algorithms. *Radio Electron Comput Sci Control* 2:90–99
13. Hsu C-F (2007) Self-organizing adaptive fuzzy neural control for a class of nonlinear systems. *IEEE Trans Neural Netw* 18(4):1232–1241
14. Lu C-H, Tsai C-C (2007) Generalized predictive control using recurrent fuzzy neural networks for industrial processes. *J Process Control* 17(1):83–92
15. Martí R, Pardalos PM, Resende MG (2018) *Handbook of heuristics*. Springer International Publishing, p 1289
16. Nakib A, Talbi E-G (2017) *Metaheuristics for medicine and biology*. Springer, Berlin Heidelberg, p 211
17. Pan HY, Lee CH, Chang FK, Chang SK (2007) Construction of asymmetric type-2 fuzzy membership functions and application in time series prediction. In: *Proceedings of the sixth international conference on machine learning and cybernetics, ICMLC 2007*, vol 4, pp 2024–2030
18. Radosavljević J (2018) *Metaheuristic optimization in power engineering (Energy Engineering)*. The Institution of Engineering and Technology, New York, p 536
19. Sean L (2016) *Essentials of metaheuristics*. Department of Computer Science George Mason University, Fairfax, p 261
20. Smerichevska S et al (2020) Cluster policy of innovative development of the national economy: integration and infrastructure aspects. *Wydawnictwo naukowe WSPiA, Poznan*, p 380

21. Subbotin S, Oliinyk A, Levashenko V, Zaitseva E (2016) Diagnostic rule mining based on artificial immune system for a case of uneven distribution of classes in sample. *Communications* 3:3–11
22. Talbi E-G (2009) *Metaheuristics: from design to implementation*. Wiley & Sons, Hoboken, New Jersey, p 618
23. Yang X-S (2018) *Nature-inspired algorithms and applied optimization*. Springer International Publishing, p 330
24. Yang X-S (2018) *Optimization techniques and applications with examples*. Wiley & Sons, Hoboken, New Jersey, p 364
25. Yu X, Gen M (2010) *Introduction to evolutionary algorithms*. Springer, London, p 433

A Testing Methodology for the Internet of Things Affordable IP Cameras



Grazyna Dzwigala, Baraq Ghaleb, Talal A. Aldhaheeri , Isam Wadhaj, Craig Thomson, and Nasser M. Al-Zidi 

Abstract IP cameras are becoming a cheaper and more convenient option for a lot of households, whether it being for outdoor, indoor security, or as baby or pet monitors. Their security, however, is often lacking, and there is currently no testbed that focuses and evaluates security of the most affordable cameras on the market. The aim in this paper is to propose a methodology to evaluate the security of IP cameras used in households these days. Methodology proposed consists of five phases that evaluate different areas of chosen IP cameras, from default settings, through software and network connections analysis, to web application testing as well as untested firmware. After performing testing on two IP cameras that were taken into account—Wansview Q5 and Tapo C100, it has been concluded that Wansview Q5 is a more secure choice. Main security recommendations based on the results were to allow users to decide on their settings and disable any open ports that are not in use, as well as update the outdated software.

Keywords Cyber security · IP cameras · Pentesting · Internet of Things · IoT attacks

G. Dzwigala · B. Ghaleb · I. Wadhaj · C. Thomson
School of Computing at Edinburgh, Napier University, 10 Colinton Road, Edinburgh E10 5DT,
UK
e-mail: 40346684@live.napier.ac.uk

B. Ghaleb
e-mail: b.ghaleb@napier.ac.uk

I. Wadhaj
e-mail: i.wadhaj@napier.ac.uk

C. Thomson
e-mail: c.thomson3@napier.ac.uk

T. A. Aldhaheeri (✉) · N. M. Al-Zidi
Faculty of Administrative and Computer Sciences, Albaydha University, Albaydha, Yemen
e-mail: talalalthahri@gmail.com

1 Introduction

The Internet of Things (IoT) is a communication paradigm referring to the integration of physical world with computing connecting a range from devices that can be used as virtual assistants such as Google Echo or Amazon Alexa, through smart kitchen appliances, such as toasters and fridges all the way to security systems, including security cameras, doorbells and even locks [1, 2]. They are often created with portability and usability in mind, with limited set up time required [2–7]. The importance of creating a system for evaluation of their security is ever-growing, as they start to encompass every aspect of life [8, 9]. Devices such as IP cameras are increasingly used for not only outdoor security, but also indoor baby or pet monitors. IP cameras have a processing power enabling the on-board analysis of videos that could be also remotely accessed and may offer some security features such as encryption and authentication with the major disadvantage being the cost [3]. The security of these devices emerges as an important issue with users unaware of the potential security risks [5]. However, and despite that such cameras can carry a huge deal of potentially sensitive data, there is not enough work on security of these devices [3]. Hence, this research attempts to create a model for evaluation of security of two low-end IP cameras that are easily available from online retailers. The main question that this paper aims to answer is: Is it possible to create one methodology for testing of multiple devices?

The rest of the paper is structured as follows. Section 2 overviews the related work. Section 3 sheds light on the main attacks IoT devices are vulnerable to and the main countermeasures. Section 4 presents the research methodology followed in this study. We demonstrate the results obtained in Sect. 5 and discuss them in Sect. 6. We conclude the paper in Sect. 7 highlighting lessons learnt and directions for future work.

2 Related Work

Several studies have been conducted with the aim to propose a methodology for analyzing IoT devices. For instance, the authors in [5] outline a methodology for testing the security of a range of IoT devices. The study involves devices such as HDMI sticks, drones, smartwatches, and IP cameras. Tools used for this testbed include Wireshark Kismet and Kali Linux, as well as OpenVAS, Nessus and Nexpose, which can be used for the initial vulnerability scanning. It proves that the ciphers, firmware and even data streams are not encrypted, and it is visible when sniffing traffic using Wireshark and other network monitoring software. By using vulnerability scanner—OpenVAS, the authors prove that tested devices are vulnerable to a number of well-known vulnerabilities. It is, however, a broader study that covers a wide number of devices. In relation to IP cameras, Varol and Abdalla [3] propose a penetration testing methodology for IP cameras, which consists of three main stages.

First phase being ‘Defining the Area’ includes a decision on whether the testing and research be carried out on the whole structure of the device; including the firmware and network connections or if the analysis should solely focus on one feature of the IoT device; example being network connections. Second phase is ‘Implementation of the Process’, which implements the penetration testing techniques outlined in the report. During this phase, the authors collect the data and information that can be found during testing. The third and the last phase is “Outcomes reporting and presentation” which essentially documents all of the findings found in phase two. However, this paper is limited to only penetration testing methodology rather than an evaluation of the security of IoT devices, namely IP cameras.

In [7], the authors tested a wide range of IoT devices. They do indicate that a lot of devices they have tested returned a number of high vulnerabilities, which in the case of our paper is not accurate. They do, however, raised an issue with a HTTP server of the device and the fact that it is prone to possible denial of service attacks and heap overflow. The authors in [8] looked at a number of different security layers for a wider range of IoT devices, and they found a large number of unencrypted information when using Wireshark as a packet sniffer. Information such as model number, manufacturer, hardware and software versions was seen during tests performed on Tapo camera. The paper explored a number of known vulnerabilities, including weak credentials, account lockout and denial of service. Out of the tested devices, baby monitor, which is the closest to the tested IP cameras, came back with a possible denial of service, which was one of the possible attacks to be carried out on the two tested IP cameras in our study.

3 IoT Attacks, Challenges and Countermeasures

3.1 Main IoT Attacks and Challenges

The study in [3] describes several countermeasures and their challenges in relation to IoT devices including IP cameras. It explains how the devices can be exploited on their physical layer, but it also deals with attacks involving social engineering and device hardware which could potentially severely impact privacy of users.

Eavesdropping attacks

One of the most prevalent issues in IoT devices is eavesdropping attacks and their potential impact on the type of data that are leaked to a potential malicious actor. Devices such as smart watches, IP cameras and baby monitors are especially vulnerable to eavesdropping attacks, and the outcome of a successful attack could have dire consequences. Information such as health statistics of the owner of a smart smart-watch, live feeds from baby monitors or IP cameras can be extracted using tools available to virtually anyone. The data are often transmitted via unsecured protocols,

which are fairly easy to crack. Many of them lack cryptographically secure channels between the servers and the devices themselves [3, 6].

Unauthorized Access and Software Update

Unauthorized access is another big issue raised in this article. This is especially prevalent in IoT devices due to default weak credentials, which are rarely changed by the less technologically advanced users. Attackers are fully aware of the credentials being easy to guess which in turn opens those devices to dictionary and brute force password attacks [3, 6]. The authors raise another issue which is software updates. These are often not enforced and in the case of cheaper, affordable IoT devices hardly a priority. Software updates are, on the one hand, a solution to a lot of IoT issues, and, on the other hand, part of a bigger problem. Because of the way IoT devices are inherently updated, they need to be available and connected to the Internet at all times. This capability opens up doors to unauthorized access and exploits at all times. This in turn enables eavesdropping attacks, malicious code injection, node insertion, and so on [3].

3.2 Attack Countermeasures

The study in [4] classifies IoT devices vulnerabilities into three layers: device-based; network-based and software based and shows security impact on the CIA triangle. It then takes presented vulnerabilities and proposes remediation strategy that can be easily implemented. As a part of the strategy, it outlines three main countermeasures as follows.

Access and authentication controls

Access and authentication controls are designed to conquer challenges such as physical security, insufficient resources, insufficient access control and audit mechanisms, as well as inadequate authentication. The attacks that can be prevented with this countermeasure are dictionary attack, device capture, sinkhole attack and battery draining attack [4].

Software assurance and Security Protocols

Software assurance is dealing with insufficient access control and weak programming practices, a prevailing issue when designing IoT devices. Attacks that can be avoided by implementation of software assurance are injection and firmware modifications attacks [4]. Implementing proper security protocols can help avoid attacks such a Sybil attack, device capture and battery draining attacks [4]. These are just main three counter measures outlined in this research. There are many other proposed ways of securing IoT devices, and this research is aiming to do just that.

4 Methodology

4.1 *IP Cameras Investigated*

Wansview Q5

The first device undergoing testing is Wansview Q5 [10], wireless IP camera that is advertised as an affordable solution for both indoor and outdoor security monitoring. The camera connects to an Android or an iPhone application and offers cloud services as an additional service. It also accepts a microSD card that can store the recorded feed locally.

TP-Link Tapo C100

Second device that is tested in this research is TP-Link Tapo C100 wireless IP camera [11]. Similarly, to the first one, it is advertised as an affordable, secure camera for everyday use. It also uses Android and an iPhone application to control the device and see the live feed. It also accepts SD card as a local storage of the feed. According to TP-Link's website, it uses a 128-bit AES encryption that supports SSL/TLS.

4.2 *Lab Considerations*

There is a number of considerations that have to be taken into account before testing the devices, and these are outlined in this section of the report. For the operating system, Kali Linux is one of the most popular, advanced and open-source operating systems that is used by many ethical hackers when performing penetration testing [12, 13]. Because of its versatility and ease of use, it is a main choice when evaluating security of any given systems and devices. This methodology utilizes couple tools available in Kali to test the security of the two IP cameras including Wireshark, Nmap, Hydra, Whois, Nessus, Dirbuster, Curl, and Nikto.

4.3 *Testing Approach*

Phase 1—Default Settings and Policies

Phase 1 consists of information that can be gathered by looking through the iPhone application that controls the IP cameras. This test is evaluating default settings on the cameras and the applications used. This test is largely based on in-app information with questions being asked in this test are:

- What is minimum password length?
- What is the required complexity of the password?

- Is there an expiration date on the password?
- Can the new password be the same as previous password?
- Is password displayed by default?
- How many failed attempts before account lockout?
- Are the default camera account credentials secure?
- What services can be enabled by the user?

Phase 2—Software

The second phase is testing carried out on the software component of the IP cameras. This is utilizing vulnerability scanning with the questions being asked in this test are:

- Is the device exposed to any known vulnerabilities?
- Is the software on the device up to date?
- Is the communication secure?

Phase 3—Network and Packet Capture

Phase 3 is testing network connections to and from the IP cameras and what information can be found based on the results obtained using the tools Nmap, Nmap NSE, Wireshark, and WhoIs with the questions being asked are:

- What is the Operating System of the IP camera?
- What is the Service running of the IP camera?
- What ports are open?
- What known vulnerabilities can be found on the device?
- What networking protocols are used?
- What encryption protocols are used?
- What information can be found in transmitted packets?
- Are there any odd IPs contacted by tested devices?

Phase 4—Web Application

Phase 4 involves web application testing including scanning, responses, directory path traversal, and Brute-forcing web application using, Curl, DirBuster, Hydra, and Nikto with the following questions:

- What response can be seen in the RTSP/HTTP/HTTPS headers?
- Can we access the web application without authentication?
- Can the directories in web application be brute forced?
- What directories are visible?
- Is it possible to access directories without authentication?
- Is the web application vulnerable to exploits available in the scanner?
- Is the server version up to date?
- Are there any version specific issues?
- Is it possible to brute-force credentials on the application?
- Are there any passwords that match with admin/root username?
- Are there any matching credential pairs in the most common credential list?

5 Results and Evaluation

5.1 Phase 1—Default Settings and Policies

When signing up for an account with Wansview application, the details such as email and password have to be filled. The password required must be between 9 and 16 characters and include at least one of numbers, uppercase and lowercase letters as well as special characters. There is also an email verification required, which adds a level of security on top of the complex password. Local account default credentials on Wansview IP cameras are arguably secure, with a random default username and random string of complex characters as a password. Table 1 outlines the answers to questions asked in methodology for both sign up and local accounts.

Similarly, to Wansview camera, Tapo C100 has two logins. For the sign-up account, Tapo camera requires an email address and a password that is between 8 and 32 characters, that include at least two of letters, numbers and symbols. There are no historical checks, meaning the same password can be used more than ones and the password does not expire. The only requirement for this account is that both username and password have to be between 6 and 32 characters. Tapo camera does not have default credentials for local account. It does, however, allow a user to create a camera account for third-party applications, such as NAS solutions. The only requirement for this account is that both username and password have to be between 6 and 32 characters. There is no requirement for any complexity. Table 2 shows the answers to the methodology questions for Tapo C100 camera.

In relation to services, Wansview has a number of security features that can be enabled. The local feed of the camera can be accessed in a browser or VLC player when a user navigates to a specific address. It can, however, be changed to any port between 554 and 1544, which would add a layer to its security. The possibility of randomizing this port is a good security feature as it would make it harder for the

Table 1 Password policy—Wansview

Questions	Sign up account	Local account
What is minimum password length?	At least 9 characters	No minimum
What is the required complexity of the password?	Lowercase, uppercase, number and special characters	No complexity required
Is there an expiration date on the password?	No	No
Can the new password be the same as previous password?	Yes	Yes
Is password displayed by default?	No	No
How many failed attempts before account lockout?	No account lockout	No account lockout

Table 2 Password policy—Tapo

Questions	Sign up account	Local account
What is minimum password length?	At least 8 characters	At least 6 characters
What is the required complexity of the password?	Two out of letters, numbers and symbols	No complexity required
Is there an expiration date on the password?	No	No
Can the new password be the same as previous password?	Yes	Yes
Is password displayed by default?	No	No
How many failed attempts before account lockout?	No account lockout	No account lockout

Table 3 Default settings

Question	Wansview	Tapo
Are the default camera account credentials secure?	Yes, but can be changed by the user with no complexity required	Yes
Is privacy mode enabled by default?	No privacy mode available	No directories
What security can be enabled by the user?	Can specify port for RTSP Can specify port for ONVIF	No

malicious factor to access live feed. That is of course assuming that they are already on the local network. There is no privacy mode that can be enabled. However, there is a possibility to enable ONVIF in case a user wants to expand their IP camera coverage. This protocol ensures standardized deployment of IP cameras from different vendors. The protocol itself does not really support any security features; Wansview camera, however, allows the user to choose their own port number between 8000 and 9000. It also enables verification by default using the same credentials as the local account. Table 3 outlines the answers to the questions asked in the methodology section for both cameras.

5.2 Phase 2—Software

When Nessus scanner is run on the Wansview camera, it comes back with 14 informational vulnerabilities. The scan revealed three main HTTP issues, which include information on what methods are allowed, server type and version as well as protocol information. It also detected the HTTP version—lighttpd 1.4.52 on the web server.

Table 4 Software vulnerability scans

Question	Wansview	Tapo
Is the device exposed to any known vulnerabilities?	CVE-1999-0524	CVE-1999-0524, CVE-2004-2761, CVE-2016-2183, CVE-2012-4929, CVE-2012-4930, CVE-2013-2566, CVE-2015-2808
Is the software on the device up to date?	No, the latest lighttpd is 1.4.59	No, Linux Kernel is running version 2.6
Is the communication secure?	No information found	No, supports TLS1.0

Nessus uses a SYN port scan, which shows three ports that are open: 80, 554 and 65,000, which corresponds to nmap results presented later in this chapter. It also highlights information that the potential attacker can gain by simply scanning this target, such as the type of the device, ethernet card manufacturer and MAC address. Additionally, it informs of timestamps supported for both ICMP and TCP/IP protocols, which could potentially give an attacker knowledge of time set on the target machine. That in turn could aid them in performing attacks on time-based authentication measures. This vulnerability was published under CVE-1999-0524. Tapo C100 has 21 vulnerabilities detected, 7 of which are medium, 2 are low and the rest are informational. Six of the medium and one low vulnerability are found in SSL service. The main issue here appears to be with the self-signed and untrusted SSL certificate for the web server which is over port 443—HTTPS. Meaning that if a possible malicious factor was to sniff enough traffic encrypted with this cipher, they would be able to decipher the stream as plain text. There is also a low vulnerability in TLS that could potentially be vulnerable to CRIME attack, which can occur due to compression when host uses both HTTPS and SPDY protocol. Table 4 outlines the answers to the questions asked in the methodology section for both cameras in relation to vulnerability scans.

5.3 Phase 3—Network Information and Packet Capture

Nmap is used for this test. To determine the operating system that was run on the Wansview camera -O flag in nmap was used. The operating system was determined to be Linux 2.6.32–3.10, which corresponds to Nessus’s findings. To determine the version of services running on the open ports -sV flag was used. The HTTP web server was determined to be running lighttpd 1.4.52. There is a number of vulnerabilities found when running this script. Most of which reside in lighttpd version of the HTTP web server. Similarly, first test carried out was on the main network to determine what was the IP of **Tapo** camera which was found to be IP is 192.168.0.33. For

Table 5 Network information

Question	Wansview	Tapo
What is the Operating System of the IP camera?	Linux 2.6.32–3.10	Unknown using nmap -O
What is the Service running of the IP camera?	lighttpd 1.4.52	Nagios-NCA DoorBird gSOAP 2.8
What ports are open?	80 554 65,000	443, 554 2020 8800
What known vulnerabilities can be found on the device?	CVE-2010-0295, CVE-2008-1531, CVE-2008-0983	Inconclusive

Tapo camera, the same flag was used: -O. Nmap could not determine the operating system. To determine the version of services running on the open ports -sV flag was used. The HTTPS web server was determined to be running Nagios NSCA. The service version for RTSP port is DoorBird, and two other ports that are open are running gSOAP 2.8 and no information on port 8800. This scan also gives the tester information of what kind of device the tester is scanning. The vulnscan.nse was run on Tapo C100 camera, and it returned a large number of results. Nagios service appears to have a large number of vulnerabilities that could be due to version found by nmap—Nagios NSCA. However, there is no exact version determined, and the results appear to be showing all vulnerabilities for all of the versions and vendors, including Cisco, Siemens and Google. Table 5 shows the answers to the questions asked in methodology for both cameras in relation to network information.

In respect to Wansview packet capture, Wireshark and WHOIs are used for this test. This test is divided into two main parts. First one is inspecting traffic between the camera itself and an app that is controlling it and the second one is checking what traffic the IP camera itself produces. Main protocol used to send live feed and any changes to the settings between the camera and the iPhone app was UDP. Information such as camera ID, access key and user agent is visible when sniffing the network. Packets exchanged between the two devices also include ICMP. When followed, the stream gives information on content type, user agent and host. For the Tapo camera, the main protocol for the live feed and app control is over TCP between port 8800 on the camera and port 60,060 on the phone. It also uses a number of UDP packets between random ports. Similarly, to the previous camera, Tapo POSTs information over HTTP to /stream. When the HTTP stream is followed, there is a number of data disclosed. Information available on client side is its model and UUID. Data such as camera name, nonce, what hashing algorithm is used—MD5, what cipher is used—AES128 and padding are also visible. Additionally, it transports username in plain text—“admin”. When setting up the camera, the first UDP packet from camera to application contains information such as device id, type and its mac address. It also shows the hardware and firmware versions as well as AES key. Similarly, to

Table 6 Packet capture

Question	Wansview	Tapo
What connection protocol is used? TCP or UDP?	UDP	TCP
What networking protocols are used?	HTTP, ICMP, IGMPv3, SSDP, DNS, DHCP	HTTP, ICMP, SSDP, DHCP, DNS, IGMPv2, IGMPv3, NTP, MPEG
What encryption protocols are used?	TLSv1.2	TLSv1.2
What information can be found in transmitted packets?	Camera ID, access key and user agent	Client model, Client UUID, camera name, hashing algorithm (MD5), cipher (AES128), username (admin), device id, device type mac address, hardware version, firmware version, AES key
Are there any odd IPs contacted by tested devices?	Digitalocean which is a cloud provider for Wansview	AWS IP addresses

Wansview, ICMP packets are used for host unreachable and echo information. Tapo also uses TLSv1.2, between 443 and 59,661 port, and the handshake is visible. Most information is scrambled, however, plaintext information that can be seen in TCP stream of TLSv1.2. The camera was found to contact 34.252.183.71, Amazon AWS via TCP protocol. The motion alerts are also visible in the traffic. They show up as an Encrypted alert, through TLSv1.2 to AWS IP address. For more detail refer Table 6.

5.4 Phase 4—Web Application

Web Application Responses

First test is for RTSP response, which checks for the options available when checking the RTSP port, which was determined to be open in tests before. Because the response is 200 OK, it can be assumed that it is possible to access the video parameters with no authentication. With the ‘DESCRIBE’ flag set, it also lists possible parameters to use in further penetration testing.

For the Tapo camera, the same test was carried out on the open RTSP port for Tapo camera. Similarly, it comes back with 200 OK response, however, gives more options that can be used for injection attacks. Three more curl commands are run on the system in order to determine the response of the web application; these, however, outline weak SSL certificate. A summary of the two cameras web application responses is shown in Table 7.

Table 7 Web application responses

Question	Wansview	Tapo
What response can be seen in the RTSP/HTTP/HTTPS headers?	200 OK 200 OK Failed to connect	RTSP: 200 OK HTTPS: curl failed
Is it possible to access the web application without authentication?	Yes, RTSP	Yes, RTSP

Table 8 Directory path traversal

Question	Wansview	Tapo
Can the directories in web application be brute forced?	No	Yes
What directories are visible?	/media	No directories
Is it possible to access directories without authentication?	No	No
What are HTTP/HTTPS responses?	403/500	N/A

Directory path traversal

For this test, Dirbuster was used with default small wordlist available in Kali. To get a sample of directories available in Wansview IP camera, Dirbuster was run on the web server on port 80. After a number of tests, the webserver kept coming offline and timing out GET requests, which in itself is a security issue, as there is no denial-of-service protection. That being said it, when disabling features such as auto-switch between HEAD/GET requests, choosing not to brute-force files and disabling recursive option, the only directory found was /media, with response of 500. When testing Tapo cameras HTTPS web server using both directory and file option, there was no results found on the wordlist used. A summary of the two cameras for the directory path traversal responses is shown in Table 8.

Web Application Scanning

It is assumed that when performing application scanning using Nikto, it should come back with similar results to Nessus. Nikto script returned similar results to the curl command with the allowed HTTP methods. It also highlights issues with headers that could result is XSS, clickjacking and injection attacks. When scanning the web service on the Tapo camera, we change the protocol to HTTPS. Nikto script ran on Tapo camera's HTTPS port returned a large number of results. It did, however, return several interesting directories available including SQL, access, admin, information and country code directories. There is also a number of possible admin login sections outlined. A summary of the two cameras of the web application scanning responses is shown in Table 9.

Table 9 Web application scanning

Question	Wansview	Tapo
Is the web application vulnerable to exploits available in the scanner?	XSS, clickjacking and injection attacks	Results inconclusive
Is the server version up to date?	No	No banner available
Are there any version specific issues?	No	Results inconclusive

Brute-forcing web application

The first test involved testing of the HTTP port on Wansview camera using Hydra. It was run using two precomputed wordlists of credentials and passwords. The flags used are:

- **-e** to specify null password and login as password
- **-f** to decrease used resources and exit when credential pair was found, and
- **-V** to see all attempted pairs.

For Tapo camera, the same command was ran, with the changed https-get request in order to again accommodate for the HTTPS server. When using the same wordlists, credential pair of root/root was found. When using ‘admin’ as a username, password ‘login’ and ‘admin’ was matched. When using ‘root’ as a username, the password ‘root’ is confirmed. A summary of the two cameras of the web application scanning responses is shown in Table 10.

Table 10 Brute-forcing web application

Question	Wansview	Tapo
Is it possible to brute-force credentials on the application?	Yes, no errors found, no lock out initiated	Yes, no errors found, no lock out initiated
Are there any passwords that match with admin/root username?	root/root. root/password. root/123456 admin/admin. admin/password. admin/123456	root/root
Are there any matching credential pairs in the most common credential list?	Yes root/root	admin/admin

6 Results Discussions

6.1 *Phase 1 and Phase 2*

Phase 1 was evaluating default settings and policies. When it comes to the default policies, Wansview camera appears to be more secure. The sign-up password policies are quite strong for both, given the minimum length and complexity requirements. Wansview, however, adds another layer by requiring verification for the email address that was signed up. That is of course assuming that their previous password was part of a breach. When it comes to a local account policy, both cameras suffer from possible weak credentials vulnerability, with no complexity or length required. When it comes to services that can be enabled on both devices, Wansview is also the one that could be considered more secure. There is no privacy mode available for this one, and the local camera feed can be accessed using possibly weak credentials; however, that can be secured further by specifying a port for RTSP itself which is an option that can be enabled. Phase 2 involved testing the IP cameras themselves using a vulnerability scanner—Nessus. In this phase, it was also determined that Wansview is more secure out of the two. The issues found in Tapo camera, all appear to be in the encryption of the traffic between the camera itself and the web server. It supports a number of weak ciphers that are known to have a number of collisions which means that the potential attacker would be able to decrypt the traffic and possibly see the feed between the devices. Though not enabled, it also supports TLS1.0, which has a number of issues in itself and can be used to perform man in the middle attack. Both cameras have a number of informational vulnerabilities that can be used in information gathering phase of any possible attack.

6.2 *Phase 3 and Phase 4*

Phase 3 involved testing of the network capability of both cameras. Tools such as Nmap and Wireshark were used. For investigation using nmap, both of the cameras showed a number of open ports with Wansview having three and Tapo having four. Nmap also highlights a number of issues in the web server itself. For Tapo camera, the results turned out to be more inconclusive. That being said, port that was open for developer toolkit, did run a version of gSOAP, which is prone to couple of denial-of-service attacks. The amount of information that can be found by simply sniffing the network is excessive. Such details as the device used to control the camera itself are available, including model and UUID. Encryption information, such as hashing algorithm and cipher, can also be seen. Username also appears to be transported in plain text over the network, which does aid in further brute force attacks. Phase 4 consisted of web application testing, using a number of open-source tools in Kali. When testing Tapo HTTPS server, there were no directories found, but the search did not time out

or deliver any errors in the process. Testing of the vulnerabilities on the web server using nikto, highlighted a number of possible issues for Wansview camera. It was determined that due to the way headers of the server are configured, it is potentially vulnerable to XSS, clickjacking and content injection attacks. When testing Tapo camera's web server, a similar issue that arose during nmap investigation emerged. Due to the server not having a clear version, the results for vulnerabilities found were deemed inconclusive. However, it did find a number of interesting directories that could potentially be used to carry out a number of attacks.

6.3 Security Recommendations

There is a couple of security recommendations for both user and a manufacturer that can be suggested after performing these tests. One of the most basic and easiest to implement security improvements is requiring a user to have complicated password that is at least eight characters long and multifactor authentication [14–17]. One of the considerations for manufacturers would be completely disabling and not supporting older encryption protocols as those can be used in various man in the middle attacks [18]. Setting a locking out option when several failed attempts is tried, can also help with such attacks. Updating the operating system and closing unused ports, are very basic security improvements that can be done on both manufacturer and the customer level. Especially for Tapo camera, port 2020, used for developer's toolkit, does not necessarily need to be open at all times. To build on this research, phase 5, or a firmware testing could be carried out, to further determine possible issues with the cameras on the physical layer. This could also include further web application and the iPhone application testing.

7 Conclusion and Future Work

The aim of this study was to create a methodology that can be used to test two low-end IoT devices—IP cameras. The proposed method of testing included overall four phases as mentioned earlier. The proposed testbed has reached the aim of the study, with both cameras showing different results, which allowed for forming of a conclusion that Wansview IP camera is a more secure choice. There is of course, room for further research. For instance, testing the firmware is another possible phase in testing of the IP cameras. In addition, carrying out proposed phase 5 would improve the methodology in itself. Dedicated vulnerability analysis and application testing itself would definitely be the next steps in evaluating the testing of the IP cameras.



References

1. Atzori L, Iera A, Morabito G (2010) The Internet of Things: A survey. *Comput. Networks.* 54:2787–2805
2. Number of Internet of Things (IoT) connected devices worldwide from 2019 to 2030, <https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide>, last accessed 2021/07/12.
3. Alladi T, Chamola V, Sikdar B, Choo KR (2020) Consumer IoT: Security Vulnerability Case Studies and Solutions. *IEEE Consum. Electron. Mag.* 9:17–25. <https://doi.org/10.1109/MCE.2019.2953740>
4. Neshenko N, Bou-Harb E, Crichigno J, Kaddoum G, Ghani N (2019) Demystifying IoT Security: An Exhaustive Survey on IoT Vulnerabilities and a First Empirical Look on Internet-Scale IoT Exploitations. *IEEE Commun. Surv. Tutorials.* 21:2702–2733
5. Tekeoglu, A., Tosun, A.S.: Investigating Security and Privacy of a Cloud-Based Wireless IP Camera: NetCam. In: 2015 24th International Conference on Computer Communication and Networks (ICCCN). pp. 1–6 (2015).
6. Abdalla, P.A., Varol, C.: Testing IoT Security: The Case Study of an IP Camera. In: 2020 8th International Symposium on Digital Forensics and Security (ISDFS). pp. 1–5 (2020).
7. Tekeoglu, A., Tosun, A.S.: A Testbed for Security and Privacy Analysis of IoT Devices. In: 2016 IEEE 13th International Conference on Mobile Ad Hoc and Sensor Systems (MASS). pp. 343–348 (2016).
8. Lally G, Sgandurra D (2018) Towards a Framework for Testing the Security of IoT Devices Consistently. In: Saracino A, Mori P (eds) *Emerging Technologies for Authorization and Authentication*. Springer International Publishing, Cham, pp 88–102
9. Ghaleb B, Al-Dubai A, Ekonomou E, Qasem M, Romdhani I, Mackenzie L (2019) Addressing the DAO Insider Attack in RPL's Internet of Things Networks. *IEEE Commun Lett* 23:68–71. <https://doi.org/10.1109/LCOMM.2018.2878151>
10. Wansview WiFi IP Camera, <https://www.wansview.com/cn/proinfo.aspx?proid=29&categoryid=4&aids=2#Spec>, last accessed 2021/04/10.
11. Tapo C100 | Home Security Wi-Fi Camera | TP-Link United Kingdom, <https://www.tp-link.com/uk/home-networking/cloud-camera/tapo-c100/#specifications>, last accessed 2021/03/07.
12. Kali Linux explained: A pentester's toolkit, <https://www.csoonline.com/article/3528191/kali-linux-explained-a-pentester-s-toolkit.html>, last accessed 2021/03/13.
13. Housani, B. Al, Mutrib, B., Jaradi, H.: The Linux review - Ubuntu desktop edition - version 8.10. In: 2009 International Conference on the Current Trends in Information Technology (CTIT). pp. 1–6 (2009).
14. Bošnjak L, Brumen B (2019) Rejecting the death of passwords: Advice for the future. *Comput Sci Inf Syst* 16:313–332. <https://doi.org/10.2298/CSIS180328016B>
15. Ometov, A., Bezzateev, S., Mäkitalo, N., Andreev, S., Mikkonen, T., Koucheryavy, Y.: Multi-Factor Authentication: A Survey. *Cryptography.* 2, (2018). <https://doi.org/10.3390/cryptography2010001>.
16. Ometov A, Petrov V, Bezzateev S, Andreev S, Koucheryavy Y, Gerla M (2019) Challenges of Multi-Factor Authentication for Securing Advanced IoT Applications. *IEEE Netw* 33:82–88. <https://doi.org/10.1109/MNET.2019.1800240>

17. Astaburuaga, I., Lombardi, A., Torre, B. La, Hughes, C., Sengupta, S.: Vulnerability Analysis of AR.Drone 2.0, an Embedded Linux System. In: 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), pp. 666–672 (2019).
18. Why Closing Unused Server Ports is Critical to Cyber Security, <https://blog.getcryptostopper.com/why-closing-unused-server-ports-is-critical-to-cyber-security>, last accessed 2021/04/02.

Detecting Equatorial Plasma Bubbles on All-Sky Imager Images Using Convolutional Neural Network



Worachai Srisamoodkham, Kazuo Shiokawa , Yuichi Otsuka ,
Kutubuddin Ansari, and Punyawee Jamjareegulgarn

Abstract This paper proposes initially to apply convolutional neural network (CNN) for detecting the equatorial plasma bubbles on the ASI images. The considered CNN model is the YOLO v3 tiny model under a deep learning API (Keras), running on top of the machine learning platform (TensorFlow). Our program for EPB detection is written in Python that is extended easily to combine into a space weather web site for detecting and notifying EPBs in our next step. The results show that the YOLO v3-based CNN can detect the EPBs in ASI images with different intensities obtained from many countries. The threshold is tested and selected to be 0.40 suitably for detecting the anomaly (EPB existence). The maximum anomalous value is selected to decide the EPB occurrence.

Keywords ASI · Convolution neural network · Plasma bubble · YOLO

W. Srisamoodkham

Faculty of Agricultural and Industrial Technology, Phetchabun Rajabhat University, Sadiang, Thailand

e-mail: hs5xij@pcru.ac.th

K. Shiokawa · Y. Otsuka

Institute for Space-Earth Environmental Research, Nagoya University, Nagoya, Japan

e-mail: shiokawa@nagoya-u.jp

Y. Otsuka

e-mail: otsuka@isee.nagoya-u.ac.jp

K. Ansari

Integrated Geoinformation (IntGeo) Solution Private Limited, New Delhi, India

P. Jamjareegulgarn (✉)

King Mongkut's Institute of Technology Ladkrabang, Prince of Chumphon Campus, Chumphon, Thailand

e-mail: kjpunyaw@gmail.com

1 Introduction

Plasma instabilities over equatorial ionosphere can be a major source of large- and small-scale density depletions during after sunset and after midnight. The depleted plasma over magnetic equator (so-called equatorial plasma bubble or EPB) formulates at the bottomside F region and rises upwardly with its structure elongating along the magnetic field lines. The plasma instabilities exist various scales ranging from 10 cm to 1000 km where they can disrupt HF communication, satellite communication, positioning, as well as navigation systems within $\pm 20^\circ$ latitudes around geomagnetic equator [1]. In general, these perturbed ionosphere conditions can lead to another phenomenon named as equatorial spread-F (ESF) in F region, because they affect directly the HF communications by producing the echo spreads in ionograms. Both EPB and ESF have been known as the main sources for Global Navigation Satellite System (GNSS) disturbances. The scintillation is the amplitude and phase fluctuations of signals that leads to disrupt satellite-based communications and deteriorate the GNSS positioning accuracy [2, 3]. The main reason is that the sudden density depletions inside EPBs disturb the GNSS velocities passing the ionosphere. This is why the occurrence characteristics as well as probabilities of EPB have been studied for space weather and ionospheric physics. (e.g., [4, 5]).

Basically, the F region plasma irregularities at height 250–350 km can be observed as dark EPBs and bright plasma blobs. The plasma blobs were observed firstly by OI 630.0 nm all-sky imagers (ASIs) at Brazil [4]. Nade et al. [5] investigated the simultaneous plasma blobs and EPBs over low latitudes, but the generation mechanism of EPBs and blobs is not obviously comprehended [6] and should be made additional investigation. Paznukhov et al. [7] studied firstly the EPBs and the scintillations over Africa in 2010 for monitoring the ionospheric irregularities. Their results released that the EPBs are directly related to the scintillations and the scintillation severity relies on EPB depth. The scintillation amplitude is identified by S4 index, and the EPBs are analyzed based on spectral analysis and GPS TEC observation. Shiokawa et al. [8] conducted the experiments of atmospheric and ionospheric waves in the upper atmosphere over several countries using ASIs. Their results reported about the features of small gravity waves and medium disturbances in mesosphere, thermosphere and ionosphere. As for our earlier EPB investigations, the obvious airglow depletions incurred by EPBs can be observed by several OMTIs and analyzed at Chiang Mai, Darwin, and Kototabang etc. After storing the ASI images, they will be post-processed and analyzed with some kinds of program such as MATLAB, SCILAB, etc.

Likewise, numerous literatures have proposed several methods to analyze the all-sky image data. For example, Kubota et al. [9] introduced a method to convert the pixel ASI images into the actual coordinates at the airglow emission layer. Afterward, Narayanan et al. [10] present an approach to convert the pixel values into the respective latitude–longitude values of each ASI image. In Thailand, the ASIs of optical mesosphere thermosphere imager were also installed at Chiang Mai and Chumphon provinces to monitor plasma bubbles. The ASI images of these two regions are very

important so as to investigate the EPB generation mechanism, the EPB movement and the impact of EPB on HF communication, positioning, and navigation over equatorial and low latitudes [11]. However, the EPB detection and notification have not proposed simultaneously; therefore, the authors have an idea to detect the nighttime ASI images and classify each ASI image with or without EPBs on web applications using convolution neural network (CNN). The ASI images at Chiangmai, Thailand, is employed as the train dataset for the proposed CNN method. Meanwhile, the ASI images are also taken from previous published manuscripts to be the test dataset such as Lynn et al. [15], observed at Darwin, Australia; Takahashi et al. [16], observed at São Luis, Brazil; and Makela et al. [17], observed at Haleakala, Hawaii [17].

2 Optical Mesosphere Thermosphere Imagers (OMTI)

Optical Mesosphere Thermosphere Imagers (OMTI) was constructed by ISEE of Nagoya University in 1997 in order to investigate the dynamics of airglow emissions in upper atmosphere. The OMTI consists of all-sky imagers (ASIs), photometers, and interferometer. The imagers employ some cooled CCDs of 512×512 pixels. All of the ASIs in the OMTI have at least four filters and some gases' filters. The BPF bandwidths are about 1–2 nm and the ASI sensitivities are less than 0.4 counts per second providing smaller than 4000 count/R/s. Further details of the OMTI can be read and studied in Shiokawa et al. [8], and the airglow images of OMTI are obtained from the web: <https://stdb2.isee.nagoya-u.ac.jp/omti/index.html>. Otsuka et al. [9] suggested that the ASIs of OMTI are a crucial instrument for better understanding the coupling between ionosphere and thermosphere and detecting the EPBs. Figure 1 shows the examples of airglow images detected by OMTI at Chiang Mai, Thailand, on February 2, 2020.

3 YOLO Tool

Object detection is a significant mission that is concerned to identify the existence and the localization of one or more objects in a given figure. The methods of object recognition and classification seem to be the challenging tasks. Hence, the YOLO with convolutional neural networks (CNNs) approach has been proposed to be the modern tool for performing the real-time object detection [12]. That is the reason why the YOLO is selected to detect the real-time EPBs from ASI images in this work.

YOLO tool was built and released to the public in April, 2018. It is recognized to outperform the previous YOLO versions. Its algorithm depends on a variant of Darknet which has 53 hidden-layer network trained on Imagenet. The latest version of YOLO is YOLOv3. In this work, a CNN program is coded with python using YOLOv3 model that is contained in Keras API and TensorFlow. Note that TensorFlow

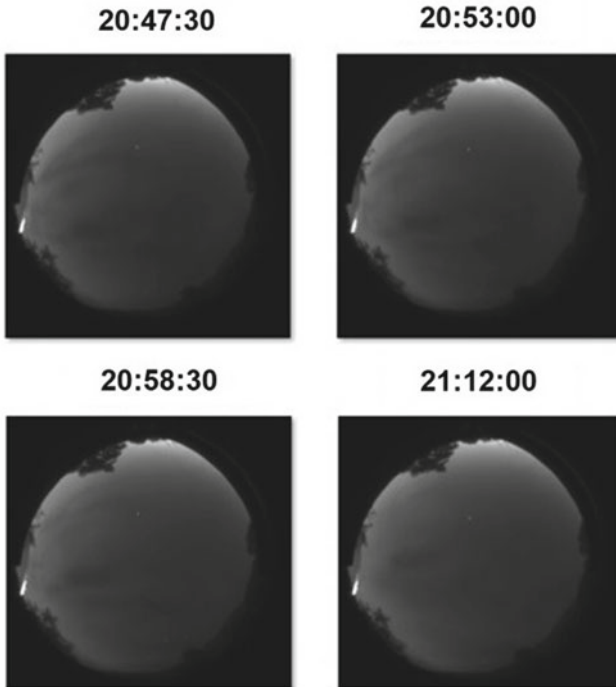


Fig. 1 The airglow images detected by all-sky airglow imagers of OMTI at Chiang Mai, Thailand, on February 2, 2020

is the open-source of Google for developing the applications based on machine learning and deep learning. It can be used on several platforms (e.g., windows, Linux, etc.) for making machine learning.

As for YOLO v3, a fully convolutional neural network (CNN) with larger than 100 layers can be processed for its object detection due to 53-hidden layers. The operating procedure of YOLO v3 is depicted in Fig. 2 where YOLO v3 includes the down-sampling three levels for the input image dimensions. Especially, the object prediction of YOLO v3 are forecasted using logistic regression [13]. As for several advantages of YOLO v3, it is thus employed to train the all-sky imager (ASI) images and classify those images with or without plasma bubbles in this work.

4 Results

The ASI images with and without EPBs at Chiang Mai, Thailand, were used as the training images. Afterward, those images were extracted the image features and classified with and without EPBs using YOLOv3 (CNN model). YOLOv3 tiny model

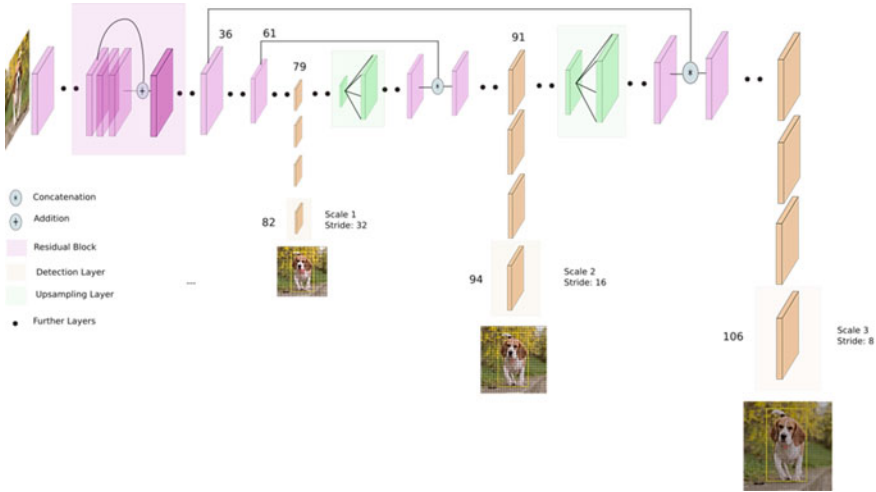


Fig. 2 The operating procedure of YOLO v3 [13]

was employed in this work, because it has a slightly lower recognition accuracy, but runs faster as compared to the standard model. The accuracy and the computational power of the tiny model are equal to 33.1 and 5.56 Bn, respectively, that are much less than those of the standard model [14]. During training the number of ASI images, we find that the suitable threshold is equal to 0.40 (or 40%) for classifying the EPBs and also use the word “anomaly” in each image to represent the EPB occurrence. In Table 1 case (a)–(d), the ASI images with EPBs show their anomalies of greater than and equal to 40%. The maximum anomalies in each ASI image are selected to decide surely the EPB existence. In contrast, the case e) of Table 1 depicts an ASI image without EPBs whose anomaly is less than 40% (no anomaly).

5 Conclusion

Equatorial plasma bubble (EPB) in all-sky imager (ASI) images are detected using the convolutional neural network (CNN) for the first time. The YOLO v3 “Tiny” model is used to detect the EPBs due to its several benefits. Our EPB detection program is written in Python that can be gathered into a web site at once. After gathering the ASI images from different countries and Chiang Mai, Thailand, we start extracting the features and classify the ASI images with and without EPBs like “supervised learning” with YOLO v3-based CNN. Note that the YOLO v3 framework is based on Keras API (deep learning) operated on TensorFlow (machine learning) platform. We find that the CNN model can be used to detect admirably the EPBs in ASI images with the suitable threshold setting of 0.40. This threshold was defined after more than two hundred ASI images with and without EPBs were trained and

Table 1 ASI Images before and after CNN model with maximum anomalies and sources

Case	ASI Images Before CNN model	ASI Images After CNN model	Maximum of anomalies	Image Sources
(a)			54%	Lynn et al. [15], observed at Darwin, Australia
(b)			45%	Takahashi et al. [16], observed at São Luis, Brazil
(c)			53%	Makela et al. [17], observed at Haleakala, Hawaii
(d)			43%	Chiang Mai, Thailand, observed by Nagoya University
(e)			No anomaly	Chiang Mai, Thailand, observed by Nagoya University

classified completely. In the future, this CNN-based EPB detection program will be combined on the space weather web site and will be used as an EPB precursor over Thailand.

Acknowledgements This research is funded by BTFP organization (project code: B2-001/6-2-63). The authors would like to express the gratitude to the ASI images taken from [15–17]. Particularly, several ASI images employed to train the CNN model were obtained from Chiang Mai station owned by Nagoya University, Japan.

References

1. Woodman RF, Lahoz C (1976) Radar observations of F region equatorial irregularities. *J Geophys Res* 81:5447–5466
2. Datta-Barua S, Doherty PH, Delay SH, Dehel T, Klobuchar JA (2010) Ionospheric scintillation effects on single and dual frequency GPS positioning. In: Proceedings of the 2010 institute of navigation ION GNSS meeting, Portland, OR
3. Carrano CS, Groves KM (2010) Temporal decorrelation of GPS satellite signals due to multiple scattering from ionospheric irregularities. In: Proceedings of the 2010 institute of navigation ION GNSS meeting, Portland, OR
4. Pimenta AA, Sahai Y, Bittencourt JA, Rich FJ (2007) Ionospheric plasma blobs observed by OI 630 nm all-sky imaging in the Brazilian tropical sector during the major geomagnetic storm of April 6–7, 2000. *Geophys Res Lett* 34(2)
5. Nade DP et al (2014) Observations of plasma blobs by OI 630 nm using ASI and photometer over Kolhapur, India. *Earth Moon Planet* 112(1–4):89–101
6. Choi HS, Kil H, Kwak YS, Park YD, Cho KS (2012) Comparison of the bubble and blob distributions during the solar minimum. *J Geophys Res Space Phys* 117(4)
7. Paznukhov VV et al (2012) Equatorial plasma bubbles and L-band scintillations in Africa during solar minimum. *Ann Geophys* 30(2012682):675–682
8. Shiokawa K, Otsuka Y, Ogawa T (2009) Propagation characteristics of nighttime mesospheric and thermospheric waves observed by optical mesosphere thermosphere imagers at middle and low latitudes. *Earth Planet Space* 61:479–491. <https://doi.org/10.1186/BF03353165/>
9. Otsuka Y, Shiokawa K, Ogawa T, Yokoyama T, Yamamoto M, Fukao S (2004) Spatial relationship of equatorial plasma bubbles and field-aligned irregularities observed with an all-sky airglow imager and the equatorial atmosphere radar. *Geophys Res Lett* 31(20)
10. Kubota M, Fukunishi H, Okano S (2001) Characteristics of medium- and large-scale TIDs over Japan derived from OI 630-nm nightglow observation. *Earth Planets Space* 53(7):741–751
11. Lakshmi NV, Gurubaran S, Emperumal K (2009) Imaging observations of upper mesospheric nightglow emissions from Tirunelveli (8.7°N). *Indian J Radio Space Phys* 38(3):150–158
12. How to Perform Object Detection with YOLOv3 in Keras. <https://machinelearningmastery.com/how-to-perform-object-detection-with-yolov3-in-keras/>. Last accessed 29 Nov 2020
13. What's new in YOLO v3? <https://towardsdatascience.com/yolo-v3-object-detection-53fb7d3bfe6b>. Last accessed 29 Nov 2020
14. YOLOv3 (2020) A machine learning model to detect the position and type of an object. <https://medium.com/axinc-ai/yolov3-a-machine-learning-model-to-detect-the-position-and-type-of-an-object-60f1c18f8107>. Last accessed 29 Nov 2020
15. Lynn KJW, Otsuka Y, Shiokawa K (2011) Simultaneous observations at Darwin of equatorial bubbles by ionosonde-based range/time displays and airglow imaging. *Geophys Res Lett* 38(L23101)
16. Takahashi H et al (2018) Equatorial plasma bubble seeding by MSTIDs in the ionosphere. *Progr Earth Planet Sci* 5(32)
17. Jonathan JM, Ledvina BM, Kelley MC, Kintner PM (2004) Analysis of the seasonal variations of equatorial plasma bubble occurrence observed from Haleakala, Hawaii. *Ann Geophys* 22(9)

A Comparative Study of Traditional Bank A and Digital Bank B from an Organizational Innovation Perspective



Easwaramoorthy Rangaswamy , Naresh Nadipilli, and Nishad Nawaz 

Abstract The digital insurgency is troublemaking the association between banks and their customers, and new features constantly seem to upgrade client experience. A digital transformation procedure comprises of innovative technology application developments, just as company aspects, for example C-suite executives uphold, digital transformation strategy, product innovations, HR systems and organizational changes. The practical implications of this research study show digital bank customers are more satisfied than traditional bank customers during COVID-19, digital banking services are cheaper than traditional banking services, product innovation plays a major role in digital transformation and also note that most of the customers are preferred to use digital bank services rather than Internet banking, POS terminals, credit cards, debit cards and ATM machines.

Keywords Traditional banking · Digital banking · FinTech · Digital transformation · Organizational innovation · Customer satisfaction

1 Introduction

During the past recent decade, there has been a rise in the number of organizations that have been presenting the idea of digital transformation. Digital transformation involves not just the utilization of new innovative technologies (for example, data analytics, machine learning, artificial intelligence and Internet of things), yet in addition the progressions of the key business components, including strategy, operating

E. Rangaswamy

Amity Global Institute, 101 Penang Road, Singapore 238466, Singapore
e-mail: erangaswamy@singapore.amity.edu

N. Nadipilli

Credit Agricole Corporate Investment Bank, Singapore, Singapore
e-mail: resh.nadipilli@ca-cib.com

N. Nawaz (✉)

Department of Business Management, College of Business Administration, Kingdom University, Riffa, Kingdom of Bahrain
e-mail: n.nawaz@ku.edu.bh

model, plan of action, business procedures, company structures and company culture. If we are able to handle all these functions effectively, it can prompt business process enhancement (optimization) and it will produce better results for the organization. It additionally triggers industry disturbance by the presentation of new plans of action and the advancement of digitized products and services [1].

We are currently living in a period of digitization, populated by the digital natives of generations anyway according to the exploration shows just around one of every eight worldwide banks (12%) have all the earmarks of being completely dedicated to digital transformation, roughly four out of 10 banks (38%) are in the transformation stage yet have not conveyed a strong and convincing digital change strategy to the market and the excess half of banks have not made huge progressions in digital transformation, to improve their significance, they should adopt and fully committed to digital transformation strategy. As per this data only just couple of banks are contributing and investing more money towards turning out to be first digital banks [2].

Starting late, the economic business has seen rapidly creating allocation of financial advancement, or FinTech. Banks and Venture Capitalist uphold have made substantial interests in FinTech, reflecting their cravings for critical change in the finance industry. The fast development of progress in FinTech has reviewed the conceivable impact on banks and their strategies testing, while some market spectators measure that an enormous piece of bank's revenues, especially in retail banking, is in hazard all through the accompanying 10 years [3].

1.1 Situational Analysis

Singapore inhabitants are progressively receiving digital payments because of the COVID-19 pandemic, as per a Global Data report. The report gauges absolute card payments will progress by 2.5% to \$85.3 billion while money withdrawals at ATMs will decrease by 3.4% in 2020. Singapore is a profoundly evolved contactless card markets, with most shoppers approaching contactless card. The COVID-19 episode has prodded shippers to embrace electronic payments. Cash payments are set decrease in the present moment in Singapore while card payments, especially contactless payments, are required to fill in not so distant future [4].

2 Review of Literature

Review of Literature is a synopsis of connected published evidence specifically point and attention. Review of Literature is normally coordinated with a depiction, precis and basic assessment of individual source and furthermore recognizes the gaps that require further exploration. The research issues mentioned under research question

will be further described, analysed with past research data and also identifies the gaps for further research purpose.

2.1 Challenges to Traditional Banking

The term ‘Traditional Banking’ is referred to as the banking system that has been followed since past. The traditional banking system is the banking system where the financial institution provides the financial services to the customer by accepting deposits from the public and creates demand deposit. The traditional banking is none other than banking facilities that are provided by the banking sectors to their customers. The traditional banking is an institution that is devoted to the administration of the deposit that is made by its customers. This banking facility or service is running from a long period and is still continuing, the banking institution has different branches where the customers can visit in order to deposit or withdraw their money. The system of customers visiting the bank’s branch and deposit their money and withdraw their money and to avail various other banking system has been a long-followed process and it is still into practice in recent times but many people prefer to do it over mobile.

It should be noticed that the conventional financial framework is right now confronting a purge as they battle to fulfil their clients. Preferably, this purge results from the expanding number of customers that are looking for digital banks. Cocking (2018), state that claims from client puts it out that they acquire monetary control with regards to digital and neo-banks. Individuals today can possess their monetary information, and also, they can straightforwardly transfer the data to other monetary organizations. As [5, 6] clarifies, neo-banks are 100% autonomous and, furthermore, they have no physical branches or service centres. At the end of the day, neo-banking implies no work area banking, it is carefully through mobile. Eminently, the digital retail banking transformation is here and numerous other financial organizations have indicated their interest to proceed with digital transformation. This desire for change in the financial framework has brought about a touchy, just as worldwide development in the rise of more clients inclined towards neo-banks.

As per [6], these sorts of banks offer fully digital services at amazingly low expenses. This perspective has pulled in numerous individuals from the millennial age since utilities can be paid at their solace. With such a circumstance within reach, the traditional banks are currently dominated by the neo-banks on numerous fronts. Then again, research from [7] demonstrates that customers have been confronting a significant issue in regard to utility payments because of the inner cost structures inside the conventional banking framework. Nonetheless, the development of neo-banks has essentially diminished the majority of the operational costs that are forced on utilities. Groenfeldt [8] suggest that the conventional banks should significantly increase their work to keep up their income contrasted with neo-banks. Additionally, insights depict that the costs forced on utilities in conventional banks range from \$150 to \$250, though the sum for advanced or neo-banks is just \$45 [9]. Aside

from the issue of costs, neo-banks offer consistent client experience, which is the main attraction to their target group (the working and considering recent college grads). Customers are provided with individual financial management application with actionable and easy-to-use features that assist when saving money or travelling for leisure activities.

2.2 Evolution of FinTech and Latest Trends

Lately the business scene in the financial industry has changed significantly. After the monetary emergency of 2007–2008 the economic industry specialists have heightened guideline of the financial area, presenting new or fortifying present norms. Moreover, advancements in technology, innovation and Finance Technologies (FinTech) [10] have expanded the need to search for additional inventive resolutions in banking [11] and replace conventional financial frameworks with new innovation-based cycles [7, 12].

As the FinTech area fringes are hard to characterize, accessible information on FinTech is to some degree disputable, contingent upon organizations remembered for the statement. Accenture study characterizes FinTech organizations as organizations that “provide advancements for banking and corporate account, investment business sectors, financial information investigation, payments and individual finance management” [13]. In this way, the advancement of FinTech and its effect on the eventual fate of banking are very effective now a days [14].

2.3 Organizational Innovation

Innovation alludes to the execution of approximately fresh that is useful to the trend-setter. It is the selection of fresh ideas or conduct [15, 16] stated innovation as “the presentation and implementation, inside a gathering, company, or more extensive culture, procedures, products, or systems fresh to the pertinent unit of reception and expected to profit the gathering, distinct, or more extensive culture.” Rogers [17] depicted innovation as “a thought, exercise, or item that is seen as new by a specific or other unit of acceptance”. According to [18], innovation is “the multi-stage measure whereby companies change thoughts into new/enhanced products, services or methods, to progress, contend and separate themselves effectively in their domain”. Innovation identifies with new products and 72 JED 21,1 services, creation techniques and methodology and innovative technologies, just as regulatory variations [19].

This research embraces the exhaustive and broadly perceived meaning of innovation presented by the [20] “A innovation is the developing of brand new or essentially enhanced invention (good or service), or method, another promoting technique, or another organizational strategy in business best practices, work environment firm

or outside relations”. The explanation involves four sorts of innovation: product development, process advancement, organizational innovation and promoting innovation—in which organizational innovation is stated as “the execution of brand-new corporation method in the organizations strategic best experiments, working environment organization or outside connections” [20].

A Forrester Consulting exploration examination authorized by Accenture Collaborative originates that the vital drivers of high-tech change are advantage to the organization, consumer loyalty and to increase rapidity of the market. To additionally explain, the examination got some evidence about their wide essential requirements during the following year. The core verdict was “enhancing the customer journey”, trailed by emerging profits, improving variation in the products and decreasing costs [9].

2.4 Strategic Ways to Upgrade with Product Innovations

New Product advancement is one of the key exercises for some organizations to accomplish upper hand [21]. Innovativeness and development are firmly related, yet they are not something very similar. A few organizations can create a ton of thoughts yet cannot place the thoughts without hesitation. Creativity is characterized as far as inventiveness while development comprises of changing a ground-breaking thought into another brand new product, method or service, which prompts picking up benefit with respect to business endeavours [22]. Advancement can suggest high beginning and non-stop investments, hazards and vulnerability, the advantages, for example distinction from rivalry, client allegiance, cost expenses for inventive products and passage obstructions for potential imitators for the most part, appear to exceed the costs [23]. In digitally developing organizations, almost 90% of strategies concentrate around enhancing procedures in developing brand new product by using advanced technology and innovation [24].

2.5 Gap Analysis

Gap analysis alludes to an investigation action among guidelines and the real distribution dissimilarities [25]. This gap examination would comprise inside and outer investigation, where outside is about how it is verbal with the client, while interior is about how the banks developed the products or service. It is to help in shutting the gaps with the end goal for business to achieve the maximum elevated potentials. This is a comparative study method to contrast and analyse the Traditional Bank A and Digital Bank B, the findings showed that digital transformation, organizational and product innovation play a major role in customer satisfaction.

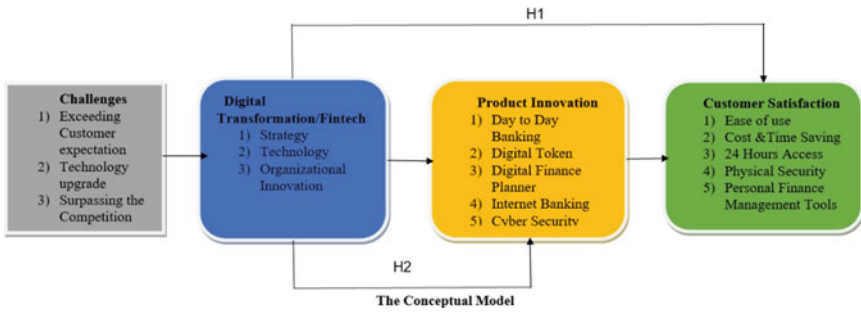


Fig. 1 Conceptual model

2.6 Conceptual Framework Development

Figure 1 shows the conceptual model, it shows what are the challenges faced by traditional banks and these are the main trigger points for digital transformation of the banks. The key drivers which will accelerate the digital transformation are strategy, technology and organizational innovation [26]. The digital transformation in banks can be achieved by product innovation, and it influences the customer satisfaction in banks. The conceptual framework in Fig. 1 also identifies the independent variables like day-to-day banking, digital tokens, digital finance planner, Internet banking and cyber security under product innovation and dependent variables like ease of access, cost saving (lower interest rates and less transaction fees), time saving (no need to go bank or ATM), 24 h access (can make payments any time), physical security (no need to go out with cash), personal finance management tools to know your past spending and future expenses are dependent variables for customer satisfaction.

3 Methodology

3.1 Research Questions

Based on the above problem statement this research study concentrates on the digital transformation of banks and FinTech evolution, in this context there is a need to discover challenges faced by traditional banks and also find the importance of product innovation which directly pertains to customer satisfaction and generates more value to the organization so above all raises the following research questions,

- What are various strategic ways for traditional banks to upgrade with product innovations to generate more value?

3.2 Aim and Objectives

With the above research questions to be addressed in the current research, the overall aim of the project is to find out the challenges faced by traditional banks, understand how technology disruption changed the banking operations, analyse how technology became a key driver for digital transformation of banks, conduct a comparative study of product innovations adopted by Bank A and Bank B from customer point of view and provide some recommendations to all the traditional banks on how we can generate more value to banks by adopting product innovation in the banks.

- To evaluate the product innovations adopted by Traditional Bank A and Digital Bank B.
- To suggest traditional banks on various strategic ways to upgrade with product innovations to generate more value.

3.3 Hypotheses

Null Hypothesis1: There is no significant relation between customer satisfaction and digital transformation of the bank.

Alternate Hypothesis1: There is a significant relation between customer satisfaction and digital transformation of the bank.

Null Hypothesis 2: There is no significant impact on day-to-day banking operations due to FinTech transformation.

Alternate Hypothesis 2: There is a significant relation between day-to-day banking operations and FinTech transformation.

3.4 Research Design

As per research philosophy realism was opted to work in this research, realism was defined by [27] that it was a philosophy that is constructed on scientific approach to develop the knowledge. This study is expected to contribute the areas of traditional banks, digital banking, organization innovation, customer satisfaction and suggestions. In the exploration approach, deductive methodology was picked to reasonable way at the examination as it was building on the sensible perspective and to reach inference from the hypothesis [27]. Deductive approach initiates from the current speculation and prototype, which proposals were made and besides thus attempted through observational examinations. Moreover, the investigation study would be a quantitative assessment with deductive technique. So as to build up the investigation method, study methodology was assumed for this investigation. It could consider diverse size of publics and from this general public to find the user experiences of

Bank A and Bank B clients. Research decision of this investigation is mono strategy as a solitary subjective information assortment method and related examination techniques was used. In any case, [28, 29] expressed that mono strategy is to be a lot simpler to design and execute.

3.5 Sample Selection and Size

Primary data were collected using convenience sampling, with the sample size of 60 users being selected to be the respondents of the survey. This research was conducted in Singapore with descriptive research design and questionnaires were used as the research instrument. The respondents were contacted online method but they will still remain as anonymous accessing the survey link which does not capture their name and email ID. The respondents are customers who have used the service of traditional or digital bank. They are general public and do not have any specific affiliations with the bank except as customer of their products; hence, the above research is only using and collecting information that is available in the public domain [30]. This research was carried out using non-probability sampling in which that everybody had the equivalent opportunity to be chosen and depended on their accessibility and helpful chance to chip away at the survey [31].

4 Findings

4.1 Research Validity and Reliability

Content validity was alluded to the subjective arrangement among the experts that a scale legitimately appeared to precisely reflect what it was supposed to quantify. Construct validity is to guarantee that the estimation was really estimating what it was expected to quantify during the statistical investigation. Content validity of the survey was finished by getting the input from 5 banking consultants, who are subject matter specialists in the financial domain. Construct validity was done by referring a statistician, to confirm that suitable variable sorts like nominal, ordinal and categorical along with intervals are preferred. Reliability analysis was determined using Cronbach's alpha coefficient, a score of at least 0.73 or above is measured as reliable [32, 33]. The reliability score for all the items in the research instrument was 0.946. Additional, reliability analysis was also shown for each segment of research instrument to comprehend the reliability of, respectively, segment.

As per the above results in Table 1, with all the scores being greater than 0.73, the research instrument is measured reliable.

Table 1 Results of reliability analysis

Questions	Cronbach's alpha	No. of items
Accessibility and customer satisfaction during COVID-19	0.750	9
Attitudes and perceptions towards digital banking	0.912	7
Innovative user interface/banking products	0.924	5
Benefits of digital banking services compared to traditional banking	0.828	6
Reliability score for all items together	0.946	27

4.2 Demographic Profile of the Respondents

Based on the data collected, Table 2 indicates that there were more male respondent's (65%) participated when compared with female respondent's (35%). This implied that more males use banking facilities than female. Coming to the age, most of the

Table 2 Gender, age, academic level and occupation

Gender	No. of respondents	Percentage (%)
Male	39	65
Female	21	35
Total	60	100
Age	No. of respondents	Percentage
18–30 years old	45	75
31–45 years old	11	18.3
46–55 years old	4	6.7
56 years old+	0	0
Total	60	100
Academic level	No. of respondents	Percentage
Secondary and below	0	0
Diploma	1	1.7
Bachelor's degree	34	56.7
Master's degree	25	41.7
Doctorate degree	0	0
Total	60	100
Occupation	No. of respondents	Percentage
Government	3	5
Private limited	50	83.3
Self-employed	3	5
Not employed	2	3.3
Student	2	3.3
Total	60	100

respondents (75%) are falls in 31–45 range, 18.3 falls in 18–30 range and remaining respondents (6.7%) are falls in 46–55 range. This indicates that younger generations are more inclined towards technology rather than conventional services. On levels of education, the majority of the respondents as presented by 56.7% were degree holders, 41.7% were master’s degree holders. Only 1.7% had a diploma. This indicates that all the educated were adopting to new technologies and they are very familiar with digital banking products and their benefits so more educated people will not use traditional banking. Coming to the occupation, the majority of the population (83.3%) are employees of private limited company. This indicates that most of the private employees use digital banking services rather than traditional banking services. For example, IT professionals are very well equipped with technology so they always prefer to use digital banking services. Out of 60 sample size, exactly 50% are traditional bank customers and 50% are digital bank customers.

4.3 Accessibility and Customer Satisfaction During COVID-19

Most of the customers Fig. 2 indicate that 88.3% respondents did not face any difficulty in opening new account or credit card or loan during COVID-19. Only 11.7% of the respondents were faced difficulty in opening new account or credit card or loan. From this finding we can say that still some banks did not fully digitalize the account opening/credit card applying/Loan processing.

Table 3 indicates that most of the respondents were rated high for mobile banking device compared to other financial devices like Internet, POS terminal, ATM machine, credit card and debit card, due to this it got the first rank. This implies that customers are more interested towards cashless transactions (digital banking) rather than traditional banking financial devices and represented in Fig. 3.

7. Do you experienced difficulty in opening a new account or credit card or loan during COVID-19?
60 responses

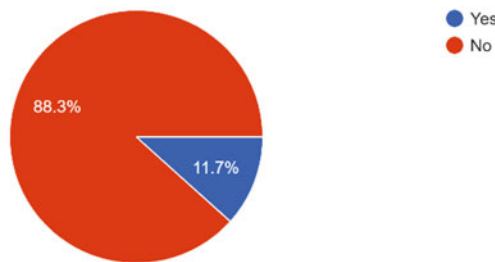


Fig. 2 Opening a new account or credit card or loan during covid-19

Table 3 Ranking of different financial devices

S. No.	Attributes	Customers assigning scores					Mean	Rank
		1	2	3	4	5		
1	Mobile banking	2	0	4	14	40	4.50	1
2	Internet	1	1	6	14	38	4.45	2
3	POS	5	2	16	17	20	3.75	6
4	ATM machine	4	6	7	14	29	3.97	5
5	Credit card	2	4	5	11	38	4.32	3
6	Debit card	5	0	9	15	31	4.12	4

Note Ranks in the last column is based on the values of mean scores

10. How would you rate your ability to use the following devices? On a scale of 1 to 5, 1 being the lowest and 5 being the highest.

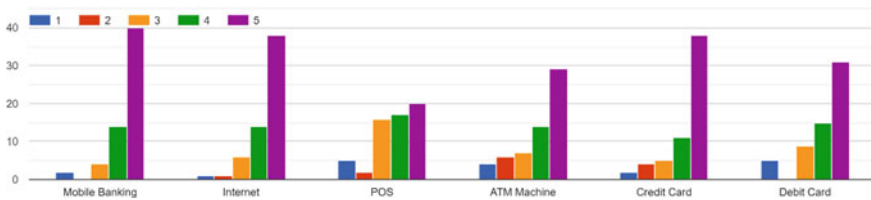


Fig. 3 Ranking of different financial devices

Figure 4 indicates that 63.3% respondents are satisfied with banking service during COVID-19, only 36.7% respondents are not satisfied with the banking services. This implies that digital banking customers are more satisfied than traditional bank customers due to product innovation adoption in their banking services.

4.4 Attitudes and Perceptions Towards Digital Banking

Figure 5 shows, 76.7% respondents confirmed that digital banking service are cheaper than traditional banking services so most of the banking customers are preferred to use digital services rather than traditional banking services due to no extra cost required to avail the digital banking services.

Table 4 indicates that most of the respondents were chosen ease of use and also rated high due to this it got the first rank compared to other attributes like, trust in banks, overall trust, trust in third-party payment network (Visa and Mastercard), trust in the technology of mobile banking and security from fraud. This implies that customers are more interested towards comfort in accessing the digital banking services rather than traditional banking services.

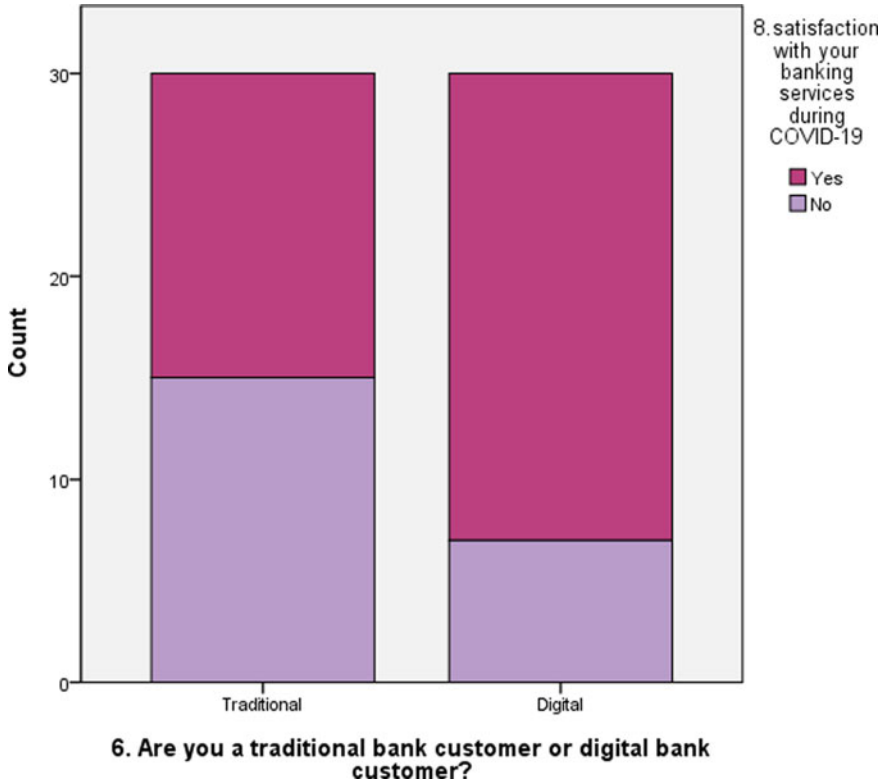


Fig. 4 Satisfaction with banking services during COVID-19

Fig. 5 Attitude and perception towards digital banking

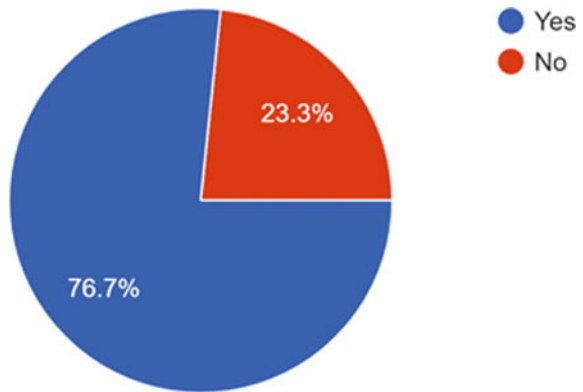


Table 4 Ranking of customer perception digital banking

S. No.	Attributes	Customers assigning scores					Mean	Rank
		1	2	3	4	5		
1	Overall trust	1	0	16	23	20	4.02	2
2	Trust in banks	3	1	13	21	22	3.97	3
3	Trust in the technology of mobile banking	1	4	14	22	19	3.90	4
4	Trust in third-party payment network (Visa and Mastercard)	1	6	13	24	16	3.80	5
5	Security from fraud	2	4	17	23	14	3.72	6
6	Ease of use	1	0	7	24	28	4.30	1

Note Ranks in the last column is based on the values of mean scores

4.5 Innovative User Interface/Banking Products

Table 5 indicates that most of the respondents were rated high for “It is easy to make a payment, transfer money and balance enquiry” compared to other digital menu

Table 5 Ranking of different digital banking menu services

S. No	Attributes	Customers assigning scores					Mean	Rank
		1	2	3	4	5		
1	It is very easy to navigate and understand about my mobile banking menu	1	1	13	20	25	4.12	2
2	It is easy to make a payment, transfer money and balance enquiry	1	1	7	17	34	4.37	1
3	Facility of digital token for easy and secure access	4	1	7	22	26	4.08	3
4	Dedicated option for credit/debit card blocking in case of any emergency	1	2	13	22	22	4.03	4
5	Past expenses and upcoming spending's can be shown in personal finance management tools	1	2	13	23	21	4.02	5

Note Ranks in the last column is based on the values of mean scores

services due to this it got the first rank. This implies that customers main reason to use digital banking is because of its availability (24 h), they can make transactions anytime and no need to go to ATM or bank physically which saves lot of time in their daily activity.

4.6 Benefits of Digital Banking Services Compared to Traditional Banking

Table 6 indicates that most of the respondents were rated high for “Time saving (no need to go to bank or ATM)” compared to other digital banking benefits due to this it got the first rank. This infer that customers do not want to waste their quality time in standing in the long queues at ATM machine and banks which saves lot of their time. Apart from this they also no need to spend any extra cost for digital banking services whereas for accessing traditional banking services, customers need to spend extra money to reach physical Bank Branch locations.

Hypothesis Testing

The digital transformation of the bank is measured by product innovation so the study was conducted based on customer satisfaction to prove the null and the alternative hypotheses.

Table 6 Ranking of different benefits of digital banking services

S. No.	Attributes	Customers assigning scores					Mean	Rank
		1	2	3	4	5		
1	Cost saving (Lower rates, transaction fees)	3	4	19	19	15	3.65	5
2	Time saving (There is no necessity to go to bank or ATM)	1	0	4	13	42	4.58	1
3	24 h services (The transaction can be made any time)	1	2	4	14	39	4.47	2
4	Physical security (There is no requirement to go out with cash)	1	0	14	18	27	4.17	4
5	Very rarely visit my branch for services, thanks to digital banking, POS terminals and Internet banking	2	2	8	15	33	4.25	3

Note Ranks in the last column is based on the values of mean scores

Table 7 Chi square test results of digital transformation of the bank and customer’s satisfaction

Tests	Value	df	Asymp. Sig. (2-sided)	Exact sig. (2-sided)	Exact sig. (1-sided)
Pearson chi-square	4.593**	1	0.032		
Continuity correction	3.517*	1	0.061		
Likelihood ratio	4.674	1	0.031		
Fisher’s exact test				0.060	0.030
Linear-by-linear association	4.517	1	0.034		
N of valid cases	60				

** denotes significance at a 1% level.

* denotes significance at a 5% level

H0: $\mu_1 = \mu_2$ $\mu_1 - \mu_2 = 0$: Digital transformation of bank does not have significant effect on customer satisfaction.

H1: $\mu_1 \neq \mu_2$ $\mu_1 - \mu_2 \neq 0$: Customer satisfaction has significant impact on digital transformation of the bank.

Table 7 indicates the chi-square test performed that shows a significant relationship between digital transformation of the bank and customer’s satisfaction, $\chi^2(1, N = 60) = 4.593$. The findings indicate that the $P < 0.05$; thus, the null hypotheses were rejected and the alternative hypotheses were considered which stated that product innovation has an influence on customer satisfaction, which in turn says digital transformation of the bank depends upon product innovation.

H0: $\mu_1 = \mu_2$ $\mu_1 - \mu_2 = 0$: FinTech Transformation does not have significant effect on day-to-day banking.

H2: $\mu_1 \neq \mu_2$ $\mu_1 - \mu_2 \neq 0$: Day-to-day banking has significant impact on the FinTech transformation.

Table 8 indicates the chi-square test performed that shows a significant relationship between FinTech transformation and day-to-day banking operation, $\chi^2(1, N = 60) = 17.385$. The findings indicate that the $P < 0.05$; thus, the null hypotheses were rejected

Table 8 Chi square test results of fintech transformation and day to day banking operation

Tests	Value	df	Asymp. sig. (2-sided)
Pearson chi-square	17.385*	4	0.002
Likelihood ratio	19.367	4	0.001
Linear-by-linear association	6.856	1	0.009
N of valid cases	60		

* denotes significance at a 5% level

and the alternative hypotheses were considered which stated that FinTech Transformation has an influence day-to-day banking operations such as mobile banking, Internet banking, POS terminal, credit card/debit card and ATM.

As per the findings from data collection and hypothesis testing, it is proved that digital customers are more satisfied when compared with traditional bank customers during COVID-19 and also digital transformation is directly correlates to customer satisfaction.

5 Discussion

Based on the data analysis it was found that the customers are more satisfied with digital banking services and the changing tendencies for customers, who are requesting better approaches to utilize financial services, and the competitive scene onto which the enormous technology and the FinTech organizations have blasted, are constraining the banks to face digitalization as an issue of persistence, so in order to meet the competition with in the financial services, all banks should adopt the strategy of digital transformation. Based on this research we can conclude that the principle three key drivers for digitalization of a bank are improvement in new inventive products to accomplish consumer loyalty, while the subsequent methods adopt to use upcoming technologies available in the market and the last requires innovation across the organization that will accelerate the digital transformation. Those banks which are already following the above process are present in advanced position now and ready to satisfy all the new demands of the customers and they are already in competition with advanced digital financial service organizations. The study was embedded general suggestion as follows,

- **Digital Banking Products** Artificial intelligence (AI) helped deals of banking products, for example stores, credits and home loans, are led through direct channels, including social media. That is in accordance with moving shopper inclinations and conduct patterns in online business, particularly coordinated at Generation Y and techsavvy clients.
- **Digital Robo-advisory** AI-based digital financial organizer oversees month-to-month pay, repeating payments, reserve funds and speculations, expanding connection between the advanced bank and clients. Legitimate continuation of the hover of trust between the advanced bank and clients, where clients depend on the Digital Robo-advisory services to improve speculation portfolios dependent on individual objectives and inclinations, routinely changes them and records steady outcomes and appropriately dispenses assets for each period of the client's journey towards everything computerized.
- **Internet of Things** Mobile or Fitbit can be used to track our fitness and the same devise can be used as digital wallet to pay bills while purchasing any product or service.

- **Blockchain-enabled** lending is a technology which can be used to speed up the loan processing when compared with legacy technologies. It will automatically verify all the KYC/financial documents in real time and process the loans very quickly.
- **Cloud Architecture** Technology made some amazing progress from data centres to cloud computing technologies like Amazon Web Service and Microsoft Azure. Nowadays banking regulators gave permission to banks to maintain the data in cloud which will save lot of operational cost and that directly leads to additional income generations for banks.

Digital transformation is about something beyond giving on the web and mobile usefulness. Conventional financial suppliers need to join advanced rapidity and accommodation with manual influences that are equally smart and kind at vital instants in the client satisfaction.

6 Conclusion and Future Research

Similar to any other research, there are limitations for this study as well. The major limitations of this study are time and financial constraints. Considering the current pandemic situation, confined time and lack of money were the significant restrictions in visiting and conducting face-to-face interviews or to direct telephonic interviews. Hence, it was chosen to plan and email a questionnaire to selected people to finish the survey. The research scope is very limited and comparative analysis is not based on the financial performance of the two banks and also noted that this research only concentrates on the Singapore bank customers.

This study should be simulated in all the traditional banks and also can stretch out to different countries as well, for instance developing country like India. India is one of the upcoming countries where mobile payments and digital banking are valuable because of the enormous populace living in rural areas. Further study is expected to comprehend whether there are different components that influence bank's client experience and financial performance in those unique circumstances. Stretching out the study to explicit banks and bank employee's perceptions also can be taken and the results can be compared with those from customer's insights. Covering all these extra lines of research will assist with growing more vigorous digital bank transformation theory in near future.

References

1. Kiron D, Kane GC, Palmer D, Phillips AN, Buckley N (2016) Aligning the organization for its digital future. MIT Sloan Manag Rev 58(1)
2. Accenture (2019) Only half of banks globally are making significant advancements in digital transformation, Resulting in lower market valuations, accenture report finds. <https://newsroom.accenture.com/news/only-half-of-banks-globally-are-making-significant-advancements-in-digital-transformation-resulting-in-lower-market-valuations-accenture-report-finds.htm>. Accessed 14 Nov 2020

3. McKinsey&Co (2015) Global banking annual review. Global Banking Annual Review. https://www.mckinsey.com/~media/McKinsey/Industries/Financial+Services/Our+Insights/The+Fight+for+the+Customer+McKinsey+Global+Banking+Annual+Review+2015/McKinsey_Global+Banking+Annual+Review_2015_d.pdf. Accessed 14 Nov 2020
4. Data G (2020) COVID-19 accelerates shift to digital payments in Singapore. Global Data. <https://www.globaldata.com/covid-19-accelerates-shift-to-digital-payments-in-singapore-says-globaldata/>. Accessed 14 Nov 2020
5. Christensen C (2011) Clay Christensen's milkshake marketing, vol 3, p 2019. Retrieved July 2011
6. Derwin J (2020) What is a neobank—and what are they offering in Australia? Business Insider. <https://www.businessinsider.com.au/what-is-a-neobank-australia-2020-2>. Accessed 14 Nov 2020
7. Carr NG (2003) Harvard business review: it doesn't matter
8. Groenfeldt T (2019) Bank 4.0 will be all-digital, low-overhead, mobile-first. Forbes. <https://www.forbes.com/sites/tomgroenfeldt/2019/04/19/bank-4-0-will-be-all-digital-low-overhead-mobile-first/#1ba69a8a122a>. Accessed 14 Nov 2020
9. Edmead M (2016) Digital transformation: why it's important to your organization. IDG Contrib Netw
10. Report F (2015) FinTech report. Investment trends in FinTech. <http://www.svb.com/News/Company-News/2015-FinTech-Report-Investment-Trends-in-FinTech/>. Accessed 14 Nov 2020
11. Arner DW, Barberis J, Buckley RP (2015) The evolution of Fintech: a new post-crisis paradigm. *Geo J Int'l L* 47:1271
12. Hochstein M (2015) Fintech (the word, that is) evolves. *The American Banker*. Erişim tarihi 1:2019
13. Skan J, Lumb R, Masood S, Conway SK (2014) The boom in global Fintech investment. Accent
14. Dapp T, Slomka L, AG DB, Hoffmann R (2014) Fintech—The digital (r) evolution in the financial sector. *Dtsch Bank Res* 11:1–39
15. Wu S-I, Lin C-L (2011) The influence of innovation strategy and organizational innovation on innovation quality and performance. *Int J Organ Innov* 3(4)
16. West MA, Anderson NR (1966) Innovation in top management teams. *J Appl Psychol* 81(6):680
17. Rogers EM (2003) Diffusion of innovations, 5th edn London. UK Free Press Scholar
18. Baregheh A, Rowley J, Sambrook S (2009) Towards a multidisciplinary definition of innovation. *Manag Decis*
19. Fay D, Shipton H, West MA, Patterson M (2015) Teamwork and organizational innovation: the moderating role of the HRM context. *Creat Innov Manag* 24(2):261–277
20. Data II (2005) Oslo manual
21. Koufteros X, Vonderembse M, Jayaram J (2005) Internal and external integration for product development: the contingency effects of uncertainty, equivocality, and platform strategy. *Decis Sci* 36(1):97–133
22. Satsomboon W, Pruetipibultham O (2014) Creating an organizational culture of innovation: case studies of Japanese multinational companies in Thailand. *Hum Resour Dev Int* 17(1):110–120
23. Rosenbusch N, Brinckmann J, Bausch A (2011) Is innovation always beneficial? A meta-analysis of the relationship between innovation and performance in SMEs. *J Bus Ventur* 26(4):441–457
24. Kane GC, Palmer D, Phillips AN, Kiron D, Buckley N (2015) Strategy, not technology, drives digital transformation. *MIT Sloan Manag Rev Deloitte Univ Press* 14(1–25)
25. Upadhyaya M (2013) Customer satisfaction measurement: an empirical study of the need-gap analysis in the service industry. *J Econ Bus Res* 19(2):54–61
26. King B (2018) Bank 4.0: banking everywhere, never at a bank. Wiley & Sons
27. Saunders M, Lewis P, Thornhill A (2009) Research methods for business students. Pearson education
28. Kothari SP, Li X, Short JE (2009) The effect of disclosures by management, analysts, and business press on cost of capital, return volatility, and analyst forecasts: a study using content analysis. *Account Rev* 84(5):1639–1670

29. Maxwell Ja (2012) *Qualitative research design: an interactive approach*. Sage publications
30. Wood MJ, Ross-Kerr J (2010) *Basic steps in planning nursing research: from question to proposal: from question to proposal*. Jones & Bartlett Publishers
31. Bryman A (2016) *Social research methods*, 4th edn. Oxford university Press
32. Bryman A, Beardsworth A (2006) Is qualitative research becoming McDonaldized? *Methodol Innov Online* 1(1):3–13
33. Becker S, Bryman A, Ferguson H (2012) *Understanding research for social policy and social work 2E: themes, methods and approaches*. Policy Press

A Novel Approach to Improve the Performance of a Classifier Using Visual and Haptic Data



Sekhar R. Aravind and K. G. Sreeni

Abstract In haptic identification of objects, various factors like friction and acceleration between the object and the haptic tool, audio, and image information contribute a lot. This paper proposes a novel technique to improve the performance of a classifier using visual and haptic data. Both conventional machine learning and deep neural network have been used for the experiment. The dataset used here is the haptic data with image, friction, and acceleration as inputs. Individual predictions models are developed for each input modality and then combined using the weighted average technique. The results showed a considerable improvement when compared to conventional multimodal feature-based classifiers. Simulation and analysis are done using Python3.7/Jupyter Notebook.

Keywords Multimodal · Haptics · Weighted average

1 Introduction

Haptics is concerned with the science of human touch and the interaction that happens in a virtual environment, and the purpose is carried out with the help of special devices known as haptic devices. The word haptics began to emerge in the early 1990s s [1–3]. To understand the sense of touch, the focus should be not only on the perception of the physical boundaries of the body but also on the analysis of surface properties. To enable the user to sense haptic vibrations, the haptic rendering algorithm should have an update rate not less than 1 KHz [3–5]. If the above-mentioned criteria are not satisfied, then the result will be unstable haptic vibrations, particularly in the case of stiffer contacts.

Tactile understanding plays a significant role in haptic identification besides visual inputs [6–9]. Humans usually identify objects using both haptic and visual feedback. In recent years, multiple input-based haptic identifications have been gaining

S. R. Aravind (✉) · K. G. Sreeni
Department of Electronics and Communication, College of Engineering Trivandrum,
Trivandrum, India
e-mail: aravindskhr.87@gmail.com

significance. Haptic identification of objects can be done using various features like image, acceleration, and friction between the object and tool, audio information, etc. [10–13]. Multimodal techniques have been recently used in haptics for classification purposes [14, 15]. Multiple input-based classification techniques have been found to improve the accuracy much better when compared to single feature-based classification [16, 17]. Multimodal feature-based surface classification techniques have been found to improve the accuracy to a great extent. In most cases, the multimodal inputs are applied to a single network which is further processed to get the final prediction. One main factor that needs to be taken care of in the multimodal feature-based classification is the network complexity as a result of feeding multiple inputs to a single network.

Both deep learning and machine learning techniques are used here to effectively classify the objects based on visual and haptic inputs [18–20]. Such deep neural networks have been found very much useful in various areas such as object identification [21] and automatic speech recognition [22]. Multimodal techniques using deep learning have been recently used in haptics for object identification [23, 24].

Adding ensemble-based classifiers in multimodal classification helps to reduce the network complexity without compromising accuracy [25–27]. Such concepts will be very much useful if applied in the area of haptic identification [28, 29]. Another way of combining multiple classifiers is by using the confusion matrix [30, 31]. The confusion matrix describes the performance of a classifier on a set of test data for which the true values are known. The concept of maximum entropy is also used to develop the ensemble classifiers in some areas [32].

This paper introduces an ensemble-based multimodal approach to haptically identify various objects. The method can also be effectively used to improve the performance of multimodal classifiers. The proposed approach was first applied to develop a multimodal-based deep learning model. Fifteen different materials have been used for the purpose, and the datasets of the objects are obtained from <http://www.lmt.ei.tum.de/texture/>. For each material image, acceleration and friction properties are considered for classification purposes. Also, the proposed method was applied to improve the performance of multimodal classifiers. Both machine learning and deep learning-based classifiers are used for this purpose.

The remaining of the paper is organized as follows. Section 2 presents some of the relevant works in haptic identification using multiple inputs. Section 3 describes the datasets and the classifiers used followed by Sect. 4 which explains the proposed technique. The results and conclusions are given in Sects. 5 and 6, respectively.

2 Related Works

In classifier construction, an object can have multiple attributes, each of which explains a specific feature of the object. Classification methods used for such multi-feature models are termed multimodal classification methods. Multiple input-based techniques are used in classification for improving accuracy in many areas.

In many cases, visual information may not be able to predict the objects correctly since visually identical objects can differ in their internal state, material, and compliance [33]. Deep learning models have been used in the field of haptics to achieve high performance in classification tasks, particularly in multimodal applications [19, 20, 23, 24]. Multiple input-based classifications have been effectively used to overcome such limitations. In [34] Nakamura et al. proposed a method for object recognition from multiple modalities such as visual, auditory, and haptic information. Gao et al. [23] proposed a deep learning technique for tactile understanding using visual and haptic inputs. Here, haptic and visual input networks have been individually trained and the activation function from the trained networks has been used to train a multimodal network. Their results show combining the inputs from multiple modalities will result in improved performance.

Li et al. used an ensembled generative adversarial networks for learning cross-modal visual-tactile information to convert texture attributes of the image to the corresponding tactile output. They proposed that the use of an ensembled generative adversarial network made the training simple and also produce stable results. In [24], Zheng et al. developed a technique for surface material classification using haptic and visual information and achieved an accuracy of 74% from a set of 69 different materials using a naive Bayes classifier. They have selected image, acceleration, friction, and sound as the multimodal inputs for classification. In [35], Zheng et al. extended their work by applying deep learning techniques. They have used a fully convolutional neural network with the max voting framework and achieved an accuracy of 98.8%.

Multiple input-based classifications of objects are an emerging area in the field of haptics and require further research. The proposed method applies a weighted average approach that can improve the performance of multimodal classifiers.

3 Dataset and the Classifiers Used

For the experiment with multimodal data, fifteen different materials are selected which are publically available to download from <https://zeus.lmt.ei.tum.de/downloads/texture>. The images of the selected materials are shown in Fig. 1. For each material image, acceleration and friction features are used for classification purposes. Three separate classifiers are used for the experiment. A naive Bayes classifier (NBC) and classification tree (CT) are used as the conventional machine learning classifiers, while for deep learning a deep convolutional neural network has been used.

4 Proposed Method

In the proposed method the image, acceleration, and friction attributes are used for classification. Figure 2 shows the structure of the proposed method. Here, the visual data is the image and the haptic data represents the acceleration and friction. Once

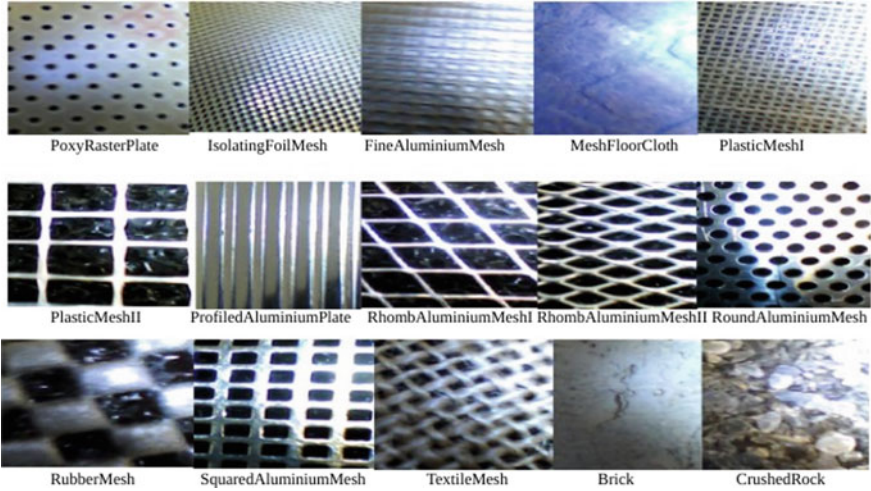


Fig. 1 Sample of materials used for the classification, data courtesy <https://zeus.lmt.ei.tum.de/downloads/texture/>

the individual models are trained, the proposed method aims to combine the models using an ensemble classification approach which is applied to the model outputs.

The proposed method combines the output of the three models using an ensemble weighted average technique. Here, three different classifier models have been developed using the image, acceleration, and friction features. In the proposed ensemble approach, weights are assigned to the individual feature-based classifiers to make a better prediction compared to the individual classifiers. The prediction accuracy of each classifier is chosen as its weight. Now, using the prediction accuracy as weights a weighted average ensemble-based multimodal classifier has been developed.

Weighted average of a non-empty finite multiset of data x_1, x_2, \dots, x_n with corresponding non-negative weights w_1, w_2, \dots, w_n is given by

$$x = \frac{\sum_{i=1}^n x_i w_i}{\sum_{i=1}^n w_i} \quad (1)$$

where $\sum_{i=1}^n w_i = 1$. The data elements with a higher weight can contribute more to the weighted mean than others with lower weights.

To illustrate the methodology used, let us consider a four-class classification problem. The classes are denoted by 0, 1, 2, 3, etc., and the vector $\mathbf{p} = [p^0, p^1, p^2, p^3]$ denotes the prediction probability vector of classifier for a given test input. Here, p^0, p^1, p^2 and p^3 denote the prediction probabilities of class 0, 1, 2, and 3, respectively. We have considered multimodal inputs such as image, acceleration, and friction data, and a classifier is built individually for each modality. The final prediction of each test input is based on the individual prediction.

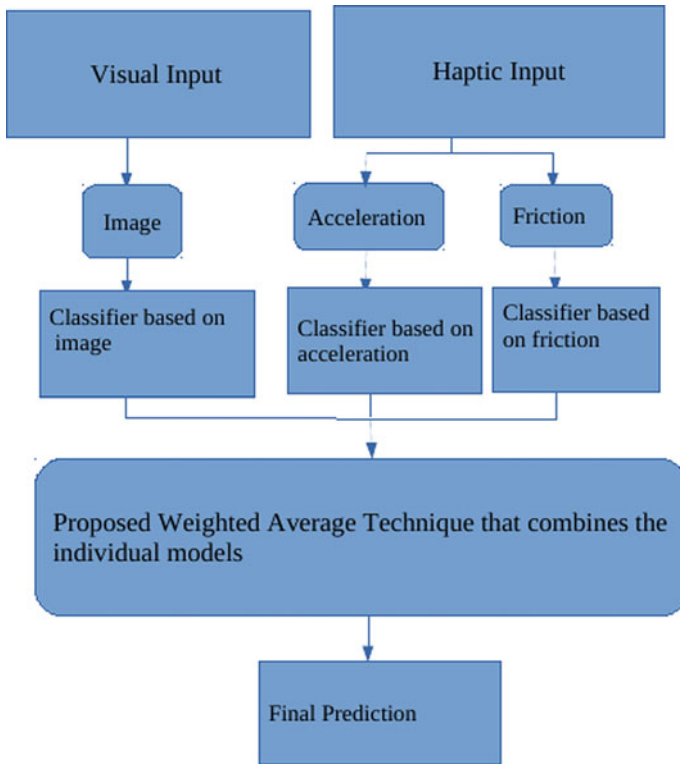


Fig. 2 Block diagram showing the structure of the proposed weighted average prediction

Let α_I denote the prediction accuracy of classifier 1 with image as input and $\mathbf{p}_I = [p_I^0, p_I^1, p_I^2, p_I^3]$ denote the corresponding prediction probability vector.

Similarly, with classifier 2 with friction data as input, α_F and $\mathbf{p}_F = [p_F^0, p_F^1, p_F^2, p_F^3]$ denote prediction accuracy and prediction probability vector.

Let α_A denote the prediction accuracy of classifier 3 with acceleration data as input and $\mathbf{p}_A = [p_A^0, p_A^1, p_A^2, p_A^3]$ denote the corresponding prediction probability vector.

Now, for the purpose of combining the three classifiers, weights have to be assigned. For the purpose of determining the proposed prediction matrix, a mathematical model has been developed using the weighted average method. Figure 3 shows the proposed mathematical model.

Here, in the proposed weighted average classifier, the weights assigned to the individual classifiers are selected in such a manner that it is equal to their overall prediction accuracy. The determination of the overall prediction probability with the multimodal input is given in Eq. 3.

$$\mathbf{p} = \frac{\alpha_I \cdot \mathbf{p}_I + \alpha_F \cdot \mathbf{p}_F + \alpha_A \cdot \mathbf{p}_A}{\alpha_I + \alpha_F + \alpha_A} \tag{2}$$

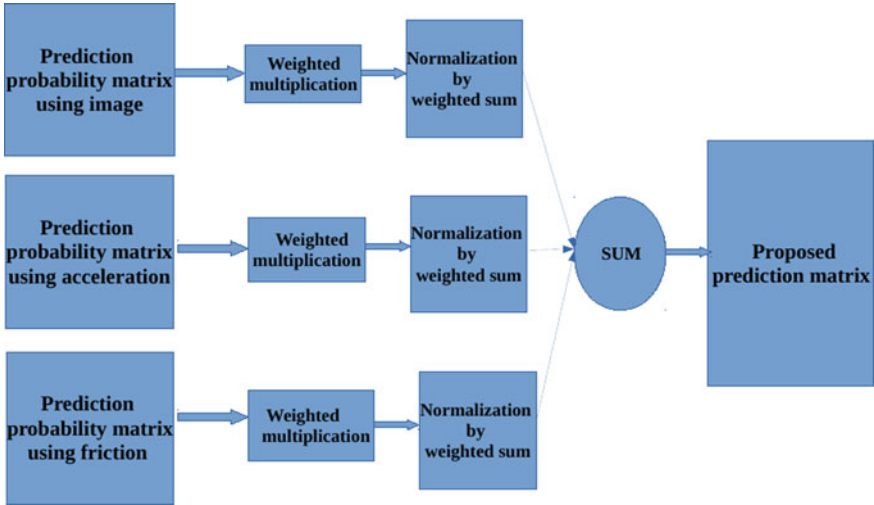


Fig. 3 Mathematical model of the proposed classifier

Here, the prediction probability vector for the j th test sample is $\mathbf{p}_j = [p_j^0, p_j^1, p_j^2, p_j^3]$ where $p_j^0 = \alpha_I \cdot p_{Ij}^0 + \alpha_F \cdot p_{Fj}^0 + \alpha_A \cdot p_{Aj}^0$.

If we have N classes based on M inputs, then the prediction probability \mathbf{p} is given by

$$\mathbf{p} = \frac{\sum_{k=1}^M \alpha_k \cdot \mathbf{p}_k}{\sum_{k=1}^M \alpha_k} \tag{3}$$

each \mathbf{p}_k is an N dimensional vector given by $\mathbf{p}_k = [p^0, p^1, \dots, p^{N-1}]$

In the proposed method, there are 15 classes ($N = 15$), and hence, the prediction probability vector is a 15-dimensional vector. The prediction probability vector for the j th test sample with the k th input is $\mathbf{p}_{kj} = [p_{kj}^0, p_{kj}^1, \dots, p_{kj}^{14}]$. The overall prediction probability vector is determined for fifteen different classes with three inputs as per Eq. 4, where $N = 15$ and $M = 3$. The overall prediction probability vector will be as shown in Eq. 5.

$$\mathbf{p} = \frac{\alpha_I \cdot [p_I^0, p_I^1, \dots, p_I^{14}] + \alpha_A \cdot [p_A^0, p_A^1, \dots, p_A^{14}] + \alpha_F \cdot [p_F^0, p_F^1, \dots, p_F^{14}]}{\alpha_I + \alpha_A + \alpha_F} \tag{4}$$

5 Results and Discussions

The simulation has been done using Python3.7/Jupyter Notebook. Here, the proposed method is first used to develop a deep learning-based multimodal classifier and also extended to improve the performance of classifiers.

5.1 Deep Learning-Based Multimodal Classifier Using the Proposed Approach

Figures 4, 5 and 6 depict the train and validation accuracy using the image, acceleration, and friction data separately. In all the cases, the training accuracy closely matches with the validation accuracy indicating that the models perform well without much overfitting.

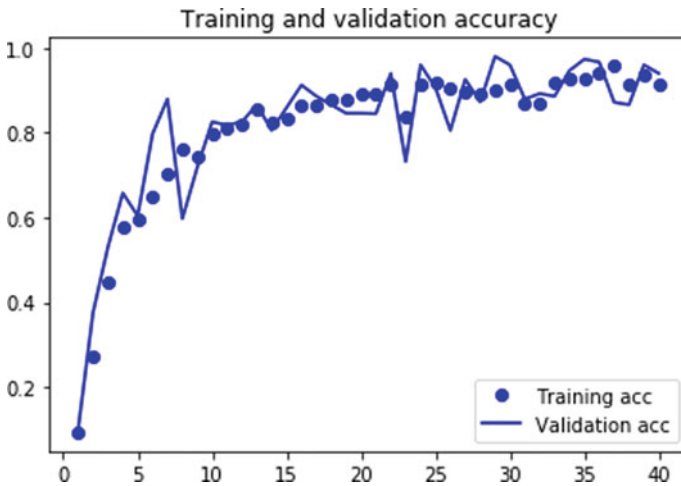


Fig. 4 Train and validation accuracy (image data)

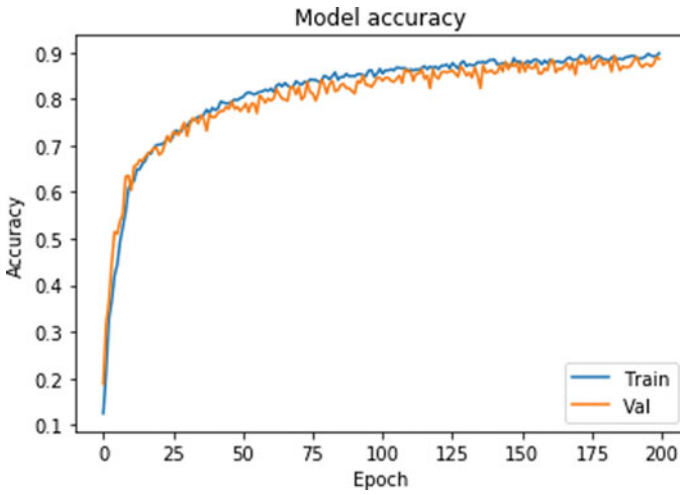


Fig. 5 Train and validation accuracy (acceleration data)

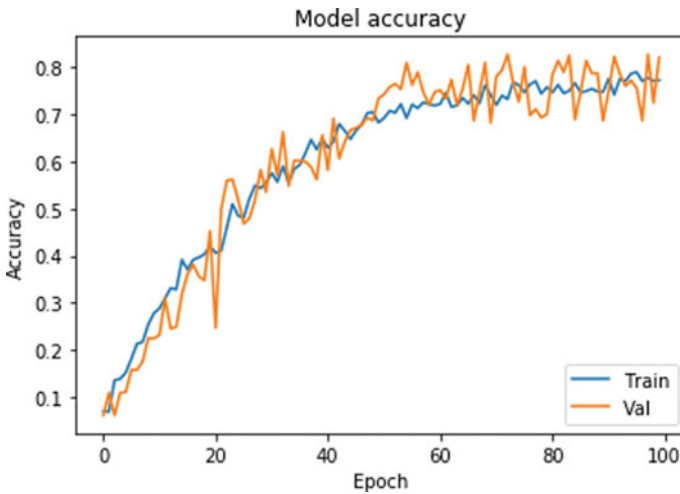


Fig. 6 Train and validation accuracy (friction data)

Let the confusion matrix of the classifiers using image, acceleration, friction, and the proposed method be denoted respectively by CM_{p_i} , CM_{p_A} , CM_{p_f} and CM_p . Equation 5–8 shows the generated confusion matrices of the classifiers for fifteen different classes with ten test inputs from each class.

Table 1 Comparison of confusion matrix of the individual feature-based classifiers and the the proposed classifier

Rank	Confusion matrix	Class score	Overall score
1	CM _p	44	5
2	CM _{p_I}	42.20	5
3	CM _{p_A}	42.10	5
4	CM _{p_F}	42.05	5

$$CM_p = \begin{bmatrix}
 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 9 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 9 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 & 0
 \end{bmatrix} \tag{8}$$

Comparison between the confusion matrices is provided in Table 1. It shows that the confusion matrix of the proposed method is better when compared to the confusion matrix generated using individual feature-based classifiers.

5.2 Performance Improvement of Classifiers Using the Proposed Approach

The proposed approach can be extended to improve the performance of conventional classifiers. Table 2 shows the performance of the conventional classifiers before and after applying the proposed method. Here, naive Bayes and classification tree are used as the machine learning classifiers and a deep convolutional network for the deep learning model. From the table, it can be seen that the conventional classifiers when implemented using the proposed approach perform better when compared to the same classifier developed using the conventional method.

The advantage of the proposed method lies in the fact that it can be used as a generalized approach to improving the performance of multimodal classifiers.

Table 2 Comparison of the prediction accuracy of the multimodal classifier using the conventional approach and the proposed technique

Classifier	Features used	Prediction accuracy (%)
Deep convolutional network	Acceleration, image and friction	94.2
Naive Bayes	Acceleration, image and friction	73.2
Classification Tree	Acceleration, image and friction	64.1
Deep convolutional network using proposed method	Acceleration, image and friction	98.7
Naive Bayes using proposed method	Acceleration, image and friction	80.1
Classification Tree using proposed method	Acceleration, image and friction	72.5

6 Conclusions and Future Work

A novel multimodal feature-based classification for haptic identification is presented in this paper. The proposed method uses a weighted average prediction technique to make the final prediction. The simulation results showed that the proposed method performs much better when compared to the predictions using individual data. The main advantage of the proposed method is that it can effectively be used to improve the performance of classifiers without any computational requirements. Also, there is not many tradeoffs between computational complexity and accuracy. In future work, the method can be improved by using a larger dataset that fine-tunes the entire multimodal network.

References

1. Saddik E et al (2011) Haptics technologies. Springer Series on Touch and Haptic Systems, Springer Berlin Heidelberg
2. Tian H, Zhang X, Wang C, Pan J, Manocha D (2016) Efficient global penetration depth computation for articulated models. *Comput-Aided Des* 70:116–125. Elsevier
3. Salisbury K, Conti F, Barbagli F (2004) Haptic rendering: introductory concepts. IEEE Computer Society, 0272-1716
4. Li Y, Zhang Y, Ye X, Zhang S (2014) Haptic rendering method based on generalized penetration depth computation. *Signal Process*. Elsevier
5. Okamoto S, Nagano H, Yamada Y (2013) Psychophysical dimensions of tactile perception of textures. *IEEE Trans Haptics* 6(1):81–93
6. Dutu LC, Mauris G, Bolon P, Dabic S, Tissot JM (2015) A fuzzy rule-based model of vibrotactile perception via an automobile haptic screen. *IEEE Trans Instrum Measure* 2015
7. Shimoe H, Matsumura K, Noma H, Sohgwawa M, Okuyama M (2014) Development of artificial haptic model for human tactile sense using machine learning. In: IEEE conference sensors. Glasgow UK
8. Yang C, Huang K, Cheng H, Li Y, Su C-Y, Haptic identification by ELM-controlled uncertain manipulator. *IEEE Trans Syst Man Cybern*

9. Culbertson H, Delgado JJ, Kuchenbecker KJ (2014) One hundred data-driven haptic texture models and open-source methods for rendering on 3D objects. In: IEEE Haptics Symposium. Houston, USA, 23–26 Feb 2014
10. Pan J, Zhang X, Manocha D (2013) Efficient penetration depth approximation using active learning. *ACM Trans Graph (TOG)* 32(6). In: Proceedings of ACM SIGGRAPH Asia 2013
11. Sreeni KG, Chaudhuri S (2012) Haptic rendering of dense 3D point cloud data. IEEE haptics symposium. Vancouver, BC, Canada, 4–7 Mar 2012
12. Zhang L, Kim YJ, Varadhan G, Manocha D (2007) Generalized penetration depth computation. *Comput Aided Des* 39(8):625–38
13. Fishel JA, Loeb GE (2012) Bayesian exploration for intelligent identification of textures. *Front Neurobotics* 6(4):1–20
14. Ruspini DC, Kolarov K, Khatib O (1997) The haptic display of complex graphical environments. In: Proceedings of ACM Siggraph, pp 345–352. ACM Press
15. Strese M, Schuwerk C, Iepure A, Steinbach E (2017) Multimodal feature-based surface material classification. *IEEE Trans Haptics* 10
16. Elavarasan N, Mani K (2017) An enhanced multi attribute depthness similarity estimation technique to improve classification accuracy. In: 2017 World congress on computing and communication technologies (WCCCT), 2–4 Feb 2017
17. Khan AA, Xydeas C, Ahmed H (2013) Probabilistic multimodal classification with dynamic feature selection. In: 16th international conference on information fusion, 9–12 July 2013
18. Qiu J, Wu Q, Ding G, Xu Y, Feng S (2016) A survey of machine learning for big data processing. *EURASIP J Adv Signal Process* 2016(1):67
19. Li Xinwu, Liu Huaping, Zhou Junfeng, Sun FuChun (2012) Learning cross-modal visual-tactile representation using ensembled generative adversarial networks. *Signal Process Mag IEEE* 29(6):82–97
20. Abderrahmane Z, Ganesh G, Crosnier A, Cherubini A (2020) A deep learning framework for tactile recognition of known as well as novel objects. *IEEE Trans Ind Inf*
21. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105
22. Hinton G, Deng L, Yu D, Dahl GE, Mohamed A, Jaitly N, Senior A, Vanhoucke V, Nguyen P, Sainath TN et al (2012) Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups. *Signal Process Mag IEEE* 29(6):82–97
23. Gao Y, Hendricks LA, Kuchenbecker KJ, Darrell T (2016) Deep learning for tactile understanding from visual and haptic data. In: 2016 IEEE international conference on robotics and automation (ICRA)
24. Zheng H, Fang L, Ji M, Strese M, Ozer Y, Steinbach E (2015) Deep learning for surface material classification using haptic and visual information. arXiv preprint [arXiv:1512.06658](https://arxiv.org/abs/1512.06658)
25. Yang B, Chen Y, Shao QM, Yu R, Li WB, Guo GQ, Jiang JQ, Pan L (2019) Schizophrenia classification using fMRI data based on a multiple feature image capsule network ensemble. *IEEE Early Access Article, IEEE*
26. Huddar MG, Sannakki SS, Rajpurohit VS (2018) An ensemble approach to utterance level multimodal sentiment analysis. In: International conference on computational techniques, electronics and mechanical systems (CTEMS)
27. Nakamura T, Nagai T (2018) Ensemble-of-concept models for unsupervised formation of multiple categories *IEEE Trans Cogn Dev Syst* 10
28. Zheng H, Fang L, Ji M, Strese M, Ozer Y, Steinbach E (2016) Deep learning for surface material classification using haptic and visual information. *IEEE Trans Multimedia* 18(12). Journal Article, Publisher, IEEE
29. Gao Y, Hendricks LA, Kuchenbecker KJ, Darrell T (2016) Deep learning for tactile understanding from visual and haptic data. In: 2016 IEEE international conference on robotics and automation (ICRA)
30. Wang X-N, Wei J-M, Jin H, Yu G, Zhang H-W (2013) Probabilistic confusion entropy for evaluating classifiers. *Entropy*. Open Access MDPI

31. Ariyanti W, Hussain T, Wang JC, Wang CT, Fang SH, Tsao Y (2021) Ensemble and multimodal learning for pathological voice classification. *IEEE Sens Lett.* IEE
32. Hong J, Mattmann CA, Ramirez P (2017) Ensemble maximum entropy classification and linear regression for author age prediction. In: *IEEE International conference on information reuse and integration (IRI)*
33. Sinapov J, Schenck C, Staley K, Sukhoy V, Stoytchev A (2014) Grounding semantic categories in behavioral interactions: Experiments with 100 objects. *Robot Auton Syst* 62:632–645
34. Nakamura T, Araki T, Nagai T, Iwahashi N (2011) Grounding of word meanings in latent dirichlet allocation-based multimodal concepts. *Adv Robot* 25:2189–2206
35. Zheng H, Fang L, Ji M, Strese M, Ozer Y, Steinbach E (2016) Deep learning for surface material classification using haptic and visual information. *IEEE Trans Multimedia* 18(12). Journal Article, IEEE

KGAN: A Generative Adversarial Network Augmented Convolution Neural Network Model for Recognizing Kannada Language Digits



H. S. Shrisha, V. Anupama, D. Suresha, and N. Jagadisha

Abstract Kannada is a south Indian language with a history of two thousand years and spoken by more than sixty million people. Kannada language has its own script for alphabets and digit representations. So there is a need for convolution neural network (CNN) model to recognize Kannada language scripts. This paper presents a design of a CNN model to recognize Kannada digits. One of the challenges faced while designing a CNN model is data over fitting. Data over fitting is a phenomenon where the trained model arrives at parameter values such that they can classify only the instances provided during training resulting in reduction of accuracy for a new unseen test instance. To overcome this problem, datasets are split into train and test sets. The detriment of this system is lesser number of instances to train the CNN. Increasing the number of training instances is a good approach, but the complexity in data collection is to be answered. In this paper, we explore generative adversarial network (GAN) as an additional data generator and its suitability. Results of analysis on the experiment revealed the following advantages; first, the data augmentation has a positive impact on CNN, next, GAN-generated data meets qualitative requirement as train and test dataset and last, epoch value for training CNN has influence on data under fitting and data over fitting phenomenon.

Keywords Kannada language · Generative adversarial network · Convolution neural network · MNIST dataset

H. S. Shrisha (✉) · V. Anupama · N. Jagadisha
Canara Engineering College, Visveswaraya Technological University,
Benjanapadavu, Bantwal 574219, India
e-mail: shree.1259@gmail.com
URL: <https://www.canaraengineering.in/>

D. Suresha
AJ Institute of Engineering and Technology, Visveswaraya Technological University,
Mangaluru 575006, India

1 Introduction

Kannada is a native language of south India, and efforts are being made to preserve and modernize the areas of application of this language. A step toward this goal is publication of Kannada-MNIST [10] dataset for Kannada digits. The said dataset is a collection of sixty thousand images. The dataset symbols provide variations in representation of the kannada digit symbols for training purposes to accommodate different styles of writing. The dimensions of the images are 28×28 and digits are from zero to nine. A sample of Kannada digit representation as in Kannada-MNIST dataset is provided in Fig. 1.

This paper proposes a neural network architecture to classify Kannada digits. There is always a need to enhance image classification accuracy for neural network architecture. Researchers have adopted many ways to achieve this goal including fine tuning training parameters of neural network, number of epochs and learning rates. A situation which is to be avoided in deep learning is data over fitting. Data over fitting decreases the accuracy of predictions for a new unseen input instance. One way to avoid data over fitting is to provide more data for training. Generative adversarial network (GAN) is a good option to generate new training data with diverse coverage of possible new instances. This experiment demonstrates the impact of GAN generated data on classification accuracy in the context of epoch values.

The paper proposes a hybrid system where GAN produces sixty thousand additional images to train a neural network, and its impact on classification accuracy is measured. This paper explores the GAN and subsequent convolution neural network (CNN) network setup, behavior of setup with different epoch values of GAN and CNN, size of additional data set generated by GAN and a discussion on classification accuracy. The contributions of this paper are as follows:



Fig. 1 Sample digits from Kannada-MNIST dataset

- A neural network architecture to recognize and classify Kannada language digits.
- Design of generative adversarial network for data augmentation of Kannada digits samples.
- Analysis of classification accuracy corresponding to epoch values and augmented data.

This paper is organized as follows; Sect. 1 provides introduction about the experiment followed by a brief literature survey on GAN and CNN in Sect. 2. Section 3 provides the description on the experiment including the design of the neural network. Section 4 presents the results of the experiment along with its interpretation. The paper is concluded in Sect. 5.

2 Literature Survey

This section provides an overview of CNN and GAN concepts to accommodate the readers with the basics required to understand the experiment.

2.1 Convolution Neural Network

CNN design involves N dimension convolution layer, pooling strategy, and drop out nodes. 1D convolution is applied for a matrix input with either one row or one column. For example, an image matrix flattened for output dense layer accepts 1D matrix input. 2D convolution is applied on a matrix with row and column. For example, MNIST dataset images have dimensions 28×28 . 3D convolution is applied for three dimensional matrices. It can be imagined as a stack of three 2D matrices. For example, RGB images represent three-dimensional matrices. Batch normalization is employed to bind the values in the CNN layers within a range. This increases computational efficiency of the layers [6]. Pooling layer captures the features for image classification [2]. Max pooling is employed in this experiment which selects the maximum value in 2×2 pool window. Pooling keeps the size of the matrix in check after every layer of convolution. Application areas of CNN include radio imaging [13], cartography [14], radar imaging[11] and imaging applications on resource constrained platforms [4].

2.2 Generative Adversarial Network

GAN has a generator and a discriminator. During training, the generator becomes more efficient in generating new images and discriminator becomes efficient in rejecting the generated image as fake by comparing it with real image [5]. At the end of

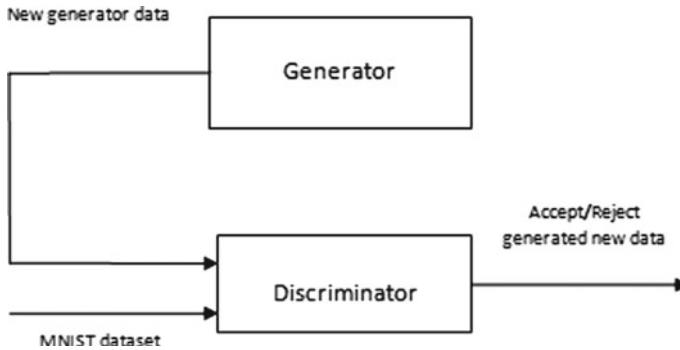


Fig. 2 Overview of a GAN

training, the generator will be capable of producing images which discriminator cannot recognize it as different from real image. An overview of GAN is shown in Fig. 2. GAN can produce additional datasets for training a neural network. Improving the resolution of images, synthesis of images from textual descriptions, medical researches are few applications of GAN [3].

3 Experimental Setup

This experiment explores the idea of using GAN and CNN introduced in Sects. 2.1 and 2.2 for producing additional data, supplementing MNIST data employed for CNN training. Data augmentation is a matured approach for ensemble of CNNs, classification of astronomical bodies [9], etc. Strategy to split the data into train and test set is required. We propose a 50:50 ratio split up of GAN generated data for each class of image. First half earmarked as train set will be merged with primary MNIST dataset. Other half is to be used as test set. Two rationales encouraged us to arrive at this arrangement for dataset;

- Introduction of new data into primary dataset with the aim of achieving better parameter tuning for a wide range of input variations [12].
- GAN-generated test set will provide new instances to the trained CNN.

An overview of experimental setup is given in Fig. 3. Generative adversarial network produces additional dataset of 60,000 images. Newly created dataset is split into train and test set in 50:50 ratios. 30,000 images are combined with Kannada MNIST dataset for training purposes. Remaining 30,000 will be deployed as test set. The accuracy of classification compared to a CNN in combination with variation of epochs in training. NVIDIA RTX 2080Ti hardware platform is employed for training the model.

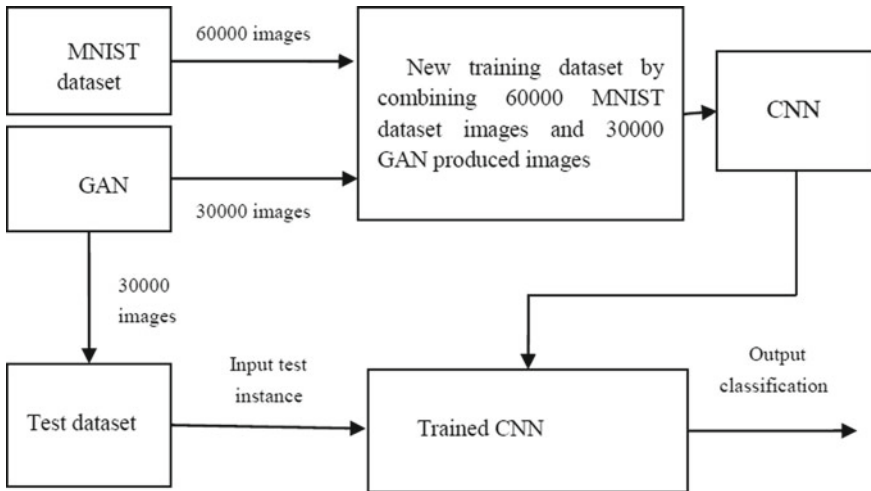


Fig. 3 Experimental setup

3.1 Design of Convolution Neural Network

Design of CNN involves deciding number of layers, number of nodes in each layer, activation function, and techniques for pooling and strategy for learning rate. Adam optimizer with learning rate 0.001 and sparse categorical cross-entropy as log loss is employed. Perceptron uses ReLu activation. Maximum pooling of size 2×2 is used. An overview of these is provided in Table 1.

3.2 Design of Generative Adversarial Network

In GAN, a generator produces images in a bottom-up approach until the discriminator cannot discriminate between real data and generator-produced data. Layers and their output shapes are shown in Table 2.

The discriminator examines the image produced by generator and rejects if comparison between real image and new image does not match. Efficiency of discriminator increases with higher number of epochs limited by computational requirement as a trade-off. Layers of discriminator with their output shapes are provided in Table 3.

Table 1 Parameter values of different layers in CNN

Layer type	Output shape
Conv2D	(None, 28, 28, 32)
Conv2D	(None, 28, 28, 32)
Batch normalization	(None, 28, 28, 32)
MaxPooling2D	(None, 14, 14, 32)
Dropout	(None, 14, 14, 32)
Conv2D	(None, 14, 14, 64)
Conv2D	(None, 14, 14, 64)
Batch normalization	(None, 14, 14, 64)
MaxPooling2D	(None, 7, 7, 64)
Dropout	(None, 7, 7, 64)
Conv2D	(None, 7, 7, 32)
Conv2D	(None, 7, 7, 32)
Batch normalization	(None, 7, 7, 32)
MaxPooling2D	(None, 3, 3, 32)
Dropout	(None, 3, 3, 32)
Flatten	(None, 288)
Dense	(None, 256)
Dropout	(None, 256)
Dense	(None, 10)

Table 2 Generator model of GAN

Layer type	Output shape
Dense	(None, 12544)
Batch normalization	(None, 28, 28, 32)
Batch normalization	(None, 12544)
Leaky ReLu	(None, 12544)
Reshape	(None, 7, 7, 256)
Convolution 2D transpose	(None, 7, 7, 128)
Batch normalization	(None, 7, 7, 128)
Leaky ReLu	(None, 7, 7, 128)
Convolution 2D transpose	(None, 14, 14, 64)
Batch normalization	(None, 14, 14, 64)
Leaky ReLu	(None, 14, 14, 64)
Convolution 2D transpose	(None, 28, 28, 1)

Table 3 Discriminator model of GAN

Layer type	Output shape
Convolution 2D	(None, 14, 14, 64)
Leaky ReLu	(None, 14, 14, 64)
Dropout	(None, 14, 14, 64)
Convolution 2D	(None, 7, 7, 128)
Leaky ReLu	(None, 7, 7, 128)
Dropout	(None, 7, 7, 128)
Flatten	(None, 6272)
Dense	(None, 1)

Table 4 Accuracy versus number of epochs by CNN

Epochs	Accuracy in percent
5000	94.25
10,000	97.40
15,000	97.75
20,000	95.85
25,000	94.75

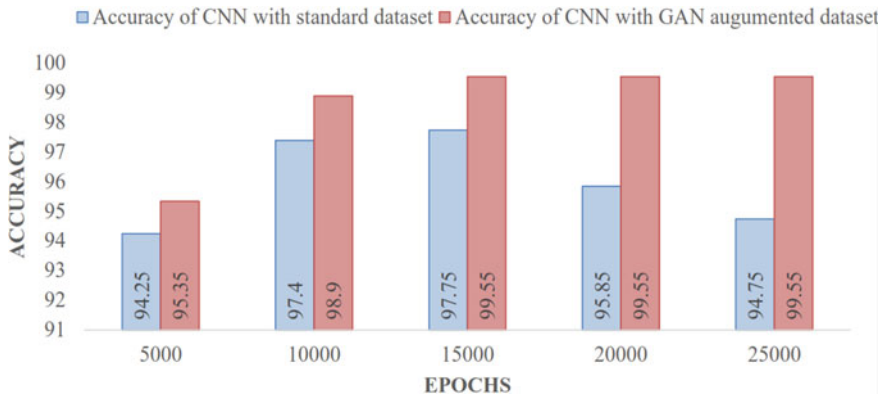
4 Results and Discussion

CNN presented in Sect. 3.1 is trained and tested on Kannada-MNIST dataset. Performance observed is tabulated in Tables 4 and 5. Table 4 presents accuracy of CNN without GAN data augmentation. Value of epoch should be in between 10,000 to 15,000 for good performance. Above these epoch values, accuracy decreases due to over fitting. Table 5 presents data on the performance of CNN when augmented with dataset produced by GAN. It can be observed that compared to Table 4, CNN when augmented with GAN data for training is resilient to over fitting. We propose that the additional GAN produced data should be at least fifty percent of original dataset to have an impact on CNN training. Below forty percent, the impact is marginal. A graphical representation of increase in accuracy achieved by the proposed GAN augmented training in comparison with standard dataset is provided in Fig. 4.

CNNs designed using different strategies has achieved 93.56 [7], 99.07 [8]. Arabic numeral recognition [1] achieves peak accuracy of 99.75%. Our results show GAN data augmentation makes CNN resilient to data over-fitting indicated by values of accuracy over different epoch values.

Table 5 Accuracy versus number of epochs by CNN trained with augmented data produced by GAN

Epochs	Accuracy in percent
5000	95.35
10,000	98.90
15,000	99.55
20,000	99.55
25,000	99.55

**Fig. 4** Epochs versus accuracy for CNN models with standard dataset and GAN augmented dataset

5 Conclusions

CNN can be used for classifying kannada language digits. This paper proposes a CNN design to achieve the same. The experiment explores the possibility of additional data generation using GAN for training and testing purpose. Accuracy of CNN increases with number of training instances available. Data collection and dataset creation require resources. Therefore, there is a need for techniques to generate additional data to train and test the CNN models. A large dataset created by combining data collected through sources and augmentation techniques is an attractive approach to improve performance of CNN. Our experiment split the GAN-generated data into train and test sets in 50:50 ratios. Analysis of results that were presented in Tables 4 and 5 draws the following conclusion:

- The phenomena of data over fitting reduced by 3.1%. This can be attributed to large dataset available for training.
- By examining the result, we can infer that the quality of data generated by GAN is acceptable as train and test data.
- Increasing the value of epoch does not yield increase in quality of generated data after a threshold value. In this experiment, the value is 10,000.

References

1. Ahamed P, Kundu S, Khan T, Bhateja V, Sarkar R, Mollah AF (2020) Handwritten Arabic numerals recognition using convolutional neural network. *J Ambient Intell Hum Comput* 11(11):5445–5457
2. Akhtar N, Ragavendran U (2020) Interpretation of intelligence in CNN-pooling processes: a methodological survey. *Neural Comput Appl* 32(3):879–898
3. Alqahtani H, Kavakli-Thorne M, Kumar G (2021) Applications of generative adversarial networks (GANs): an updated review. *Arch Comput Methods Eng* 28(2):525–552
4. Anupama VG, Kiran A (2022) Extrapolating z-axis data for a 2d image on a single board computer. In: Saraswat M, Roy S, Chowdhury C, Gandomi AH (eds) *Proceedings of international conference on data science and applications. Lecture notes in networks and systems*, vol 288. https://doi.org/10.1007/978-981-16-5120-5_38
5. Cheng K, Tahir R, Eric LK, Li M (2020) An analysis of generative adversarial networks and variants for image synthesis on MNIST dataset. *Multimedia Tools Appl* 79(19):13725–13752
6. Jiang X, Lu M, Wang SH (2020) An eight-layer convolutional neural network with stochastic pooling, batch normalization and dropout for fingerspelling recognition of Chinese sign language. *Multimedia Tools Appl* 79(21):15697–15715
7. Lei F, Liu X, Dai Q, Ling BWK (2019) Shallow convolutional neural network for image classification. *SN Appl Sci* 2(1):97
8. Madakannu A, Selvaraj A (2020) Digi-net: a deep convolutional neural network for multi-format digit recognition. *Neural Comput Appl* 32(15):11373–11383
9. Mittal A, Soorya A, Nagrath P, Hemanth DJ (2020) Data augmentation based morphological classification of galaxies using deep convolutional neural network. *Earth Sci Inform* 13(3):601–617
10. Prabhu VU (2019) Kannada-MNIST: a new handwritten digits dataset for the kannada language. *CoRR* abs/1908.01242
11. Wan J, Chen B, Xu B, Liu H, Jin L (2019) Convolutional neural networks for radar HRRP target recognition and rejection. *EURASIP J Adv Sign Process* 1:5
12. Xu Y, Goodacre R (2018) On splitting training and validation set: a comparative study of cross-validation, bootstrap and systematic sampling for estimating the generalization performance of supervised learning. *J Anal Test* 2(3):249–262
13. Yamashita R, Nishio M, Do RKG, Togashi K (2018) Convolutional neural networks: an overview and application in radiology. *Insights Imaging* 9(4):611–629
14. Zhou H, Ummerhofer B, Brox T (2020) DeepTAM: deep tracking and mapping with convolutional neural networks. *Int J Comput Vis* 128(3):756–769

Sparse Autoencoder-Based Speech Emotion Recognition



Vishal Balaji Sivaraman, Sheena Christabel Pravin , K. Surendaranath, A. Vishal, M. Palanivelan, J. Saranya, and L. Priya

Abstract One of the most natural methods for humans to express themselves is through speech. People nowadays are drawn to alternative ways of communication including emails, text messages, and the usage of emoticons to express their feelings. Given the importance of emotions in communication, recognizing and analyzing them are crucial in today's digital age of remote communication. As emotions are complicated, recognizing them can be difficult. There is no universally accepted method for quantifying or categorizing them. A speech emotion recognition system is defined as a set of methods for processing and categorizing speech signals to detect the emotions inherent within them. This work proposed a hybrid model, namely the sparse autoencoder-multi-layer perceptron (SAE-MLP) model. The SAE model is used for feature extraction and the MLP for the categorization of speech emotions.

Keywords Speech · Emotion · Features · Classification · Sparse autoencoder · Multi-layer perceptron

1 Introduction

There are three types of characteristics in speech: lexical characteristics such as the vocabulary; visual characteristics, viz. the speaker's expressions; and auditory characteristics like pitch, tone, and jitter. One or more of these attributes can be used to address the challenge of speech emotion recognition (SER). If emotions were to be predicted from real-time audio, a transcript of the speech would be required, which would demand a second stage of text extraction from speech. Further visual analysis requires access to video recordings of conversations that are not available in all situations, but there is a significant demand for audio, allowing acoustic analysis

V. B. Sivaraman · S. C. Pravin (✉) · K. Surendaranath · A. Vishal · M. Palanivelan · J. Saranya
Department of Electronics and Communication Engineering, Rajalakshmi Engineering College,
Chennai, India
e-mail: sheena.s@rajalakshmi.edu.in

L. Priya
Department of Information Technology, Rajalakshmi Engineering College, Chennai, India

to be performed in real time while the conversation is ongoing because there is a significant need for audio data to complete the task. As a result, it was decided to concentrate on the acoustic components of speech. Furthermore, there are also two methods of representing emotions:

- Discrete Classification: Sorting emotions under discrete labels such as angry, happy, surprise, and fear.
- Dimensional Representation: Emotions are represented using variables such as valence, which ranges from negative to positive scale, activation or vitality, which is on a low to high scale, and dominance, which ranges from active to passive scale.

Both systems have pluses and minuses. The dimensional approach is more complex and provides more detail for prediction, but it is also harder to execute and shortage of annotated audio data. The discrete approach is more straightforward and simpler to execute, but it lacks the predictive context that the dimensional representation provides. The discrete classification strategy was used in this investigation due to a lack of dimensionally annotated data in the public domain.

The use of speech to recognize emotions has gained popularity in the scientific community. Several efforts have been done earlier by numerous scholars throughout the world on the same subject. Identification of eight different emotions [1] has been experimented by extracting pitch and energy, mel-frequency cepstral coefficients (MFCC), and mel energy spectrum dynamic coefficients (MEDC) from speech signals with an accuracy of 50%. Convolutional Neural Network (CNN)-based speech emotion detection system [1] could recognize 6 classes of emotion. A speech emotion recognition system [2] that uses a deep neural network (DNN) that can recognize emotions from a one-second frame of raw speech spectrograms was also built with the eNTERFACE database and the Surrey Audio-Visual Expressed Emotion (SAVEE) database as training datasets. In the recent literature, hybrid machine learning models [3, 5] have found popularity owing to their greater performance and reduced complexity. This research work also aims to introduce one such hybrid model for speech emotion recognition.

1.1 Contributions

The significant contributions of this research paper are as follows:

- A hybrid model, namely the SAE-MLP model, is proposed for speech emotion recognition
- Model regularization has been introduced in the form of drop out at each layer of the sparse autoencoder to avoid over-fitting.
- Fine hyper-parameter tuning has been introduced to make the proposed model efficient in categorizing speech emotions with high accuracy and precision.

2 Speech Emotion Dataset

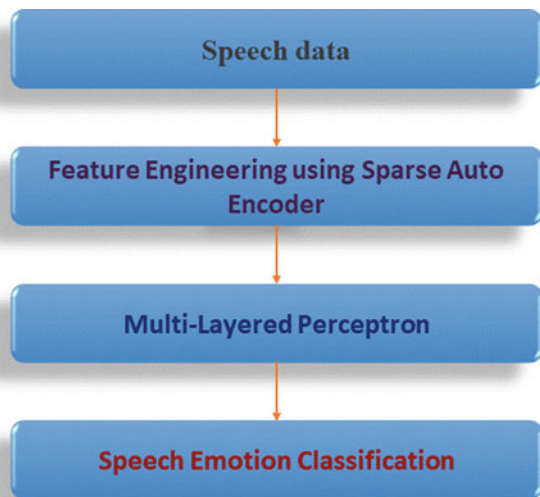
In this research, the Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) was employed. The collection contains 7356 files. Twenty-four professional actors in the database vocalize two lexically matched sentences in a neutral North American accent. Speech has calm, happy, sad, angry, terrified, surprise, and disgust expressions, whereas music contains peaceful, happy, sad, angry, and scary emotions. There are two levels of emotional intensity such as normal and strong for each expression and a neutral expression.

3 Proposed SAE-MLP Model

The proposed model workflow is depicted in the schematic representation as shown in Fig. 1. Initially, the raw speech input signal should be provided to the sparse autoencoder which performs automatic feature engineering. The encoder part of the SAE creates a new latent representation of the input speech signal, while the decoder reconstructs the latent features to bring out the same speech signal. After adequate training, the decoder of the SAE is removed and the latent representations, which have reduced dimensionality, are provided as input to the multi-layer perceptron (MLP) classifier. The overall block diagram of the proposed framework is given in Fig. 1.

The MLP [3] classifies the signals into definite speech emotion classes after being trained on the latest features from the SAE. Thus, the sparse autoencoder acts as a

Fig. 1 Pipeline structure for final prediction



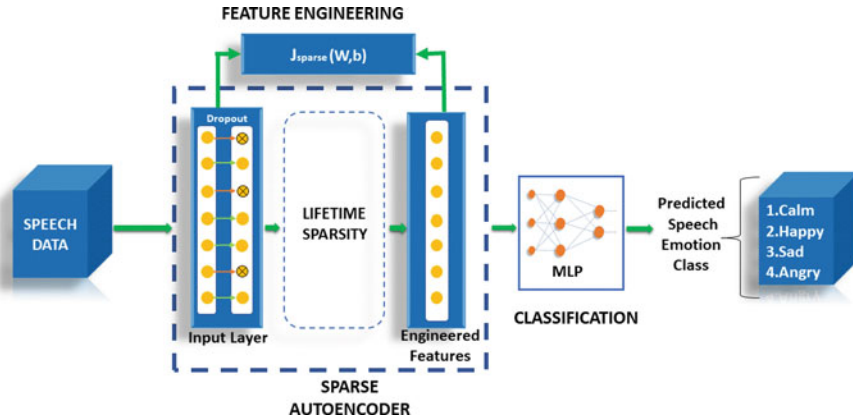


Fig. 2 Proposed SAE-MLP model architecture

generative model, while the MLP functions are the discriminative model. A detailed description of the proposed model is picturized in Fig. 2.

3.1 Train/Test Data Allocation

The complete dataset is divided in a ratio of 70:30, with 70% of the dataset being used for model training and 30% for model validation. This method [4] is widely used for training and validating the proposed sparse autoencoder and the model.

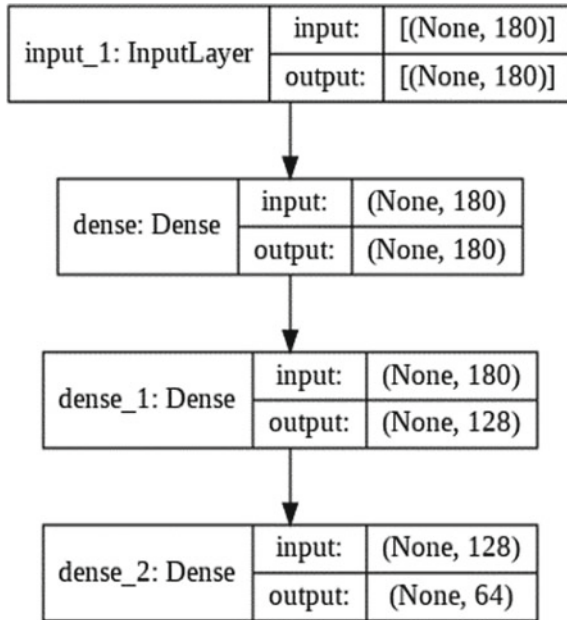
3.2 Feature Engineering Using Sparse Autoencoder

The sparse autoencoder model was utilized to reduce dimensionality, resulting in the best features being allocated to the suggested pipeline model for emotion analysis. The proposed autoencoder model was trained for a total of 10,000 epochs during the feature engineering phase.

3.3 Model Summary of a Sparse Autoencoder

The proposed sparse autoencoder model comprises an encoder and decoder section, both of which are made up of three layers of dense networks each with a distinct count of neurons.

Fig. 3 Sparse autoencoder’s encoder block model summary

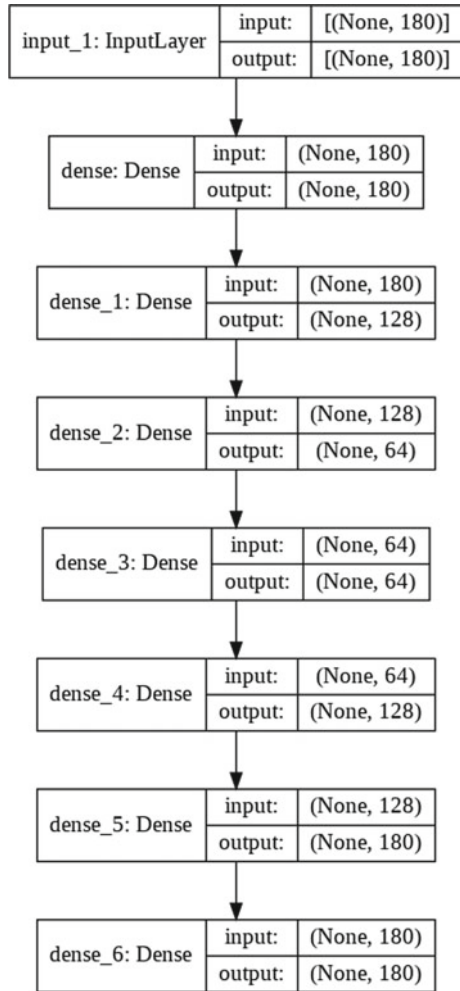


The number of neurons in the first layer of the encoder section is adjusted to the number of features (180), as shown in Fig. 3. Similarly, the number of neurons in the final layer of the decoding portion is adjusted to the number of features (180). In addition, all layers’ activation functions have been modified to Rectified Linear Unit (ReLU) with a constant learning rate, as shown in Fig. 4.

3.4 Multi-layer Perceptron

Multi-layer perceptron (MLP) is a kind of feed-forward artificial neural network (ANN) as shown in Fig. 5. The term MLP is ambiguous; it is applicable to any feed-forward ANN [5] or networks composed of many layers of perceptron with threshold activation. MPL has three levels of nodes: an input layer, a hidden layer, and an output layer. Each node except the input node is a neuron with a nonlinear activation function. Backpropagation is a supervised learning method used by MLP during training. The multi-layered and nonlinear activation of MLP helps to distinguish it from linear perceptrons. MLP excels at discriminating data that is not linearly divisible. The multi-layer perceptron consists of two types of models that follow the same mechanism, one for regression (MLP regressor) and the other for classification (MLP classifier). As a result, the classification model (MLP classifier) has been chosen for examination and substantiation of the earlier assertion.

Fig. 4 Sparse autoencoder model summary



4 Model Evaluation

SAE model evaluation was performed using accuracy and loss metrics for each epoch, whereas for the MLP classifier model, score metrics such as classification accuracy score, balanced accuracy score, Cohen’s Kappa score, *F1* score, and Jaccard score were used [5]; similarly in the case of loss metrics, Hamming loss metric was used; and finally, confusion matrix was drafted to determine the individual class predominance. The metrics are computed using Eqs. (1) to (6).

$$\text{Jaccard Score} = \frac{|A \cap B|}{|A \cup B|} \tag{1}$$

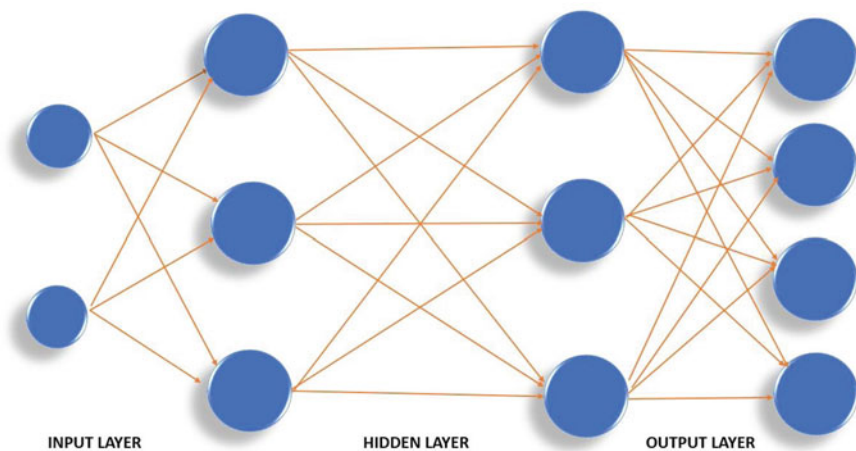


Fig. 5 Multi-layer perceptron architecture

$$\text{Cohen Kappa Score} = \frac{P_0 - P_e}{1 - P_e} \quad (2)$$

$$F1\text{score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (3)$$

$$\text{Balanced Accuracy} = \frac{\text{Sensitivity} + \text{Specificity}}{2} \quad (4)$$

$$\text{Classification Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (5)$$

$$\text{Hamming Loss} = \frac{1}{nL} \sum_{i=1}^n \sum_{j=1}^L [I(y_j^{(i)} \neq y_j'^{(i)})] \quad (6)$$

The Jaccard score [6, 7] is calculated by substituting the set of labels predicted by the model, and the actual set of labels in Eqs. (1) and (2) was used for computing the Cohen's Kappa score [8, 9] for the model upon substitution of P_0 and P_e , where P_0 is the ratio of observed agreement and P_e is the expected agreement when both annotators assign labels at random. The $F1$ score for the model is quantified by substituting the values of precision and recall in Eqs. (3) and (4) that were used for reckoning the balanced accuracy score for the model, upon substituting the values of sensitivity and specificity. Substituting the components TP (true positive), TN (true negative), FP (false positive), and FN (false negative) into Eq. (5) yields the accuracy score for the model. Aside from that, the Hamming loss [10] for a model was enumerated using the set of labels predicted by the model and the actual set of

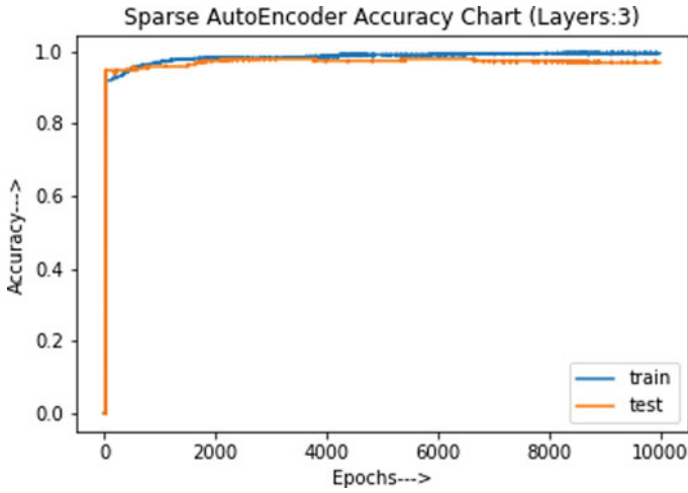


Fig. 6 Sparse autoencoder accuracy chart

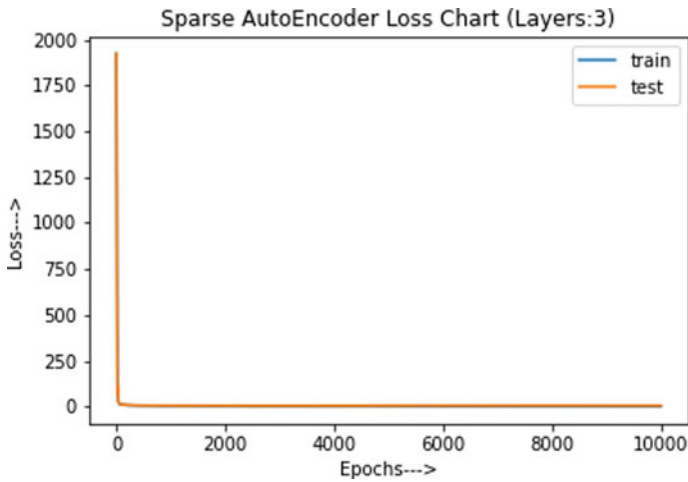


Fig. 7 Sparse autoencoder loss chart

labels, as illustrated in Eq. (6). Finally, a confusion matrix for the model is drafted by utilizing the components required in computing a model’s accuracy score.

Table 1 Evaluation of the Proposed Model

Metrics	MLP classifier with sparse encoder (%)	MLP classifier without sparse encoder (%)
Classification accuracy score	95	55
Balanced accuracy score	95	54
F1 score (Macro)	95	51
F1 score (Micro)	95	55
F1 score (Weighted)	95	52

4.1 Sparse Autoencoder Evaluation

As depicted in Figs. 6 and 7, the proposed autoencoder model achieves a training score of 0.9948 (99.48%) and a validation score of 0.9688 with minimal training and validation loss of 0.05.

4.2 Multi-layer Perceptron (MLP) Classifier with and Without Sparse Autoencoder Results

An experimentation on the performance of the MLP model with and without the SAE model was executed. The MLP’s performance was well-enhanced when the SAE-based latent features were used to train it. The performance of MLP dipped when the SAE model was removed. The results rendered in Tables 1 and 2 represent the results concerning the inclusion and exclusion of sparse autoencoder model with the MLP classifier model. Figures 8 and 9 represent the confusion matrix heatmap

Table 2 Performance of the Proposed Model

Metrics	MLP classifier with sparse encoder (%)	MLP classifier without sparse encoder (%)
Cohen’s Kappa score	94	39
Jaccard score (Macro)	91	36
Jaccard score (Micro)	91	38
Jaccard score (Weighted)	91	37
Hamming loss	4.6875	45.3125

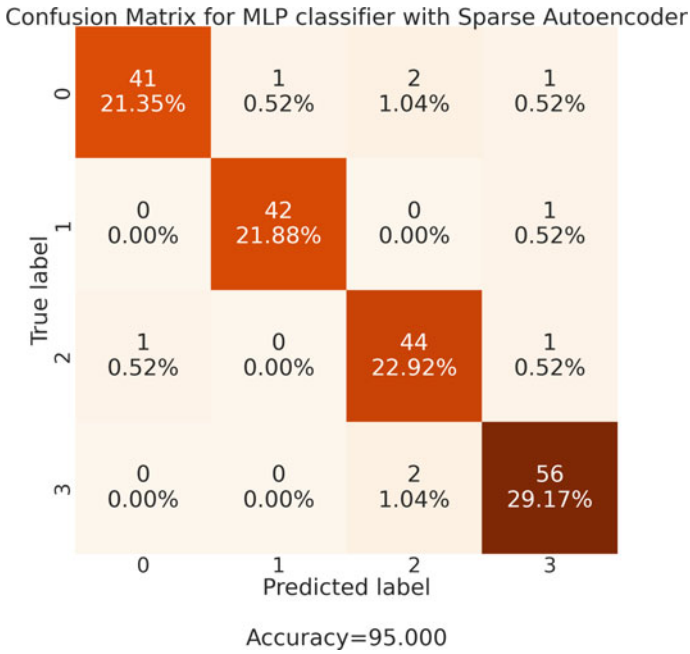


Fig. 8 Confusion matrix heatmap for MLP classifier with sparse autoencoder

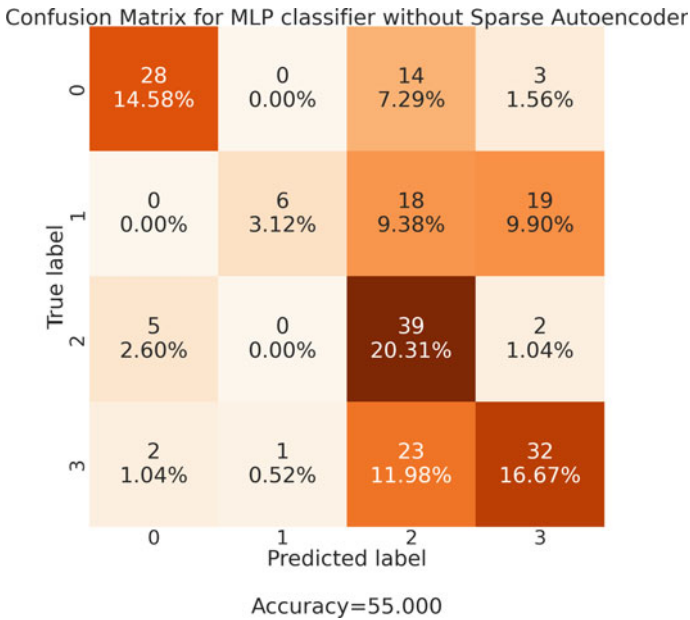


Fig. 9 Confusion matrix heatmap for MLP classifier without sparse autoencoder

for the respective models.

5 Conclusions and Future Scope

Sparse autoencoder was appended with the multi-layer perceptron classifier model to produce a hybrid SAE-MLP model, which has been proved to improve speech emotion classification. On experimentation, it was observed that the multi-layer perceptron did not perform well over the given speech emotion dataset, and so, SAE-based automated feature engineering was employed to enhance the classification performance of the MLP. This research paper has introduced a novel hybrid model SAE-MLP for speech emotion detection, with enhanced accuracy and precision.

In the future, the speech features, namely the MFCC, RAS-MFCC, LPCC, PLP, and harmonic cepstrum, would be experimented on the speech emotion recognition system.

Acknowledgments This project is partially aided by the All-India Council of Technical Education, India, under the Research Progress Scheme (Ref. 8-40/RIFD/RPS/Policy-1/2017-18)—15th March 2019.

Declaration of Conflict of Interest The authors alone are responsible for the content and writing of the paper, and they report no conflict of interest.

References

1. Juyal S, Killa C, Singh GP, Gupta N, Gupta V (2021) Emotion recognition from speech using deep neural network. In: Srivastava S, Khari M, Gonzalez Crespo R, Chaudhary G, Arora P (eds) Concepts and real-time applications of deep learning. EAI/Springer innovations in communication and computing. Springer, Cham. https://doi.org/10.1007/978-3-030-76167-7_1
2. Huang F, Zhang J, Zhou C et al (2020) A deep learning algorithm using a fully connected sparse autoencoder neural network for landslide susceptibility prediction. *Landslides* 17:217–229. <https://doi.org/10.1007%2Fs10346-019-01274-9>
3. Khalil RA, Jones E, Babar MI, Jan T, Zafar MH, Alhussain T (2019) Speech emotion recognition using deep learning techniques: a review. *IEEE Access* 7:117327–117345. <https://ieeexplore.ieee.org/document/8805181>
4. Chollet F (2016) Building autoencoders. In: Keras. <https://blog.keras.io/building-autoencoders-in-keras.html>
5. Pravin SC, Palanivelan M (2021) A hybrid deep ensemble for speech disfluency classification. In: Circuits, systems, and signal processing, vol 40, No 8. Springer, Berlin, pp 3968–3995. https://www.researchgate.net/publication/349228170_A_Hybrid_Deep_Ensemble_for_Speech_Disfluency_Classification
6. Krishnan PT, Joseph Raj AN, Rajangam V (2021) Emotion classification from speech signal based on empirical mode decomposition and non-linear features. *Complex Intell Syst* 7:1919–1934. <https://doi.org/10.1007%2Fs40747-021-00295-z>

7. Mohamad Nezami O, Jamshid Lou P, Karami M (2019) ShEMO: a large-scale validated database for Persian speech emotion detection. *Lang Resources Eval* 53:1–16. <https://doi.org/10.1007%2Fs10579-018-9427-x>
8. Deshpande M, Rao V (2017) Depression detection using emotion artificial intelligence. In: *International conference on intelligent sustainable systems (ICISS)*, pp 858–862. <https://ieeexplore.ieee.org/document/8389299>
9. Stolar MN, Lech M, Bolia RS, Skinner M (2017) Real-time speech emotion recognition using RGB image classification and transfer learning. In: *11th International conference on signal processing and communication systems (ICSPCS)*, pp 1–8. <https://ieeexplore.IEEE.org/document/8270472>
10. Pravin SC, Palanivelan M (2021) Regularized deep LSTM autoencoder for phonological deviation assessment. *Int J Pattern Recogn Artif Intell* 35(4):2152002. <https://doi.org/10.1142/S0218001421520029>

Hyperspectral Image Classification Using Transfer Learning



Usha Patel, Smit Patel, and Preeti Kathiria

Abstract Hyperspectral images offer abundant spectral details for spatially related material to be identified and distinguished thorough analysis. Based on spectral data and spatial correlation, a broad range of advanced classification techniques is possible. Computer technological advances have fostered the growth of modern, efficient deep-learning (DL) techniques that show a wide variety of applications with encouraging performance. Particularly in the field of remote sensing information gathered by Earth Observer (EO) instruments, deep-learning techniques have been successfully used. Given the abundance of information contained in this type of pictures, hyperspectral imaging (HSI) is one of the major topics in remote sensing research which enables greater earth surface analysis and exploitation through the combination of rich spatial and spectral information. Given the high dimensions of the data and the restricted supply of training samples, HSI presents significant difficulties for supervised classification methods. Transfer learning architectures having the great potential in the classification of HSI information have been established recently to resolve some constraints. This paper provides an experiment of HSI classification using transfer learning. Experiment results show that limited number of training sample model gives better accuracy.

Keywords Deep learning · VGGNet · Hyperspectral image classification · CNN · Transfer learning

U. Patel · S. Patel · P. Kathiria (✉)
Institute of Technology, Nirma University, Ahmedabad, India
e-mail: preeti.kathiria@nirmauni.ac.in

U. Patel
e-mail: ushapatel@nirmauni.ac.in

S. Patel
e-mail: 17bit078@nirmauni.ac.in

1 Introduction

The emerging analytical method focused on spectroscopy is hyperspectral imaging (HSI). The hyperspectral images provide the details of visible bands along with infrared bands. The data obtained from the hyperspectral image is in the form of a cube, where x and y dimension represents the spatial scale, and the third dimension is the spectral contents [1].

In a broad range of uses, hyperspectral satellite imagery is used. Although initially it was used for mining and geology, now it is extended into fields that are so widespread as agriculture and monitoring, as well as historical document study [2]. Some of its applications are agriculture, eye care, food processing, mineralogy, surveillance, astronomy, chemical imaging, environment and many more.

HSI may also leverage the power of structural associations between different spectra in a vicinity, enabling for even more comprehensive spectral-structural approaches for more precise image processing and classification.

1.1 Advantages and Disadvantages of HSI

Hyperspectral imaging generates images, consisting of hundreds of bands, whereas multispectral imaging generates images of up to ten broader bands. A greater level of spectral information in HSI improves one's possibility to see the invisible.

The main drawbacks are costing and sophistication. To analyse spectral information, powerful devices, sensitive detectors and huge storages are needed. As all uncompressed hyperspectral cubes are massive multidimensional data which require huge data storage space. Most of these aspects boost the cost of collecting and analysing hyperspectral data significantly.

1.2 HSI Classification Models

Since last three decades, various supervised and unsupervised machine learning techniques are used for HSI classification. Compared to unsupervised learning techniques, supervised machine learning algorithm has given better performance. In literature, authors used supervised algorithms like Naive Bayes, support vector machine (SVM), random forest, etc. [3]. In supervised machine algorithm, feature extraction was done manually, where the performance of the classification highly relies on the extracted features. Nowadays, deep-learning (DL) is highly used for computer vision task. DL is a subset of machine learning, where the model can able to extract low level of features automatically. There are various deep-learning models used, among all convolutional neural network (CNN) is widely used for image processing algorithm [4]. CNN is having unique convolutional layer which can able to consider the

correlation of neighbour pixels. Due to that, it can able to consider spatial correlation for feature extraction.

To train customised CNN network, huge amount of labelled samples are required. Sample labelling is very costly and time-consuming task in the field of hyperspectral images. There are various approaches to overcome the problem of less number of labelled samples. People use semi-supervised learning, active learning and transfer learning for the same. Here, in our paper, we use transfer learning. Transfer learning is one of the machine learning methods, in which model is trained with dataset having enough labelled samples. Now, the learned weights can be transferred to different datasets with limited labelled samples. Here, we use pre-trained model VGGNet 16. This model was trained with ImageNet dataset.

1.3 Scope of the Paper

The scope of this paper is to improve hyperspectral image classification using deep-learning methods with limited number of training sample. This paper presents results of HSI classification using the predefined transfer learning model for HSI classification on widely used dataset, i.e. Indian pines.

1.4 Contribution

The paper suggests a prediction model for the classification of spatial HSI datasets using deep learning. For this purpose, we have used pre-trained transfer learning model to build our classification model. In this article, the prime objective is to develop a hybrid model that includes VGG-16 framework, an advanced neural network, which also has additional layers of CNN2d, global average pool and dense following the basic model of pre-trained VGG-16 transfer learning model.

2 Related Work

A variety of HSI classification approaches have been suggested in recent years. Several supervised classification approaches have been previously studied and amended for HSI classification in the computer vision field.

Support vector machine is a prominent classification approach for hyperspectral data classifications that is presented by Melgani et al. [5]. The Hughes effect is unlikely to impact SVM because of its poor sensitivities to large dimensionality [6]. SVM-based models, in most circumstances, outperform other popularly used image recognition algorithms in terms of classification [6]. Such algorithms were the cutting-edge algorithms for a long time [7].

Chen et al. [8] used the technique of deep learning in the categorisation of hyperspectral imaging. The usefulness of a stacked automatic encoder in comparison to the standard spectral knowledge-based classifying was investigated in this work. Experimental findings with popularly used hyperspectral datasets demonstrate that the classification algorithm produced in this deep-learning framework performs efficiently.

After AlexNet managed to win the ImageNet classification challenge in 2012, deep-learning techniques have been a trending topic. AlexNet can retrieve top discriminatory characteristics. Several deep-learning approaches for HSI classification have recently been developed. Chen et al. [8] attempted to incorporate PCA-dependent dimensionality reductions and logistic regression classification a deep spatial domain approach based on sparse auto-encoder (SAE) stacked for HSI classification.

Paoletti et al. [9] build and compare different machine learning and deep-learning models for hyperspectral image classification. They have implemented six different methods and also have provided their results, and comparison of different models includes an in-depth analysis of the latest state of the art in deep learning for hyperspectral image classification, discussing the strengths and shortcomings of the most commonly used classification models in the research. They have presented comparative analysis for every described approach using many commonly used hyperspectral image scenes, giving a comprehensive overview of the explored strategies.

He et al. [10] proposed a transfer learning approach for HIS classification using custom-made CNN model. Digital pictures and HSI differ significantly. Therefore, heterogeneous transfer learning based on CNN is introduced for a hyperspectral image classification in order to make use of the potential of transfer learning. In this, they have proposed a solution using the weight taken from only first seven layers of VGGNet 16 and transfer that weight to custom-made CNN model. In this paper, attention function is applied in order to minimise the negative impact of the variation between the digital image and HSI. In this, they have achieved 87% and 93% testing accuracy using 200 samples from both Indian pine and Salinas Valley dataset.

Li et al. [11] suggested a deep belief network (DBN) strategy for hyperspectral image analysis. The model consisted of a pile of limited Boltzmann machines that have been trained using greedy model unsupervised learning. The spectral features of the pictures were not exploited mostly by the DBN modelling because the data was reduced to the three main PCA components.

Patel et al. [12] consider diversity and uncertainty to select sample to be labelled in active learning approach for hyperspectral image classification. Recently, Shekhawat et al. [13] developed an approach for data transformation to achieve good accuracy. Patel et al. [14] used semi-supervised approach with CNN classifier for HSI classification.

Fig. 1 Indian Pine ground truth



3 Methodology

This section explores the pre-processing of the HSI and the hybrid model built for HSI classification, and the model consists of predefined transfer learning model.

3.1 Dataset

For our experiment, here we used publicly available and most popular dataset Indian Pines.

Indian Pines [15]: The hyperspectral image is collected by an AVIRIS sensor in the Northwest Indiana, consisting of 145×145 pixels and 224 spectral bands. This image is a portion of big image. Two-thirds of cultivation and one-third of a woodland or other natural seasonal crops are present at the Indian Pine scene. There are two main lanes, a train line, high-way and some low population homes, some buildings and minor roads. In June, a few of the crop species are present, maize and soybeans, which are less than 5 per cent in the early development stage. There are total 16 classes present in ground truth of this scene. Figure 1 shows ground truth image of Indian Pine dataset. Table 1 represents different classes and number of samples in each class of Indian pine dataset.

3.2 Pre-processing

For reducing the dimension of the hyperspectral image, we have used principal component analysis (PCA). PCA helps to reduce spectral redundancies and the

Table 1 Classes in Indian Pine dataset

S. No.	Class	Samples
1	Alfalfa	46
2	Corn-notill	1428
3	Corn-mintill	830
4	Corn	237
5	Grass-pasture	483
6	Grass-trees	730
7	Grass-pasturemowed	28
8	Hay-windrowed	478
9	Oats	20
10	Soybean-notill	972
11	Soybean-mintill	2455
12	Soybean-clean	593
13	Wheat	205
14	Woods	1265
15	Buildings-grass-trees-drives	386
16	Stone-steel towers	93

dimensions while maintaining intact spatial knowledge [16]. In this implementation, we have reduced three number of bands.

Figure 2 shows the similarity between the image obtained after the pre-processing steps and the original ground truth image. The image on the left is the pre-processed image, and the image on the right is the ground truth image.

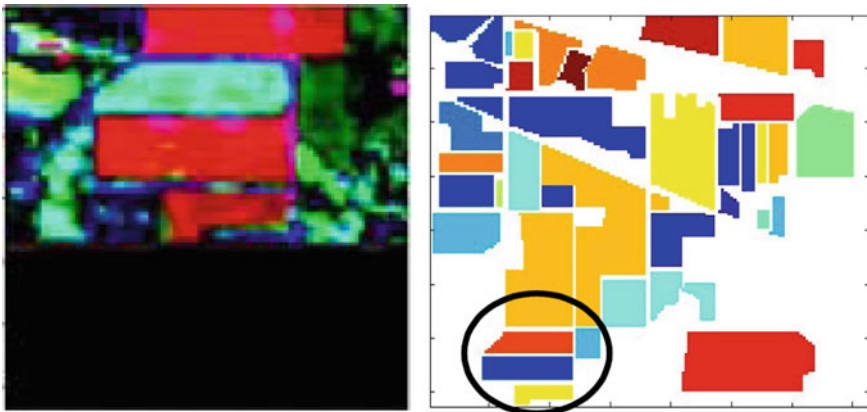


Fig. 2 Similarity between pre-processed image and ground truth image

3.3 Classifiers

For the task of classifying HSI, we have used pre-trained VGG-16 [17] transfer learning model. VGG-16 is a 16-layer model, and it mainly consists of CNN and max pooling layers. And additional layers of CNN, dense and global average pooling are added to extract more features and hence improve accuracy. ImageNet weights are being used as initial weights. Here, we use input dimensions, and it is $64 \times 64 \times 3$. This model is trained with three different percentage of training samples and 10% validation samples of total samples, and the remaining samples are used for testing.

The input in first convolution layer is 64×64 RGB picture, which is predefined. The image is surpassed by a stack of convolutionary layers, in which the filtering is utilised in a quite narrow field, namely 3×3 . It moreover uses 1×1 CNN filters, in one of the configurations, which may also be considered as a linear change of the incoming channels and are then followed by nonlinearity. The padding step is set to 1 pixel, the positional padding of the inputs of the fully connected layers. Layers are sufficient that after the CNN layers, the spatial quality is retained, i.e. the pad is one pixel per 3×3 CNN layers. Pooling takes place with five different levels of max pooling that are accompanied by segments of the convolution layers. Max pooling takes place across a perimeter of 2×2 pixels, using 2 as a stride. After this VGG-16 model, extra CNN, dense and global average pooling layers have been added to improve the accuracy of the model. The architecture of the described model is shown in Fig. 3.

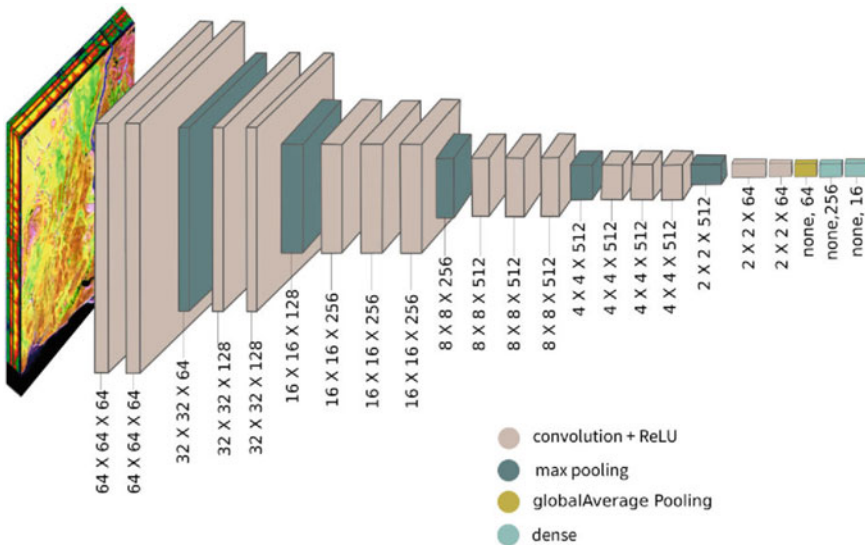


Fig. 3 Model for Indian Pine dataset

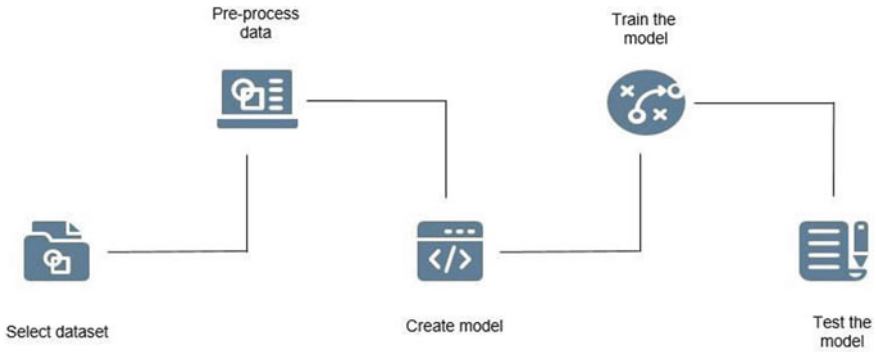


Fig. 4 Implementation flow

4 Experimental Work

This section gives a brief idea about the experimental steps used in the proposed architecture in Fig. 4. It gives details of the dataset preparation, data pre-processing, model architecture and hyperparameters tuning.

4.1 Training

The created model is trained with 1, 3 and 5% of dataset for each dataset. 10% of the dataset is used for validation, and the remaining dataset is used for testing purpose. And the model ran for 30 epochs.

Figure 5 shows the accuracy/loss graphs for different amount of Indian pine dataset. With 1% dataset, model is able to reach 99% training accuracy and around 80% validation accuracy while training loss is near to none and validation loss is around 1. However, the results have been improved using more amount of dataset. Using 3% dataset, model was able to achieve 99% training accuracy and 95% validation accuracy, and both training and validation losses were less than 0.3. The model got near to 100% training accuracy and 96% validation accuracy using only 5% of dataset, and both the losses were less than 0.25.

5 Result and Discussion

This section covers the results of testing samples. We have used precision, recall and *F1* score as evaluation metrics and also have made confusion matrix for the same.

From Table 2, we can observe that the model trained with 1% dataset is unable to classify the classes in which there are samples less than 30, and this could be because

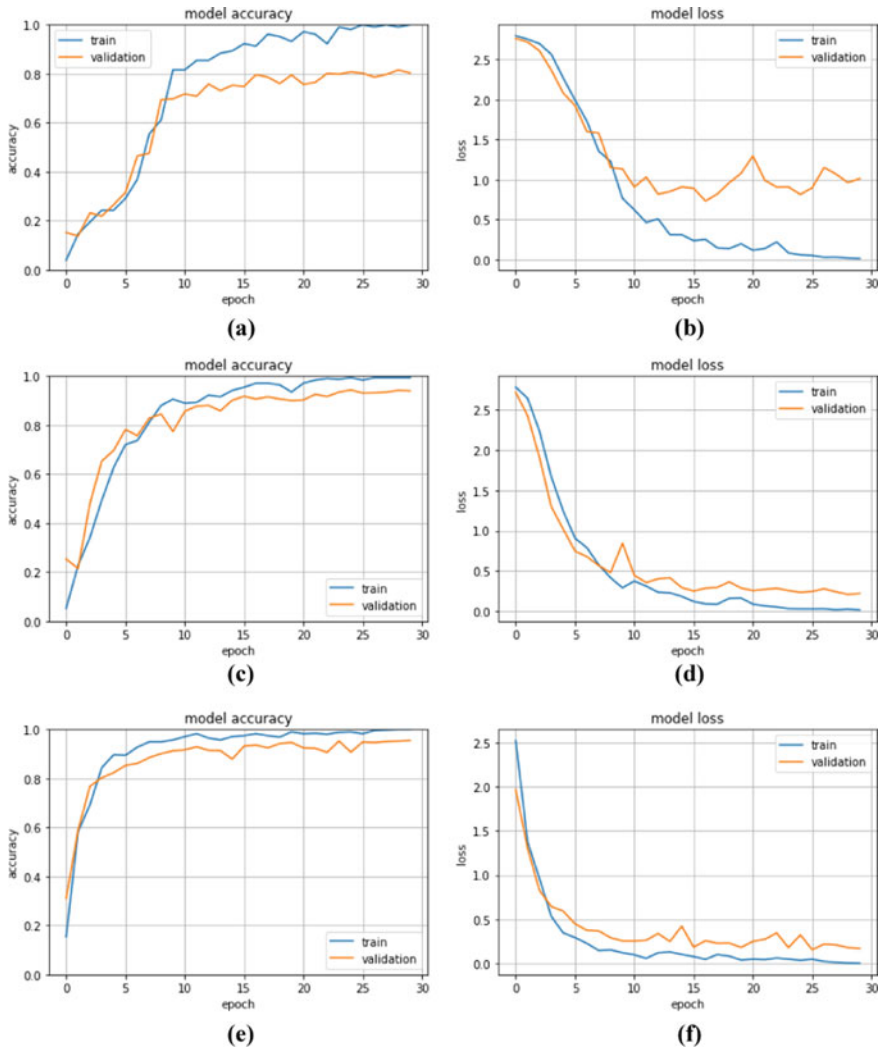


Fig. 5 a Accuracy, b loss with 1% of training data, c accuracy, d loss with 3% of training data, e accuracy, f loss with 5% of training data

of the smaller number of availabilities of training samples of that particular classes. Overall testing accuracy is 80%. The model trained with 3% dataset overcomes the problem with the model trained with 1% dataset and therefore is able to classify the classes in which there are samples less than 30. Overall testing accuracy is 93%. Finally, the model trained with 5% dataset performs slightly better as compared with the model trained with 3% dataset and is able to classify almost all the classes accurately. We got overall testing accuracy as 95%.

Table 2 Test result on Indian pine dataset

	1% Training dataset (89% testing dataset)			3% Training dataset (87% testing dataset)			5% Training dataset (85% testing dataset)			
	Precision	Recall	F1 Score	Precision	Recall	F1 Score	Precision	Recall	F1 score	Support
0	0.26	0.95	0.41	1	0.59	0.74	0.76	0.97	0.86	40
1	0.75	0.7	0.72	0.88	0.92	0.9	0.94	0.95	0.94	1221
2	0.65	0.61	0.63	0.85	0.89	0.87	0.87	0.98	0.92	709
3	0.94	0.63	0.76	0.87	1	0.93	0.84	0.75	0.8	202
4	0.7	0.82	0.75	0.88	0.93	0.9	0.99	0.92	0.95	413
5	0.74	0.86	0.8	0.89	0.99	0.94	0.98	0.93	0.95	624
6	0	0	0	1	0.83	0.91	1	1	1	24
7	0.91	1	0.95	1	1	1	1	1	1	409
8	0	0	0	0.63	1	0.77	0.81	1	0.89	17
9	0.87	0.67	0.76	0.96	0.93	0.95	0.95	0.97	0.96	831
10	0.87	0.9	0.89	0.98	0.94	0.96	0.96	0.97	0.97	2099
11	0.68	0.81	0.74	0.95	0.79	0.86	0.98	0.84	0.91	507
12	0.83	0.74	0.78	0.86	0.96	0.91	0.93	1	0.96	175
13	0.99	0.98	0.99	1	0.97	0.98	0.99	1	0.99	1082
14	0.72	0.56	0.63	0.92	0.99	0.95	0.94	0.88	0.91	330
15	0.4	0.53	0.46	0.93	0.88	0.9	0.87	0.92	0.9	79
Accuracy			0.8			0.93			0.95	8762
Macro Avg	0.64	0.67	0.64	0.91	0.91	0.9	0.93	0.94	0.93	8762
Weighted Avg	0.81	0.8	0.8	0.94	0.93	0.93	0.95	0.95	0.95	8762

6 Conclusion and Future Work

Deep-learning methods have greatly impacted computer vision and proven to be a powerful technique for classification of remotely sensed images, adapting their performance to the different properties of HSI data. In this paper, we have developed a model which is able to achieve good training and testing accuracy with very a smaller number of training samples. The model is trained and tested on widely used dataset.

CNN-based design models have proven their ability for extracting highly discriminating features and efficiently using the spatial and semantic details present in hyperspectral image data cubes to be extremely efficient.

Cloud computing is an appealing future trend for investigation in which the higher computing specifications that DL-based hyperspectral image data processing actually involves can be fulfilled effectively, because image ratios as well as the size of future datasets are intended to be absolutely enormous, advocating for the execution of various DL-based models quicker and more efficiently on cloud.

Another essential element worth investigation in the future improvements is the invention of novel appropriate sampling techniques, which can prevent any duplication between the training samples and testing samples because of the patch size utilised by spatial techniques in the training phase.

References

1. Prasad S, Chanussot J (2020) Introduction. In: Prasad S, Chanussot J (eds) *Hyperspectral image analysis. Advances in computer vision and pattern recognition*. Springer, Cham. https://doi.org/10.1007/978-3-030-38617-7_1
2. Chang C-I (2007) *Hyperspectral data exploitation: theory and applications*. Wiley, Hoboken, NJ
3. Signoroni A, Savardi M, Baronio A, Benini S (2019) Deep learning meets hyperspectral image analysis: a multidisciplinary review. *J Imaging* 5(5):52. <https://doi.org/10.3390/jimaging5050052>
4. Li S, Song W, Fang L, Chen Y, Ghamisi P, Benediktsson JA (2019) Deep learning for hyperspectral image classification: an overview. *IEEE Trans Geosci Remote Sens* 57(9):6690–6709
5. Melgani F, Lorenzo B (2004) Classification of hyperspectral remote sensing images with support vector machines. *IEEE Trans Geosci Remote Sens* 42(8):1778–1790
6. Gualtieri JA, Chettri S (2000) Support vector machines for classification of hyperspectral data. In: *Proceedings of the IEEE geoscience and remote sensing symposium (IGARSS)*, Honolulu, HI, USA, 2000, pp 813–815
7. Zhuo L et al (2008) A genetic algorithm based wrapper feature selection method for classification of hyperspectral images using support vector machine. In: *Proceedings of the geoinformatics joint conference on GIS built environment classification remote sensing images*. International Society of Optics Photonics, Nov 2008, pp 71471J–71471J
8. Chen Y, Lin Z, Zhao X, Wang G, Gu Y (2014) Deep learning-based classification of hyperspectral data. *IEEE J Sel Topics Appl Earth Observ Remote Sens* 7(6):2094–2107
9. Paoletti ME, Haut JM, Plaza J, Plaza A (2019) Deep learning classifiers for hyperspectral imaging: a review. *ISPRS J Photogrammetry Remote Sensing* 158:279–317, ISSN 0924-2716. <https://doi.org/10.1016/j.isprsjprs.2019.09.006>

10. He X, Chen Y, Ghamisi P (2020) Heterogeneous transfer learning for hyperspectral image classification based on convolutional neural network. *IEEE Trans Geosci Remote Sens* 58(5):3246–3263. <https://doi.org/10.1109/TGRS.2019.2951445>
11. Li T, Zhang J, Zhang Y (2014) Classification of hyperspectral image based on deep belief networks. In: 2014 IEEE international conference on image processing (ICIP), pp 5132–5136, Oct 2014
12. Patel U, Dave H, Patel V (2021) Hyperspectral image classification using uncertainty and diversity based active learning. *Scalable Comput: Pract Exp* 22(3):283–293
13. Shekhawat SS, Sharma H, Kumar S, Nayyar A, Qureshi B (2021) bSSA: binary Salp swarm algorithm with hybrid data transformation for feature selection. *IEEE Access* 9:14867–14882
14. Patel U, Dave H, Patel V (2020) Hyperspectral image classification using semi-supervised learning with label propagation. In: 2020 IEEE India geoscience and remote sensing symposium (InGARSS). IEEE, New York
15. Baumgardner MF, Biehl LL, Landgrebe DA (2015) 220 band AVIRIS hyperspectral image data set: June 12, 1992 Indian pine test site 3, Sep 2015
16. Licciardi G, Marpu PR, Chanussot J, Benediktsson JA (2011) Linear versus nonlinear PCA for the classification of hyperspectral data based on the extended morphological profiles. *IEEE Geosci Remote Sens Lett* 9(3):447–451
17. Rizk Y, Hajj N, Mitri N, Awad M (2019) Deep belief networks and cortical algorithms: a comparative study for supervised classification. *Appl Comput Inform* 15(2):81–93

Design, Implementation and Performance Analysis of Shift Register Using Reversible Sayem Gate



Ruqaiya Khanam, Gitanjali Mehta , and Vinod Kumar Yadav

Abstract Reversible logic design in present scenario has gained importance because it releases minimum heat generation in the design circuit which prolongs the lifetime of the circuits. However, in irreversible design, they dissipated more heat due to a large number of internal switching. This paper is mainly focused on the design of the sequential circuits like shift register using reversible gates which is reducing the number of gate counts, garbage outputs and number of constant inputs. The proposed design of shift register is well designed using D-flip flops made by the reversible Sayem gate which will give the low output power and delay of the circuit design and compare the result of conventional master–slave D-flip flop with reversible D-flip flop. The output power and delay of the proposed reversible shift register is proved better in terms of power dissipation and delay by 33.99% and 24.53% respectively. The output of shift register obtained using Cadence Virtuoso tool with 180 nm technology is verified.

Keywords Reversible logic design · Reversible shift register · Sayem gate · D-flip flop

1 Introduction

Reversible logic circuits are famous for its design style. The application of reversible logic designs is for modern nanotechnology and quantum computing since it produces low heat generation as compared with existing irreversible approaches. The main objective of this design is to optimize the logic design style by reducing gate counts,

R. Khanam (✉)

Computer Science and Engineering, Center for Artificial Intelligence in Medicine, Imaging and Forensic, Sharda University, Greater Noida, India
e-mail: gitanjali.iitr@gmail.com

G. Mehta

Electrical and Electronics Engineering, Galgotias University, Greater Noida, India

V. K. Yadav

Electrical Engineering Department, Delhi Technological University, Delhi, India

constant inputs and garbage outputs to get a minimum power and delay of the design circuit. Using these design techniques, sequential and combinational logic circuits are developed which will produce low output power and delay of the circuits.

Krishna et al. [1] designed the reversible logic design like latches and flip flops which are highly improved in the matter of delay, the garbage outputs and quantum cost. Optimization of reversible sequential circuits in concern of reversible gates and garbage outputs is clearly shown and also the design of new reversible D-latch and D-flip flop which will give better performance in VLSI applications. All reversible logic gates are used to design any complex combinational circuits using hardware descriptive language. Simulation and Synthesis of all the reversible design is done by using Xilinx software which will calculate the power consumption and compare it with the irreversible Full adder [2]. Ali et al. [3] presented the reversible gates are utilized to design RS and D-flip flops since flip flops are the most important memory element in the digital circuits. The design is highly optimized in the connection of minimum number of gates and garbage outputs. Yelekar and Chiwande [4] approached to realize multipurpose binary reversible gates used in regular circuits realizing Boolean Functions. Jamal and Prasad [5] proposed 16 bits synchronous and asynchronous up/down counter using both reversible and irreversible logic gates. The reversible design of this circuit produced less power consumption comparing with irreversible design [5]. Mamum and Menville [6] show that the memory block of quantum devices is an important block for reversible latches. Using SAM gate, D and JK latches are designed. The main aspect of this design is the reduction in logic gates, garbage outputs and delays such that this approach will help to design all the reversible logic device [6]. All the primitive reversible gates which are collected from the literature reveals in [7]. Therefore, these logic gates help in designing complex computing circuits for low power CMOS, nanotechnology, cryptography, computer graphics, digital signal processing etc. The design of 4×4 reversible TSG forward and backward computation and design of Fredkin gate with four-bit carry skip adder block was proposed in [8] using 130 nm technology. Raghukanth et al. [9] give the complete description regarding the difference of reversible and conventional logic gates along with better results. The arithmetic operation like addition and subtraction was verified using DKG gate and its result was being compared with the conventional design. The design of fundamental building blocks of adder using Fredkin gate is implemented by dual line pass transistor logic. This design contains three CNOT gates and one Fredkin gates and the technology used is 0.35 μm CMOS technology [10]. Mamataj et al. [11] represent another approach to realize 4 bit 2's complement adder subtractor using these gates like Feynman gate, DKG gate and one control line and also proposed a new reversible gate DKFG. Babu et al. [12] compared the reversible and conventional gates to design the low power adder circuits. Although, DKG and TSG are used to realized addition and subtraction operations and results were compared with conventional gates. Kurian et al. [13] and Singh et al. [14] proposed a low power ALU [15] using reversible gates and also implemented on FPGA with reducing quantum cost and transistor cost. The propagation of carry bits was not considered in proposed carry save adder which gives better performance by 20% gate count and 17% quantum cost. Abbas alizadeh et al. [16] deals to maximize

4-bit reversible comparator which is based on the classical circuits using reversible logic gates, it reduces primary parameters such as the quantum cost, the garbage outputs and the constant inputs. The proposed comparator obtained four garbage outputs and one constant input.

Panchal and Nayak [17] presented a multiplier circuit with different reversible gates (Peres, Double Peres and Toffoli gates), for the minimization of the quantum cost, the garbage outputs and the constant input. Raj and Syamala [18] reveal the optimized design of the transistor level implementation of comparators using reversible circuits. The design of four-bit ripple carry adder/subtractor and eight-bit carry save adder are implemented. And also design CMOS GDI circuit which gives the optimal solution of combinational circuits [19]. Shukla et al. [20] presents a design of 4:1 multiplexer using reversible gates in terms of the number of gate inputs and outputs. Singla and Malik [21] proposed the programmable logic array design using reversible gates to get better performance of the design. MUX gate and three-inputs Feynman gate is used for the design to perform any reversible three input logic function. A 2-bit universal shift register has been designed in quantum-dot cellular automata (QCA) technology using fault tolerant circuits in such a way 4×1 multiplexer and D-flip flop [24]. To improve yield in fabrication, defect tolerance is an important feature of QCA systems with minimum clocking and area consumption.

The main feature of reversible logic circuits is low power consumption in digital design. Quantum computers are the most important devices which are designed by reversible circuits. Power losses in the circuit are one of the important parameters that is needed to concern deeply which will produce heat generation of the circuit thereby collapsing the devices. Therefore, this problem can be solved by only using reversible gates in the circuits. As concerns of the device's performance, first, it is required to design the circuit which has less power consumption with the help of reversible logic. Therefore, in this paper, the main target is to reduce power and delay to get better performance of the design using reversible logic gates.

Energy dissipation is also important concerns in the recent scenario due to the information loss in the circuits and system in irreversible design. Landauer showed in his work that heat of $KT \ln 2$ J is being generated in each lost bit, where K is the Boltzmann constant and T is the temperature [1, 22]. As per Moore's law, the transistor count will double in every 18 months [2]. The energy dissipation is depending on the erased bits during the computation of the circuit. Reversible logic circuits are those circuits that do not lose information even though, irreversible gates produced lots of heat due to high current in its devices and tend to decrease life-time of conventional circuits. However, in reversible logic operations will never remove the data and also dissipate limited or negligible heat comparing with irreversible logic. Thus, reversible logic is the only one to solve the problem of heat generation and increase the performance of the devices [23–25].

The array of inputs and outputs are equal in reversible logic circuits. It can generate a unique output pattern with the corresponding input set. While constructing the reversible logic circuits with the help of reversible gates, there will be some restriction to be kept.

- Fan out is not allowed
- Loops are not allowed

Several design constraints have been imposed in reversible logic circuits, hence there is a need to be either confirmed or optimized for implementing any specific Boolean functions. Both inputs and outputs must be equal in number.

- Each input has a distinct output pattern
- Output will be used only once
- As a result, the circuit must be acyclic

The reversible logic circuit has one of the prominent logic designs for the purpose to obtain the best performance. The different reversible logic design is considered for this work to know the reversibility nature of the different circuit. One of the challenges in designing the circuit is power consumption and delay which will slow down the performance of the circuit. Hence the main focus in reversible logic design is, the reduction in gate counts, constant inputs, quantum cost and garbage outputs.

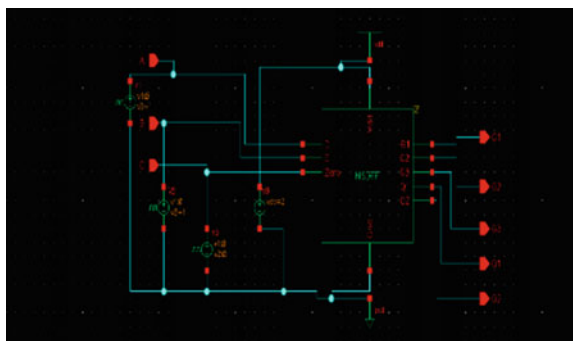
2 Proposed Architecture of Shift Register

2.1 Conventional MSDFF

In this conventional master–slave (MS) D-flip flop with reversible Fredkin and Feynman gate, one latch works as a master while second latch is used as slave in a circuit. This design style helps to optimize the circuit performance. Slave latch was needed for the inverted clock in all available designs but in this case, there is no need an inverted clock for latch act as a slave. Since the negative enable D-latch is used as slave, clock inversion is not required here. Fredkin and Feynman gates are used to design MS D-flip flop (Fig. 1). Design of MS D-flip flop is given below.

The internal design of MS D-flip flop has Fredkin and Feynman gate which is used as a function of the master and slave circuit. It produced garbage output which is three

Fig. 1 Master slave D flip flop



in the circuit. The input of the circuit is enabled signal, D input and constant input zero. The circuit diagram is given in Fig. 2. A Fredkin gate is a 3×3 conservative reversible gate with one-to-one mapping of “ $W=A$ ”, “ $X=A'B+AC$ ”, “ $Y=AB+A'C$ ” where A, B and C are inputs and P, Q and R are outputs. The internal diagram of the Fredkin gate is shown in Fig. 3. It is a 2×2 reversible gate with quantum cost of 1 which maps inputs A and B to output “ $W=A$ ” and “ $X=AB$ ”. The internal diagram is illustrated in Fig. 4.

Fig. 2 Internal diagram of master slave D FF

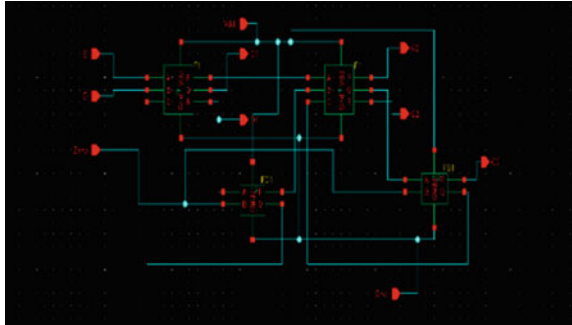


Fig. 3 Internal diagram of Fredkin gate

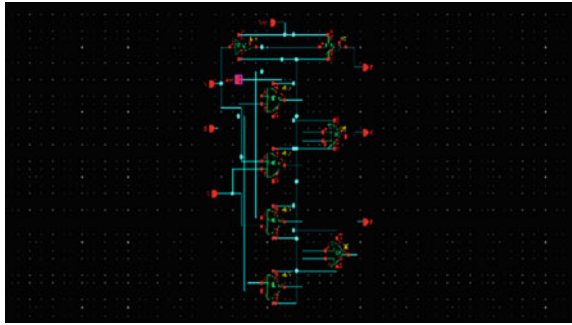


Fig. 4 Internal diagram of Feynman gate

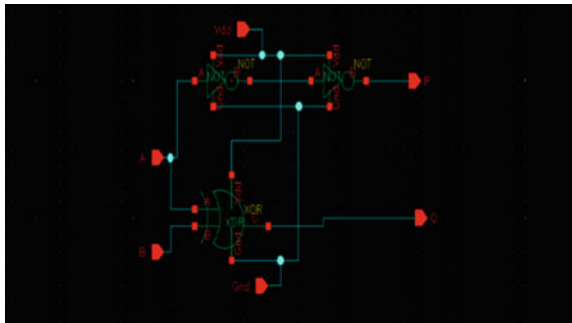


Fig. 5 D flip flop using Sayem gate

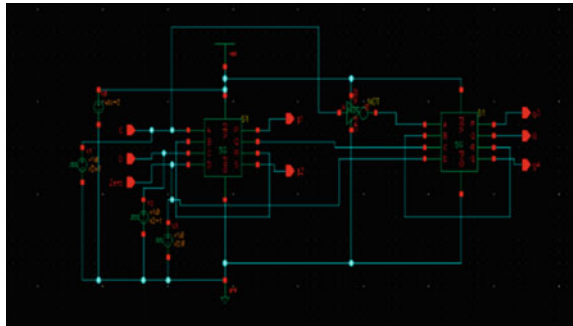
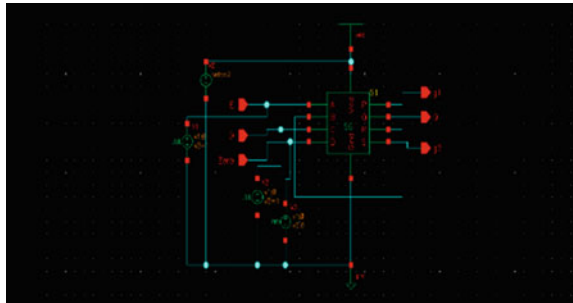


Fig. 6 Design of D latch using SG



2.2 DFF Using Sayem Gate

Design of D-flip flop is compared with the conventional design of master–slave D-flip flop which produces less output power and delay. The characteristics equation of D-latch is “ $Q=DE+E'Q$ ” which is realized with one Sayem gate with the input of E, Q, D and 0 (refer Fig. 5). The design of D latch is given in Fig. 6. Sayem gate is 4×4 reversible gate where A, B, C, D are the inputs and the corresponding outputs are “ $P=A$ ”, “ $Q=A'B \text{ XOR } AC$ ”, “ $R=A'B \text{ XOR } AC \text{ XOR } D$ ” and “ $S=AB \text{ XOR } A'C \text{ XOR } D$ ”. It can be used as a universal gate with two inputs and also performs any two input Boolean function. Sayem gate design is as shown in Fig. 7. Internal circuit of the Sayem gate has three XOR gate, four AND gate, two buffers and lastly one Feynman gate. Figure 8 demonstrates its internal circuit which comprises the logic gates.

3 Reversible Shift Register Design

Shift register is an array of flip flops which is connecting each other and also sharing the same clock. As a result, the output of any shift register that each bit shifts by one position. But in case of the reversible shift register it is designed using D-latch

Fig. 7 Design of Sayem gate

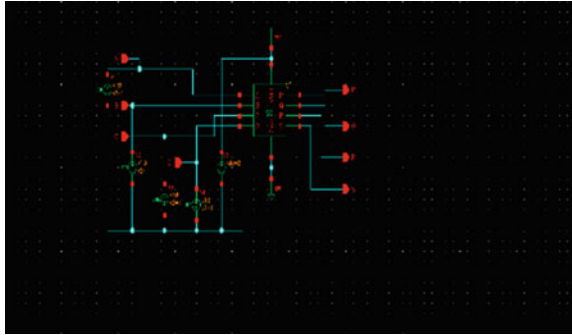
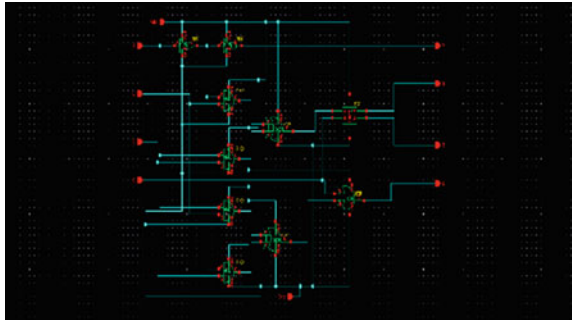


Fig. 8 Internal design of Sayem gate



which is formed by reversible Sayem gate that has produced less delay and power to improve the performance of the design. Shift register is classified into four types.

- Serial In Serial Out (SISO)
- Serial In Parallel Out (SIPO)
- Parallel In Serial Out (PISO)
- Parallel In Parallel Out (PIPO)

Four types of shift registers are designed with reversible Sayem gate to reduce the delay and power of the circuit as compared with the conventional one. Although the design of the circuit is easy by reducing the number of gate counts and constant inputs. So that the performance of the design will be improved which will be used in future needs. The different types of the shift register are designed and the internal circuits are given in Figs. 9, 10, 11 and 12 respectively.

4 Simulation Results

Simulation tool, simulation parameters, analysis of newly offered design along with simulation results are given in this section. Furthermore, comparative analysis of

Fig. 9 Design of SISO

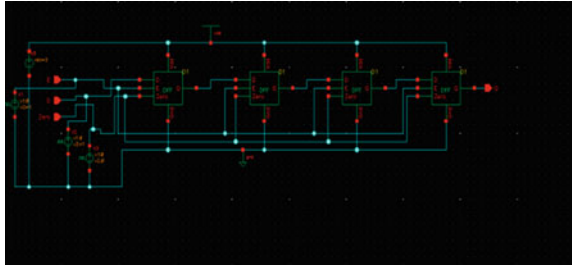


Fig. 10 Design of SIPO

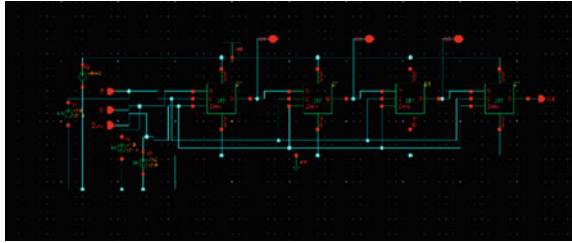


Fig. 11 Design of PISO

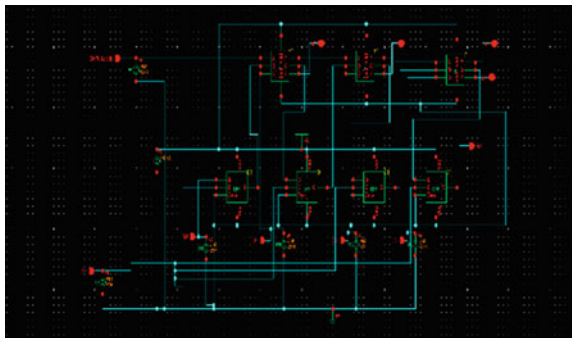
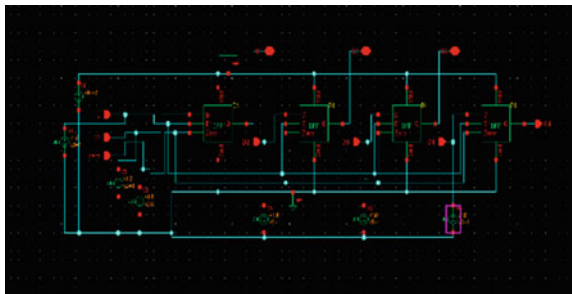


Fig. 12 Design of PIPO



reversible and conventional logic gates are also discussed in the below section. Graphical presentations are shown in the below figures (Figs. 13, 14, 15, 16, 17, 18, 19, 20, 21 and 22) with inputs and outputs of signals of all respective gates. The results are obtained from the Cadence Virtuoso tool which is a precise and simulator. Figures 13, 14, 15, 16, 17 and 18 show the result of conventional MSDFF, Fredkin gate, Feynman gate, DFF using Sayem gate, Sayem gate and D-latch using Sayem gate respectively. Figures 19, 20, 21 and 22 shows the result for SISO, SIPO, PISO and PIPO respectively.

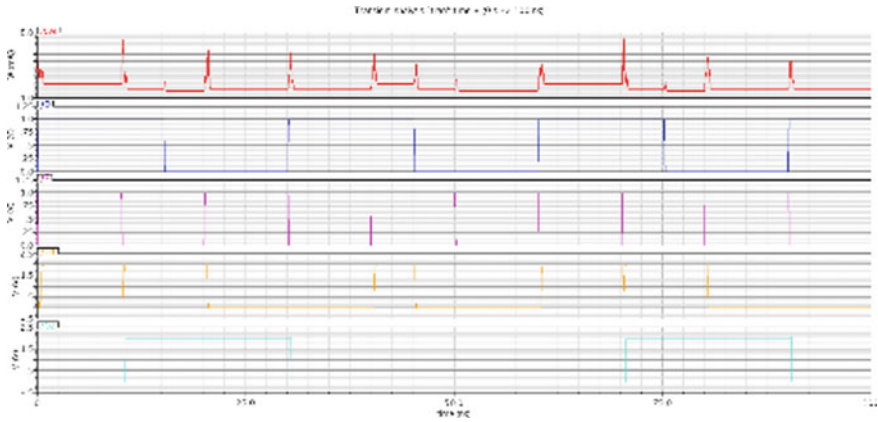


Fig. 13 Result of conventional MSDFF

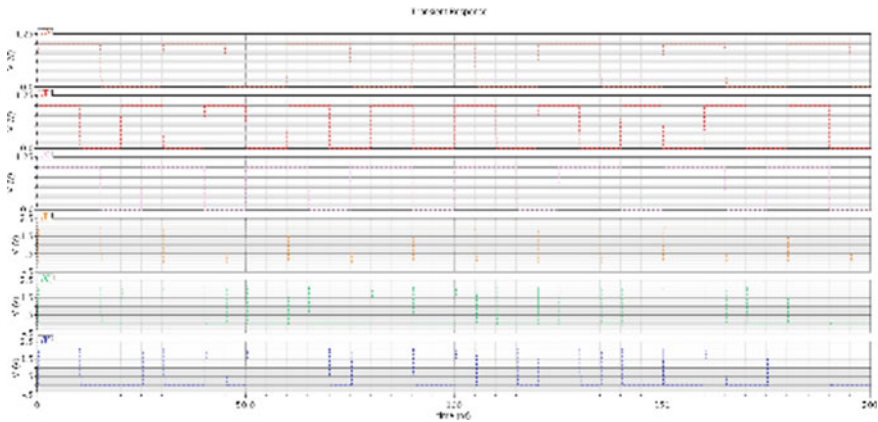


Fig. 14 Result of Fredkin gate

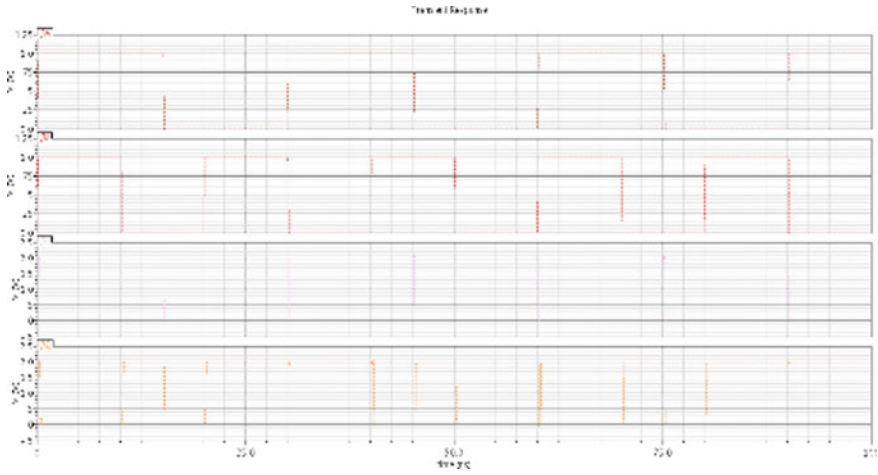


Fig. 15 Result of Feynman gate

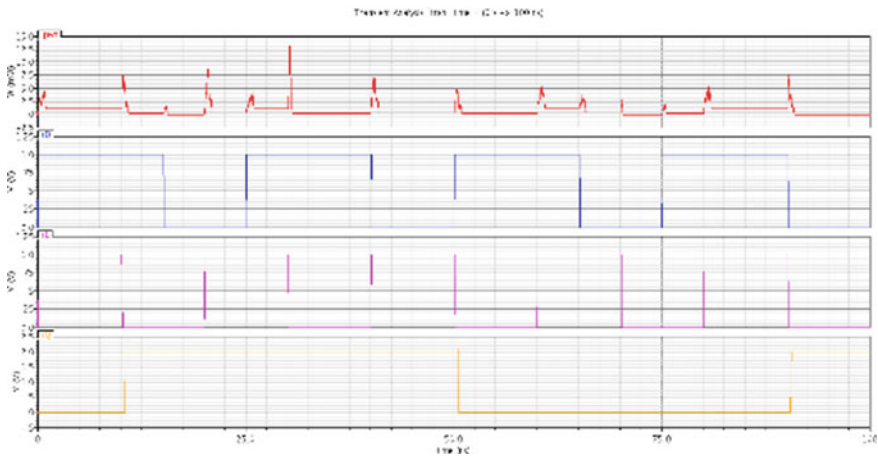


Fig. 16 Result of DFF using Sayem gate

5 Experimental Results and Discussion

The different output result based upon the proposed design of sequential circuit which is designed by the cadence virtuoso tools is analyzed in Table 1. The proposed design reduces gate counts and constant inputs of the reversible gates. It will show the performance of the design comparing with the conventional design of the sequential circuits. The output power and delay is shown clearly that will describe the improvement of the design. The table of the power and delay of different circuits is given below. The verification of the proposed design is done by showing the different

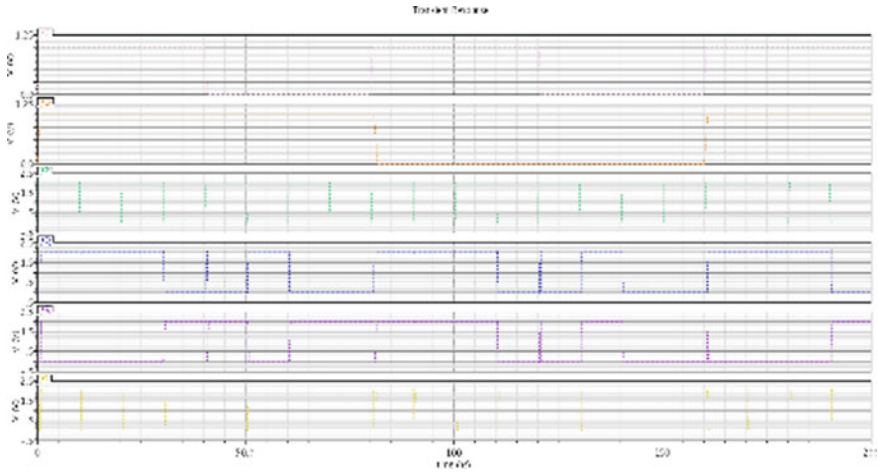


Fig. 17 Result of Sayem gate

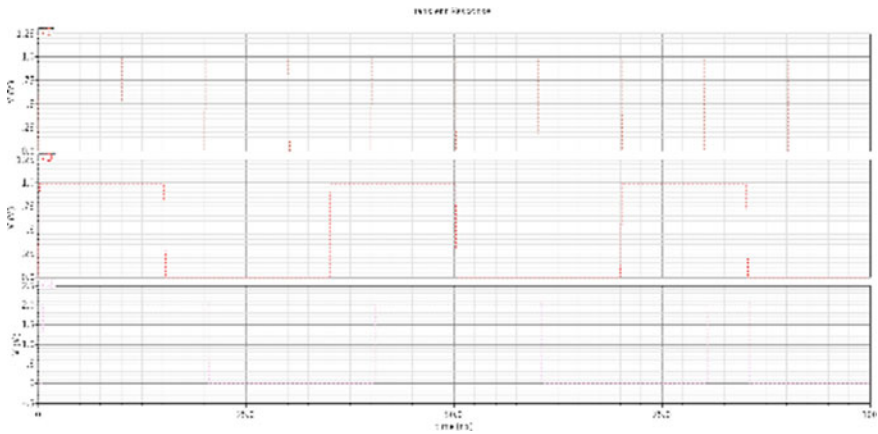


Fig. 18 Result of D latch using SG

obtained results using Cadence virtuoso tools. Power and delay are obtained from proposed design with reduction of the gate count, the constant inputs and the garbage outputs thereby increasing the design performance.

The proposed sequential circuit that is shift register used reversible Sayem gate which is a well design that has less number of gate counts, constant inputs and garbage output. So that the output power and delay is getting reduced as compared with the conventional design to improve the performance of the proposed design. Thus, the proposed design is well developed and verified to increase the desire performance of the design circuit so that the lifetime of the circuits also improved. Table 2 gives the

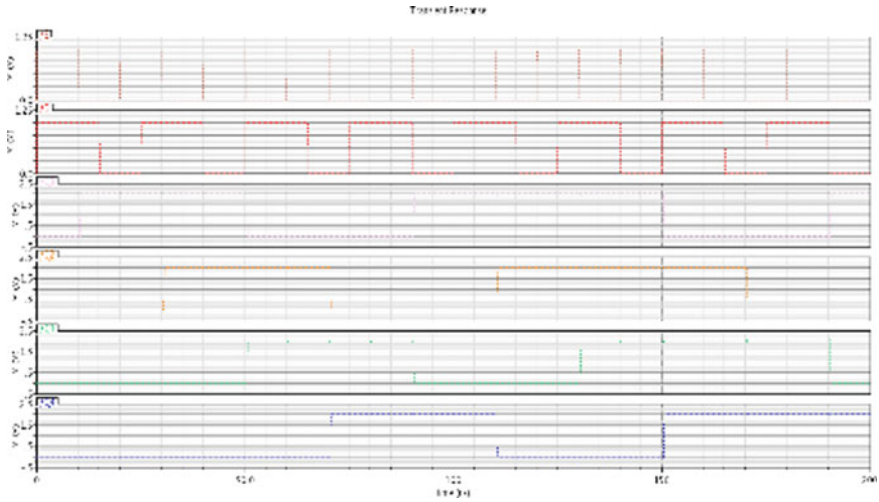


Fig. 19 Result of SISO

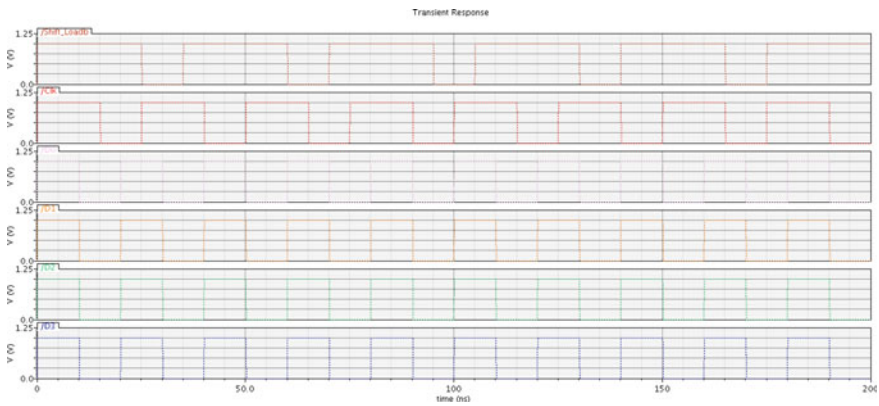


Fig. 20 Result of SIPO

details of different circuits regarding gate counts, the garbage output and a constant input.

6 Conclusions

This paper proposes designing of sequential circuits that is mainly the shift registers are designed using the reversible Sayem gate. Moreover, the Sayem gate is used for designing the basic internal circuit that is D-flip flop and D-latch. Internal design

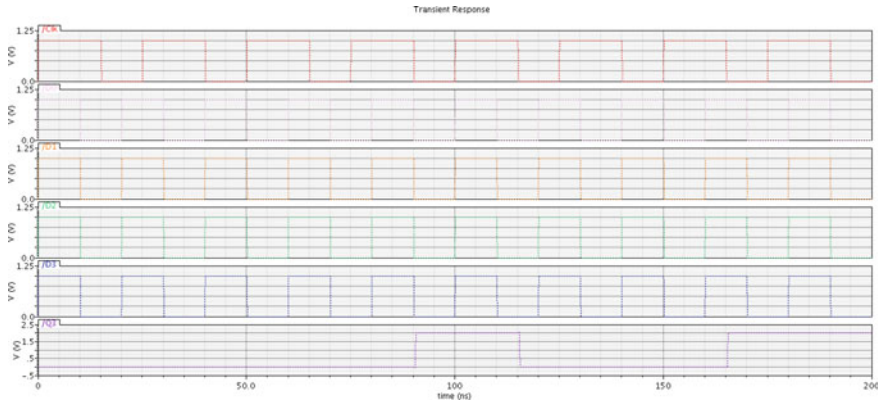


Fig. 21 Result of PISO

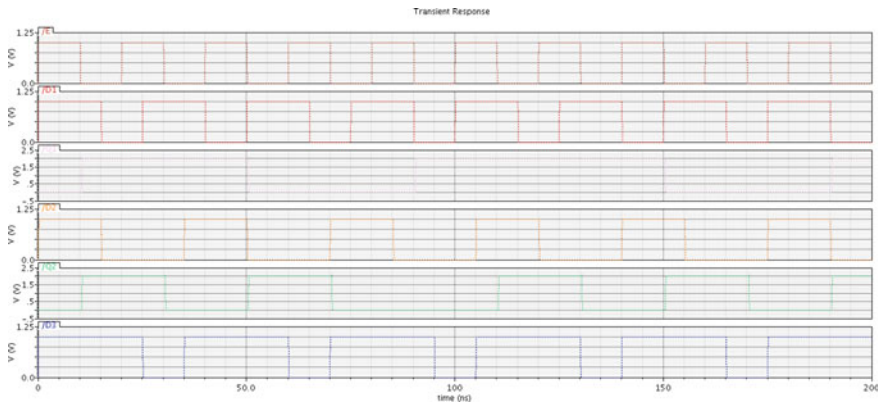


Fig. 22 Result of PIPO

Table 1 Output power and delay of different circuits

Design circuits	Output power	Delay (Picosecond)
DFF using SG	497.3 μ W	338.7
MSDFF (Conventional)	753.4 μ W	421.8
SISO	2.221 mW	338.8
SIPO	2.221 mW	330.6
PISO	2.91 mW	338.7
PIPO	3.1 mW	340.5

Table 2 Number of gate counts, garbage output and constant input

Design circuits	Gate counts	Garbage output	Constant input
DFF using SG	10	3	1
MSDFF	20	4	1
SISO	21	16	1
SIPO	21	16	1
PISO	38	20	1
PIPO	24	16	1

of reversible circuits of D-flip flop/D-latch based on CMOS circuit, all the shift registers were designed in a unique manner which is giving a high performance of the design by lowering the power and delay of the design. In this paper, it is clearly shown the comparing of reversible design of D-flip flop and conventional design of Master–Slave D-flip flop. Hence, it will show the improved output power and delay of the circuit. According to the design of shift registers have been completed by looking at the more effective results of both flip flops (D and MSD). The design is mainly concerned about the reduction in gate counts, the constant inputs and the garbage outputs. Design and simulations were done with 180 nm technology of Cadence Virtuoso Tools. The output power and delay of the proposed design is getting efficiently reduced as compared to the conventional design of the sequential circuits. Also, the gate counts, the constant inputs and the garbage output are also minimized. But the design is not mainly focused on the quantum cost of the reversible design. Hence there is future scope of getting the better performance of the design when the quantum cost of the design will be considered.

References

1. Krishna P, Swaroop V, Harika S, Sravya N, Vani V (2014) Implementation of optimized reversible sequential and combinational circuits for VLSI applications. *Int J Eng Res Appl* 4(4):382–438
2. Goyal D, Sharma V (2012) VHDL implementation of reversible logic gates. *Int J Adv Technol Eng Res* 2(3):157–163
3. Ali B, Hossin M, Ullah E (2011) Design of reversible sequential circuit using reversible logic synthesis. *Int J VLSI Des Commun Syst* 2(4):37–45
4. Yelekar P, Chiwande S (2011) Introduction to reversible logic gates and its applications. *Int J Comput Appl*, pp 5–9
5. Jamal A, Prasad J (2014) Design of low power counters using reversible logic. *Int J Innov Res Sci Eng Technol* 3(5):12792–12799
6. Mamum S, Menville D (2013) Quantum cost optimization of reversible sequential circuits. *Int J Adv Comput Sci Appl* 4(12):15–21
7. Mamataj S, Saha D, Banu N (2014) A review of reversible gates and its application in logic design. *Am J Eng Res* 3(4):151–161
8. Vijeta B, Hemalatha K (2013) Novel reversible gate based circuit design and simulation using deep submicron technologies. *Int J Emerg Trends Technol Comput Sci* 2(5):3–7

9. Raghukanth B, Krishna M, Sridhar M, Swaroop V (2012) A distinguish between reversible and conventional logic gates. *Int J Eng Res Appl* 2(2):148–151
10. Rentergem Y, Vos A (2005) Optimal design of a reversible full adder. *Int J Unconv Comput* 1:339–355
11. Mamataj S, Das B, Rahaman A (2012) An approach for realization of 2's complement adder subtractor using DKG reversible gate. *Int J Emerg Technol Adv Eng* 3(12):205–209
12. Babu T, Sounder D, Rao B, Babu P (2012) A low power adder using reversible logic gates. *Int J Res Eng Technol* 1(3):244–247
13. Kurian J, Alex L, Venugopal G (2013) Design and FPGA implementation of a low power arithmetic logic unit. *IOSR J VLSI Sign Process* 2(3):57–61
14. Singh R, Upadhyay S, Saranya S, Jagannath K, Hariprasad S (2014) Efficient design of arithmetic logic unit using reversible logic gates. *Int J Adv Res Comput Eng Technol* 3(4)
15. Khanam R, Mehta G, Astya R (2021) Design and implementation of ALU-based FIR filter. In: *Smart computing: proceedings of the 1st international conference on smart machine intelligence and real-time computing*. CRC Press, Taylor and Francis, pp 702–707
16. Abbasalizadeh S, Forouzanadeh B, Aghababa H (2013) 4-bit comparator design based on reversible logic gates. *Lect Notes Inform Theory* 1(3):86–88
17. Panchal V, Nayak V (2014) Analysis of multiplier circuit using reversible logic. *Int J Innov Res Sci Technol* 1(6):279–284
18. Raj K, Syamala Y (2014) Transistor level implementation of digital reversible circuits. *Int J VLSI Des Commun Syst* 5(6):44–61
19. Basha M, Kumar V (2012) Transistor implementation of reversible comparator circuit using low power techniques. *Int J Comput Sci Inform Technol* 3(3):4447–4452
20. Shukla V, Singh O, Mishra G, Tiwari R (2013) Novel design of a 4:1 multiplexer circuit using reversible logic. *Int J Comput Eng Res* 3(10):30–35
21. Singla P, Malik N (2012) A cost-effective design of reversible programmable logic array. *Int J Comput Appl* 41(15):42–46
22. Frank M (2017) Foundations of generalized reversible computing. In: *Reversible computation. Lecture notes in computer science*, vol 10301, pp 19–34
23. Frank M ((2017)) Asynchronous ballistic reversible computing. In: *IEEE international conference on rebooting computing (ICRC)*
24. Prakash G, Darbandi M, Gafer N, Jaberullah N, Jalali R (2019) A new design of 2-bit universal shift register using rotated majority gate based on quantum-dot cellular automata technology. *Int J Theor Phys*
25. Rajeshwari M, Hongal R, Shettar R (2018) Design and Implementation of 8-bit shift register using reversible logic. In: *IEEE international current trends toward converging technologies*, Coimbatore

Automated Oxygen Blender for Regulation of Oxygen Saturation in Hypoxia Patient



Samruddhi Anikhindi , Shreyas Patil , and Pauroosh Kaushal 

Abstract Hypoxia is a condition wherein the oxygen saturation (SpO_2) level in the tissue drops below 90%, and this may cause improper functioning of the human body. An oxygen blender is a medical device which mixes medical grade oxygen and medical air in a variable ratio to restore the normal oxygen saturation levels in hypoxic patients. Currently, oxygen blender is operated manually based on clinician's experience that could lead to exposure to higher or lower levels of oxygen than required. Automating this process will ensure optimal delivery of oxygen to patient leading to better recovery. In this work, we propose an automated oxygen blender in closed-loop system to automatically sense SpO_2 and provide optimal oxygen flow rate to achieve and maintain the desired oxygen saturation levels. The robustness and transient response of the system is improved by designing proportional integral derivative controller using direct synthesis and root locus method. The controller actuates oxygen blender to provide fraction of inspired air (FiO_2) within practical range from 0.21 to 0.4. The feedback control system was simulated at different oxygen saturation levels on Simulink. The results indicate that the controller is able to achieve the desired oxygen saturation levels with reduced steady-state error, peak overshoot, and settling time. The designed PID has steady-state error of 0.02 and 0.03, settling time of 103.82, and 139.26 s and overshoot of 0.0027 and 1.28% using direct synthesis and root locus approach, respectively.

Keywords Oxygen blender · Oxygen saturation · Controller · Closed-loop · Simulation

S. Anikhindi (✉) · S. Patil · P. Kaushal
MIT School of Bioengineering Sciences and Research, MIT Art, Design and Technology
University, Pune 412201, India
e-mail: samruddhianikhindi@gmail.com

P. Kaushal
e-mail: pauroosh.kaushal@mituniversity.edu.in

1 Introduction

Hypoxemia refers to low oxygen saturation levels in the blood which thereby leads to hypoxia wherein the oxygen levels in the tissues are reduced. These conditions could lead to further problems in the human body which could be mild such as shortness of breath and headache, or severe such as interference with the functioning of brain and heart. To ensure the accurate delivery of oxygen to such patients so that normal oxygen saturation levels are achieved, oxygen blenders are used. Oxygen blender mixes medical grade oxygen and medical air with oxygen concentration ranging from 21 to 100% according to the patient's requirements. Currently, the FiO_2 output of the blender is controlled manually by medical personnel based on their experience. This is inefficient as there could be human errors involved, thereby causing delivery of more or less oxygen than what is required leading to worsening the patient condition. If the blender is mechanically controlled, the time required to reach the desired oxygen saturation will be more and it demands continuous attention of the medical personnel. To overcome the mentioned issues, a closed-loop feedback system has been reported in the literature.

Tehrani et al. [1] proposed a closed-loop system for automatic control of FiO_2 in mechanical ventilation. The patient was represented with a mathematical model. The plant is composed of lungs, body tissues and the brain tissue, an arterial transport delay, and controllers of cardiac output and cerebral blood flow. Two closed-loop control mechanisms, one being a rapid stepwise control system that instantly responds to fast declines in arterial saturation of the patient and the second being a slower PID control system that controls inspired fraction of oxygen when there is no sharp decline in the arterial saturation, were combined to form this system. In a work by Iobbi et al. [2] a closed-loop control scheme was proposed, which used feedback from the oximeter to maintain a target SpO_2 of 91% by regulating the oxygen flow rate to the patient. The patient arterial oxygen concentration (PaO_2) is modeled by the function which has three different components: a baseline level (P_0), an arbitrary disturbance (P_d), and an oxygen flow rate dependence (P_f). Their sum represents PaO_2 of the patient. The P_d introduces fluctuations from the baseline level and simulates the variable conditions possible in COPD patients. With the help of the controller, the time arterial blood saturation level spent below the threshold was reduced by 76%. Given the same volume of oxygen, the closed-loop controller also showed an improvement of 63% compared to the fixed flow rate long-term oxygen therapy (LTOT). In this work, the model parameters were selected based on average values found in the literature and these values may differ among different patients. Though positive results were obtained in the work, there is no evidence of the robustness of the closed-loop method. Alkurawy [3] proposed a controller for FiO_2 to regulate SpO_2 in neonatal infants. Several controllers were designed and tested for performance. The respiratory model was based on earlier research completed by Yu [4]. It is a nonlinear model that describes the relationship between SpO_2 and FiO_2 . The heart rate and respiratory rate were taken as disturbances. In the PI digital controller, a good response was obtained for output without zero steady state, and the

minimum settling time was 180, while for PID, the minimum settling time at 170 s. However, an abnormal condition such as hypoxia has not been implemented in this work. Yu et al. [5] developed a PI controller and a multiple model adaptive controller (MMAC) for regulating arterial oxygen saturation levels in mechanically ventilated animals. The patient model was represented by a first-order system, and time delay and simulations were carried out. The designed controllers were tested, and a comparison was made. The MMAC was found to be better in terms of better regulating SpO_2 even in the presence of noise and disturbances. Morozoff and Evans [6] developed a microprocessor-based device to control SpO_2 in neonates by adjusting FiO_2 delivered by the blender. The controller attempts to maintain the target SpO_2 set by the user. An alarm is activated in case there is SpO_2 and FiO_2 sensor disconnection, blender disconnection, and SpO_2 limiting errors. It is observed from the literature that a closed-loop system with optimal design of controller can be used to develop automated oxygen blender to regulate oxygen saturation of a patient. Mollazadeh-Moghaddam et al. [7] designed a blending device based on Venturi principle. The blender is made up of three parts: a nozzle, an air entrainment window, and an orifice. This work illustrates the viability of developing a low-cost air-oxygen blender that does not require electricity or compressed air and can give precise oxygen concentrations to newborns in respiratory distress. Islam et al. [8] attempted to provide an overview of the current technologies that are regularly used to assist infected patients in breathing. The review discussed the most recently produced breathing aid technologies, such as oxygen therapy devices, ventilators, and CPAP. For the right selection of inexpensive technologies, a comparative study of the developed devices is also given, together with essential difficulties and probable future directions.

This work aims to develop a negative feedback-based self-regulating oxygen blender to maintain oxygen saturation at desired level. The oxygen saturation of the patient is feedback and compared to the desired oxygen saturation level. A mathematical model is used to approximate the SpO_2 level of patient. The error is calculated and fed to the PID controller which then actuates the oxygen blender to provide the FiO_2 within the range of 0.21–0.4, which thereby changes the flow rate of oxygen. The PID controller is tuned using direct synthesis and root locus method, and its performance is evaluated on an oxygen saturation model. It is observed that the controller is able to achieve the oxygen saturation level with improved transient and steady-state response. Further, the change in FiO_2 indicates the patient is not exposed to undesirable levels of oxygen.

2 Methodology

2.1 Oxygen Saturation Model

The oxygen saturation model used in this work takes reference from prior research by Yu and Alkurawy [3]. The equation used for neonatal oxygen saturation model

that predicts the oxygen saturation levels of the infant patient is given as:

$$\frac{SpO_2(s)}{FiO_2(s)} = \frac{G_p}{\tau s + 1} \tag{1}$$

where $G_p = \frac{(1 - x_d)V_I'}{8.63(1 - y_s)Q\beta_c + (1 - x_d)V_I'} \frac{(1 - y_s)}{\beta_a} (P_B - P_{H_2O})$ (2)

$$\tau = \frac{V_A}{V_A' + 8.63Q_p\beta_c} \tag{3}$$

where SpO₂ is the oxygen saturation level given by the pulse oximeter readings, FiO₂ is the fraction of inspired oxygen, G_p is the process gain indicative of how much the SpO₂ changes on changing the FiO₂ and τ is the time lag signifying how fast the desired SpO₂ level is achieved. In this work, the model has been modified to incorporate transport lag and the pulse oximeter delay. A time delay of 34.8 s, which accounts for 19.8 s lag for the oxygen saturation from the lungs to be reflected in the tissues and 15 s pulse oximeter delay, has been added to τ to obtain effective time lag or E_τ. The modified oxygen saturation model is given as:

$$\frac{SpO_2(s)}{FiO_2(s)} = \frac{G_p}{E_\tau s + 1} \tag{4}$$

The derived model is implemented for neonates with normal neonate parameter values as tabulated in Table 1. The values provided in the table are used to implement oxygen saturation model. The values of β_a and β_c are changed to simulate the hypoxia conditions.

The values for β_a and β_c are derived from the graph showing the derivative of oxygen dissociation curve plotted using the following equation [3]:

Table 1 Parameter values for normal neonates

Parameter	Description	Normal value
V _A	Ventilated part of lung is perfused with blood	0.5 L
V _A '	Respiratory rate of the V _A section of lung model	4 L/min
Q	Total blood flow to respiratory system	5 L/min
x _d	Dead space ratio	5%
y _s	Shunt ratio	5%
P _B	Barometric pressure	760 torr
P _{H₂O}	Water vapor pressure	47 torr
β _a	Apparent solubility of oxygen in arterial blood	0.017596
β _c	Apparent solubility of oxygen in capillary	0.013278
V _I '	Total respiratory rate	4.2 L/min
Q _p	Pulmonary blood flow	10 cm/s

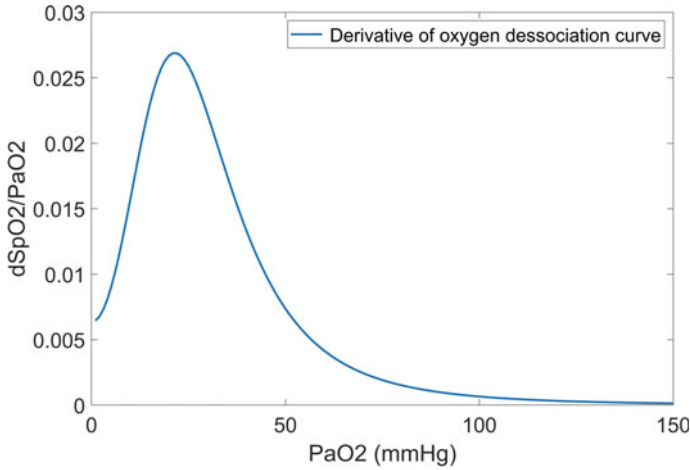


Fig. 1 Graph of derivative of oxygen dissociation curve

$$\frac{dSpO_2}{dPaO_2} = \frac{23,400(3PaO_2^2 + 150)}{(PaO_2^3 + 150PaO_2)^2 \left(\frac{23,400}{PaO_2^2 + 150PaO_2} + 1 \right)^2} \tag{5}$$

where PaO₂ represents the partial pressure of oxygen. By keeping the range of PaO₂ between 0 and 150 mm Hg, we plotted a graph of PaO₂ vs the derivative dSpO₂/d PaO₂ as shown in Fig. 1.

The partial pressure of oxygen in capillary is about 40 mm Hg, and at this value of x-axis, the y-axis value gives β_c that equals 0.013278. The partial pressure of oxygen in finger tissue from where the pulse oximeter readings are taken is approximated to be 35 mm Hg. The y-axis value for this gives β_a that equals 0.017596. This value of β_a holds true only for a normal human who does not have conditions like hypoxia. In hypoxia, the oxygen saturation levels of tissue are reduced. Hypoxia condition can be mimicked by changing the apparent solubility of oxygen in arterial blood (β_a), and this value is taken as 0.0189.

2.2 Closed-loop Feedback System

The closed-loop system consists of a blender, oxygen saturation level model of the patient and a negative feedback loop. Figure 2 shows the block diagram of the closed-loop feedback system. As shown in the figure, negative feedback sensing actual SpO₂ level of the patient is given to comparator. The comparator computes error between the set point of 95 with the actual SpO₂ of the patient and fed it as an input to PID controller. The PID controller actuates the oxygen blender system which further changes the SpO₂ level of the patient.

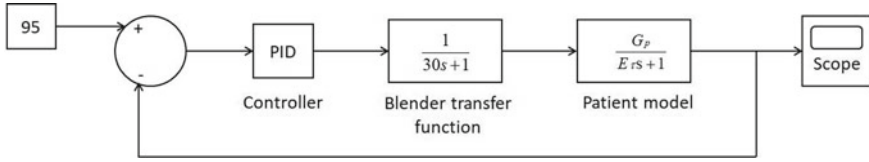


Fig. 2 Closed-loop negative feedback control system

The blender has a mixing time constant of 30 s. It is represented by the following transfer function:

$$\frac{1}{30s + 1} \tag{6}$$

Controller

The controller compares the actual oxygen saturation value with the reference input (set point), determines the deviation, and produces a control signal that reduces the deviation to zero or a small value. Here, a proportional integral derivative (PID) controller is designed using two methods: direct synthesis method and root locus method. The transfer function of the PID is given by:

$$K_p + \frac{K_I}{s} + K_D s \tag{7}$$

where K_p is the proportional gain, K_I is the integral gain, and K_D is the derivative gain.

Direct synthesis method

In this method, the desired closed-loop transfer function is assumed to be a first-order system, represented as

$$g_{CL}(s) = \frac{1}{\lambda s + 1} \tag{8}$$

$$g_c(s) = \frac{1}{g_p(s)\lambda s} \tag{9}$$

Here $g_p(s)$ is the oxygen saturation model and $g_c(s)$ is the controller transfer function. Substituting Eq. (4) in the above equation, we get:

$$g_{CL}(s) = \frac{1046s^2 + 64.87s + 1}{420.3\lambda s} \tag{10}$$

Different values of λ were taken, and the controller parameters were found for these different values. The optimal value was found to be 2 at which no oscillations or overshoot was obtained.

Root locus method

The natural frequency ω_n was found using the time constant obtained in direct synthesis method by taking λ as 2. The damping ratio ζ is taken as 0.9 so that the system is robust and $\omega_n = 0.0396$. Using these values of ζ and ω_n , the desired poles were found

$$s = -\omega_n\zeta \pm j\omega_n\sqrt{1 - \zeta^2} \tag{11}$$

$$|s| = 0.0395$$

$|G(j\omega)|$ and $\angle G(j\omega)$ when $s = -0.03564 \pm 0.01726j$ were determined. The angle made by the plant with desired pole (ϕ), and the angle formed by the desired pole with the origin (β) were also calculated. All these values were taken to tune the PID [9].

3 Simulation

3.1 Validation of Oxygen Saturation Model

The oxygen saturation model is validated for hypoxia and normal conditions. The normal oxygen saturation level of an infant is in the range of 95–100% by taking $\beta_a = 0.017596$. In case of hypoxia, the oxygen saturation level is considered to be 88% by taking $\beta_a = 0.0189$. The model is simulated and is shown in Fig. 3.

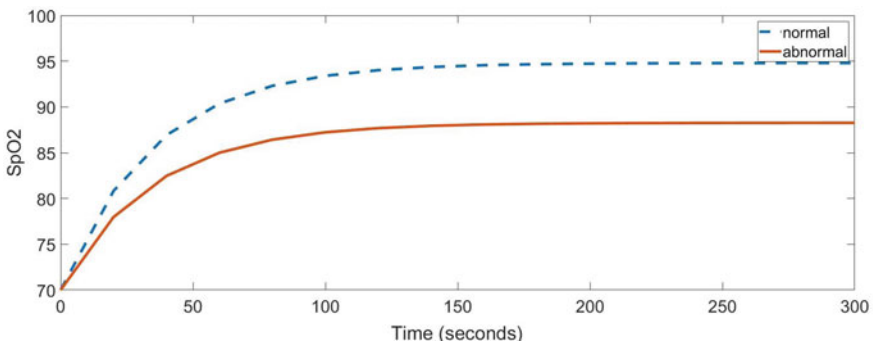


Fig. 3 SpO₂ response in normal and hypoxia conditions

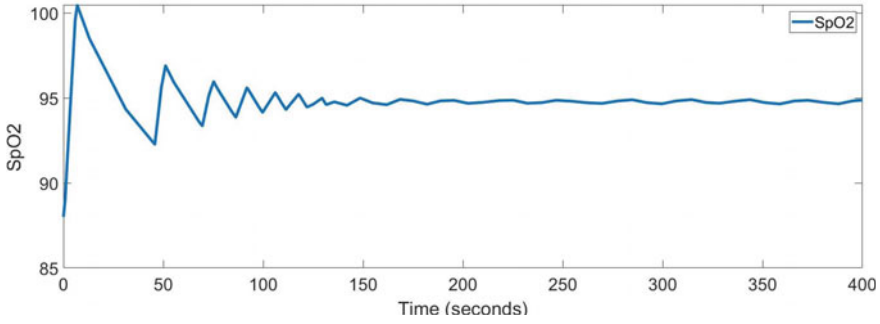


Fig. 4 SpO₂ response of closed-loop system without controller

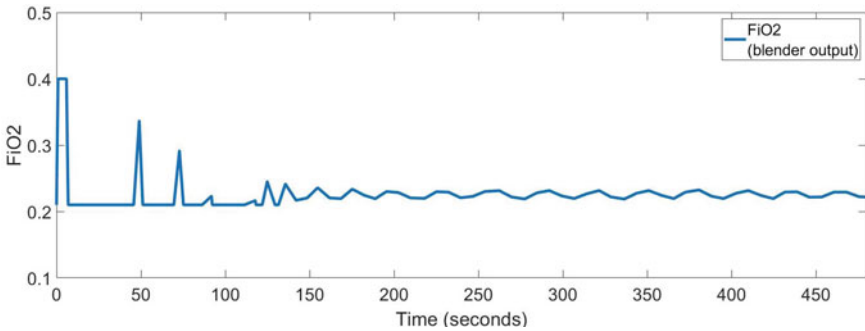


Fig. 5 FiO₂ output of the blender in closed-loop system without controller

In the figure, the response of normal infant settles at a SpO₂ of around 95% while the response of hypoxic infant settles at a SpO₂ of 88%. An oxygen blender is required to provide an increased oxygen flow rate by increasing the FiO₂ and thereby maintaining a desired SpO₂ level.

3.2 Closed-loop System

In order to achieve desired SpO₂, manual blenders are currently used. A closed-loop system will automatically regulate the oxygen flow rate to achieve the set point value of oxygen saturation. Initially, the closed-loop system is implemented which consists of the oxygen blender and oxygen saturation model of the patient in a negative feedback loop. Figures 4 and 5 represent the SpO₂ response and FiO₂ level obtained for the closed-loop system.

As can be seen, the response is undesirable. The limitations of this system are:

- (1) It is highly oscillatory which means the SpO₂ levels of the patient are changing very quickly from very high to very low values. This could result in worsening of the condition of the patient. Supplying oxygen more than the required value will cause the SpO₂ to increase to toxic levels, leading to hyperoxia and supplying oxygen lesser than the required value will cause the SpO₂ of the patient to reduce, leading to hypoxia conditions.
- (2) The time required for the SpO₂ levels to stabilize is around 200–250 s. This is a lot of time considering the patient could be in critical condition and requires immediate stabilization of the oxygen saturation level of the patient to the normal values.

To eliminate these shortcomings, we design a PID controller in the forward path of the closed-loop system. The controller uses oxygen saturation level of the patient and regulates the oxygen flow rate to bring it to the set point value. Due to the control action, an improvement in steady state and transient response can be observed.

4 Results and Discussion

The oxygen saturation model is validated for both hypoxia and normal conditions. Normal condition is mimicked by taking β_a value such that the oxygen saturation model gives a response settling at 88%. Hypoxia condition is mimicked by taking β such that the oxygen saturation model gives a response of 95%. Through the simulations, it is seen that the closed-loop system can be used for autoregulation of the oxygen blender. The system is successfully able to take the SpO₂ levels of the patient as feedback and actuate the blender to change the FiO₂ to maintain oxygen saturation at desired level. This is an automatic system and avoids all the problems that could result through the manual control of blender. However, the closed-loop system without the controller shows higher settling time with oscillatory response. A PID controller is designed to overcome these limitations and is added to the closed-loop system.

The PID controller is designed through the methods of direct synthesis and root locus to provide the values of K_p , K_I , K_D , as shown in Table 2.

Figure 6 shows the FiO₂ output of the blender when the system is simulated with designed controllers. It is observed that the level of FiO₂ is increased initially and settles at value of 0.228.

Table 2 Derivative, integral, and derivative components of the PID

Parameters	Direct synthesis method	Root locus method
K_p	0.07717	1.5795
K_I	0.001189	0.0347
K_D	1.244	22.2170

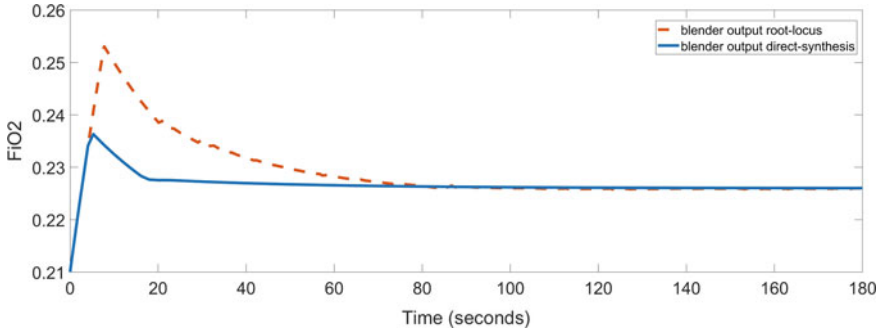


Fig. 6 FiO₂ output of the blender in closed-loop system with PID controller designed through root locus and direct synthesis methods

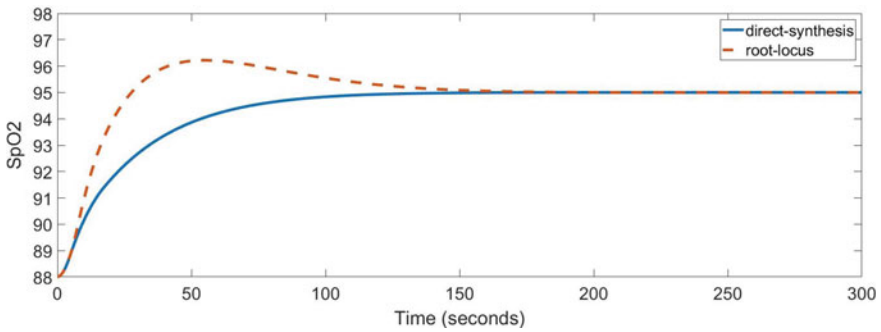


Fig. 7 SpO₂ response of closed-loop system with PID controller designed through root locus and direct synthesis methods

Figure 7 shows the output of the system when PID designed through direct synthesis method and PID designed through root locus method is implemented. The PID designed through both the methods, i.e., direct synthesis and root locus, improve the performance of the closed-loop system. It is seen that the FiO₂ output of the blender does not vary much, and it initially goes up to 0.236 in case of PID designed through direct synthesis method and around 0.254 for PID designed through root locus method. It then stabilizes and gives FiO₂ around 0.225–0.23.

The performance of closed-loop system using the controllers is computed in terms of transient and static characteristics and is shown in Table 3. Through this table, it is observed that PID designed through root locus method has a slight overshoot but it only lasts for about 55 s. The time needed for the PID designed through direct synthesis to settle at a steady value is less as compared to the one designed through the root locus. The time constant needed for the direct synthesis designed PID is less as compared to that needed for PID designed through root locus. It can be said that initially, the transient behavior of the PID designed through the root locus method is better, and it gives better response due to lesser rise time and time constant as

Table 3 Performance comparison of PID designed through direct synthesis method and PID designed through root locus method

Parameter	Direct synthesis method	Root locus method
Rise time	58.7483	18.2357
Settling time	103.8212	139.2651
Peak overshoot %	0.0027	1.2863
Peak time	212	55
Time constant	15	14
Steady-state error (ess)	0.0205	0.0345

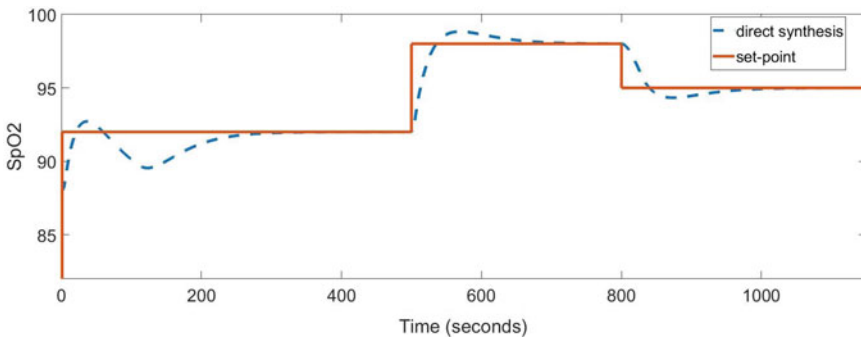


Fig. 8 SpO₂ response of the closed-loop system with PID designed through direct synthesis method when varying set point values are given

compared to the PID designed through the direct synthesis method. However, the steady-state error and the settling time are lesser in case of the PID designed through direct synthesis as compared to the PID designed through root locus method.

The performance of the system with change in desired oxygen saturation level is investigated. The system is subjected to change in desired level from 92 to 98 and then to 95. The response to change in desired level for both the controllers is shown in Figs. 8 and 9.

It can be seen that the controller is able to regulate the oxygen flow rate so that the desired SpO₂ level is reached. It is seen that the control strategy is robust to achieve the desired SpO₂.

5 Conclusion

In this work, an automated oxygen blender for the regulation of SpO₂ levels in hypoxia patients is simulated. A closed-loop system is designed using PID controller that actuates the blender to provide FiO₂ within the practical range of 0.21 to 0.4.

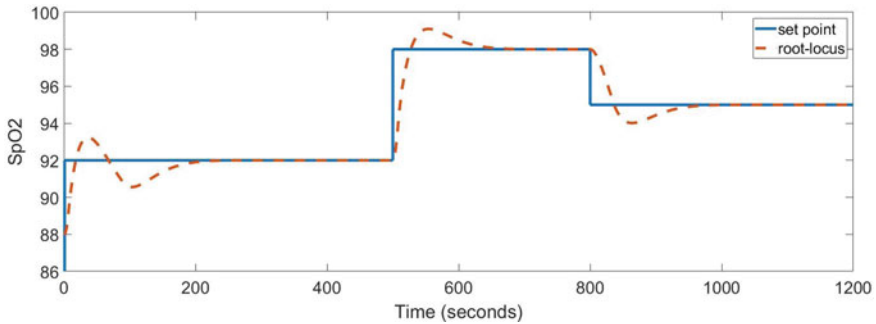


Fig. 9 SpO₂ response of the closed-loop system with PID designed through root locus method when varying set point values are given

The controller was designed using direct synthesis and root locus method and performance of PID is evaluated. The values of steady-state error, settling time and overshoot for the controller designed using direct synthesis method are 0.0205, 103.8212, and 0.0027%, respectively, and that for the controller designed using root locus method are 0.0345, 139.2651, and 1.2863%, respectively. An improvement in the performance of the system is observed by the addition of the controller. The results indicate that closed-loop system can be used to actuate the blender for regulation of the oxygen flow rate to get the desired oxygen saturation level in the patients, and hence, self-regulation of the blender is made possible. This system, however, has been simulated with neonatal parameters and can be extended to adults by replacing the parameter values with the values suitable for adults. The simulation has been performed, and the future aspects involve hardware implementation of the proposed control system and investigating the application of controllers such as model predictive controller and fuzzy logic controller.

References

1. Tehrani F, Rogers M, Lo T, Malinowski T, Afuwape S, Lum M, Terry M (2002) Closed-loop control of the inspired fraction of oxygen in mechanical ventilation. *J Clin Monit Comput* 17(6):367–376
2. Iobbi MG, Simonds AK, Dickinson RJ (2007) Oximetry feedback flow control simulation for oxygen therapy. *J Clin Monit Comput* 21(2):115–123
3. Alkurawy LE (2013) Design of an efficient controller for arterial oxygen saturation in neonatal infants. University of Missouri-Columbia
4. Yu CL (1986) An arterial oxygen saturation controller. Rensselaer Polytechnic Institute
5. Yu C, He WG, So JM, Roy R, Kaufman H, Newell JC (1987) Improvement in arterial oxygen control using multiple-model adaptive control procedures. *IEEE Trans Biomed Eng* 8:567–574
6. Morozoff PE, Evans RW (1992) Closed-loop control of SaO₂ in the neonate. *Biomed Instrum Technol* 26(2):117–123

7. Mollazadeh-Moghaddam K, Burke TF, Dundek M, Yeung SH, Sharma R, Ravi R, Bellare A (2020) A low-cost Venturi ambient air-oxygen blender for neonatal oxygen therapy. *Acad J Pediatr Neonatol* 8(5):48–56
8. Islam MM, Ullah SMA, Mahmud S, Raju STU (2020) Breathing aid devices to support novel coronavirus (COVID-19) infected patients. *SN Comput Sci* 1(5):1–8
9. Ogata K (1997) *Modern control engineering*. Prentice Hall, Upper Saddle River, NJ

An Interleaving Approach to Control Mobile Device and Elements via Screen Buffer and Audio Streaming



Jayavel Kanniappan, Rajesh Kumar Jayavel, and Jithin Gangadharan

Abstract Nowadays mobile devices are powered with huge features (voice enabled, OnDevice AI, health, IoT) which requires device access and validation methods should be in smarter and efficient way. In this paper, we proposed a unified and efficient framework design to access remote devices via screen buffer and live audio streaming concepts which can benefit multiple projects. Our framework has 5 novel aspects: (1) streaming your devices screen buffer in real time using decoding raw H264 buffers (2) streaming your audio input from external PC in real time to access and operate your device in remote conditions (3) dynamic sharing of device screen and elements help to create automated scripts in real time (4) framework is platform and language independent to support Windows and Linux platform (5) faster device screen element detection using json RPC client server architecture. The proposed framework integrated into different projects (Bixby, Internet of Things, OnDevice AI) to benefit with core features like Remote (device access anywhere, live audio streaming), benchmark solution across devices in single frame and real-time device sharing between users and played major role in differentiating our quality products to market in short span of time.

Keywords Mobile device communication · Device screen layout element detection · Audio encoding and decoding · Video encoding and decoding · H.264 encoding · Samsung neural acceleration platform

J. Kanniappan (✉) · R. K. Jayavel · J. Gangadharan
Intelligence & IoT, Samsung R&D Institute, Bengaluru, India
e-mail: jay.sds@samsung.com

R. K. Jayavel
e-mail: rajesh.j@samsung.com

J. Gangadharan
e-mail: jithin.gang@samsung.com

1 Introduction

In modern world the expectations of user experience on mobile devices is almost similar to a personal computer and it is loaded with essential configuration and features like IoT, AI based solutions, voice interactions, health, office, entertainment. Nowadays mobile phone plays pivot role on professional engagement with different levels of users with higher expectations.

Indeed user expectations are high rich of features and reliable solutions to meet business demand. In recent times, most of the solutions are powered with OnDevice AI where solutions provider tries to enhance user experience in terms of speed by accelerating application through GPU and NPU and improvement can't be visible by human eyes. So the challenge rises much higher for mobile application quality assurance more than personal requirement to business demand. As devices become smarter it is very essential to bring efficient and smart automation framework to assure the quality of solutions in short span of time to lead in market.

The proposed framework enabled with state of art features like access your device screen in remote, access your device through remote audio streaming, multi-device streaming in single window, screen rendering/UI elements in single window, device sharing between the users and advanced element identification for script creation and execution. This framework has seamless adaptation capability to integrate with existing automation tools and framework. In market screen rendering concept and other features are not tightly coupled with efficient way to enhance the productivity and quality assurance.

2 Related Work

This research and development has been carried out after deep research on market tools and framework availability for related areas to meet the feature requirements.

In market there are few tools available but with specific and limited features as shown in Table 1. Due to that the requirement cannot be accomplished and cannot be widened, the scope of development for the features and modules listed as part of this proposed research.

One of the competitor Tool A [1] is standalone application which can render the screen and cannot receive or send the audio played by the android mobile application and also it cannot fetch the device screen element. In another instance Tool B [2] can fetch the element of the screen but with lot delay in it but cannot render either device screen or audio playback. As per review there is no existing tools or framework available to stream the user audio and get through the device screen along with application elements at real-time together with very high speed.

As we move forward the current framework built with many use cases such as remote validation of mobile devices and applications, automation of voice enabled devices and assistants in very effective way to improve the user productivity for the

Table 1 Tool comparison with features

Features	Proposed framework	Tool A	Tool B
Element identification	O	O	O
Screen mirroring	O	O	X
Two-way device control	O	X	X
Remote monitoring	O	O	X
Peer-to-Peer device sharing	O	X	X
Automation tool Integration	O	X	X
Screen render latency	<600 ms	<600 ms	4000–6000 ms
Script creation speed	~1 min/tc	X	~5 min/tc

development and assessment. This framework has been deployed in Samsung across multiple projects and integrated with the existing automation framework to adapt this research features and gain the advantages of the complete end-to-end verification of android devices and application related features in efficient manner.

3 Proposed Framework

The proposed framework has master and client architecture through which desktop solution acts as a data requestor and receiver whereas the client devices acts as a data provider wherein continuous data from screen buffer and the layout hierarchy information will be shared with the desktop solution to stream the data constantly in real-time. The framework has an ability and techniques to stream and control any devices based on Android platform by decoding the device screen buffer [3], audio PCM buffer and screen layout hierarchy information which will be fetched at real-time to the master in which user has interface to view and control the client devices.

The framework is capable of identifying the device screen layout orientation changes and also the real-time monitoring of positions on user interaction behaviour in the device screen to fetch the corresponding screen layout content which is intended to control or operate.

The proposed framework has well-researched solutions and modules represented in a layered architecture (Fig. 1) to remotely control the user devices like mobile through device screen and using user voice at real-time without device in hand for enhanced productivity and validation effectiveness.

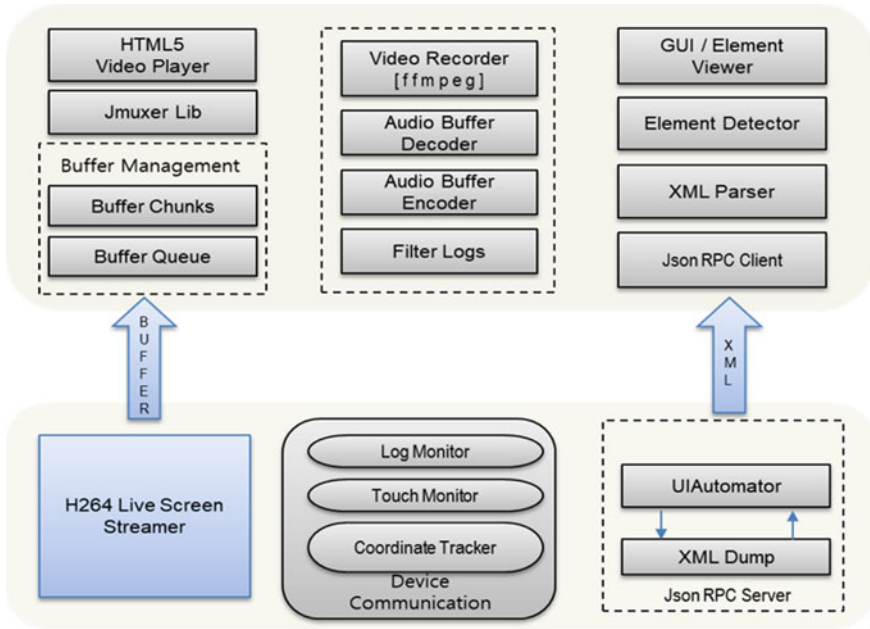


Fig. 1 Architecture design and framework components

The synchronization between the Desktop and Client device is achieved with efficient buffer management system where the chunk of buffered data has been streamed in a real-time with higher frame rate to avoid glitches in continuous streaming. The delay between user operation on the device screen and the action takes place in the client device and then rendering back has been achieved with very low latency to provide better user experience and not missing any device information to visualize on a time critical applications for remote monitoring.

This proposed framework has been a vital factor in remote assistance, monitoring and testing of Android based device application seamlessly with higher efficiency in an easy manner as shown in Fig. 2 and also its been widely used for automation script creation in a simplified manner wherein user just operate the device and its application in usual manner once for the evaluation of features and in parallel, the framework monitors the user execution and operations to know the system user flow using the application layout information monitoring through Android framework in real-time and fetch the element information and passes through the desktop client using json remote procedure call protocol with very low latency to provide real-time experience of script creation in easy manner in very less time and effort.

This framework has majorly three main sources of data to stream over to the desktop client, those are video (mobile screen) buffer, mobile application (element information) layout data and audio (user speech) buffer. All these sources are from the mobile device and desktop client has service to enable all these sources decoded

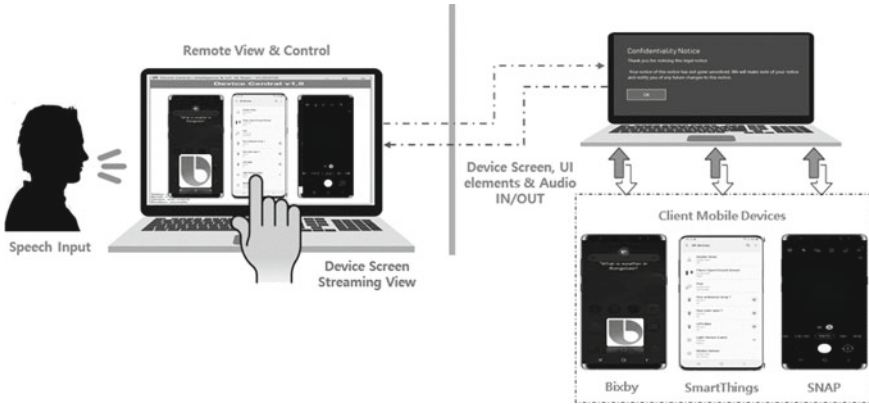


Fig. 2 User flow representation

and rendered with appropriate channels with different techniques adapted for each of these source components as represented in Fig. 3.

In addition to these components, the separate service has been enabled with real-time monitoring of the user device screen navigation with touch gestures to get the element information and user flow captures as an automation script. The controlling

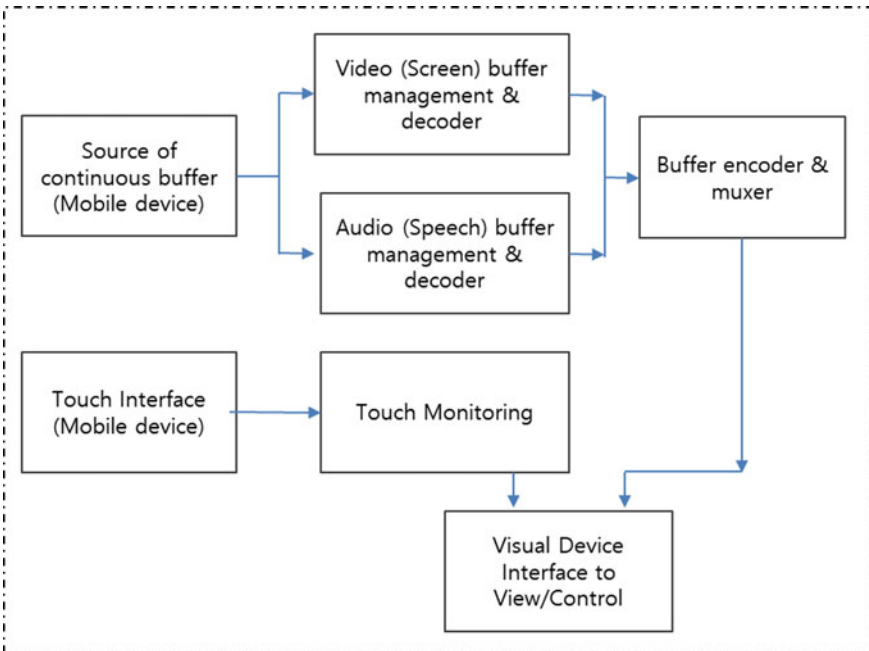


Fig. 3 System flow representation

of the mobile application has been carried out with different techniques and wrapper functions and framework provided by the android platform.

3.1 Streaming Remote Devices Screen

This concept of devices screen buffering and streaming [4] over to the remote user interface is currently used in wide application testing and remote device execution monitoring in the lab setup in various projects, few of among those are voice enabled solution (Bixby) and IoT (SmartThings app) testing and execution monitored remote manner. Using this concept user can control the multiple devices and respective solutions and monitor device status in line manner.

In order to decode and play the live video buffer from the mobile device which we get from android surface flinger and graphics path. Our framework uses mp4 muxer that works in both browser and node environment [5]. It is communication protocol agnostic and it is intended to play media files on the browser with the help of the media source extension. It also can export mp4 [6] on the node environment and expects raw H264 video data and/or AAC (Advanced Audio Coding) audio data in ADTS (Audio data transport stream) container as an input. Each packet consists of 4 bytes of header and the payload following the header. The first two bytes of the header contain the chunk duration and remaining two bytes contain the audio data length. Packet format is shown in Fig. 4.

The buffer chunks of device screen will be decoded using mp4 muxer techniques [7] with buffer chunk represented in Fig. 5 and has been streamed continuously in which the video buffer [8] has been rendered in the desktop client with high speed. The buffer management has been tuned to render the video buffer without any glitch and also it changes the buffer stream depend on the device orientation by users.

The continuous buffer is decoded constantly to render the device screen smoothly with buffer array defined in buffer management in periodic transformation as depicted as shown below and also at specific instance of the buffer stream, buffer data frame will be measured with below expression.

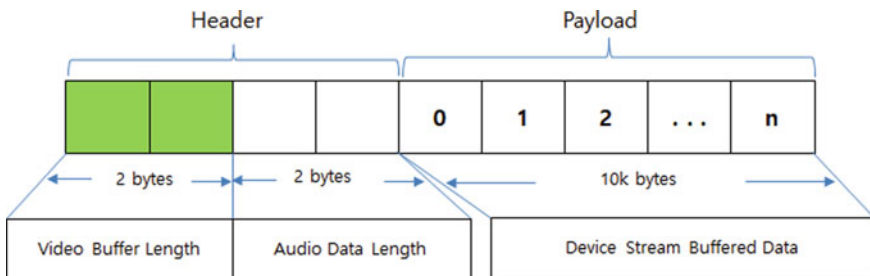


Fig. 4 H264 video buffer data packet format

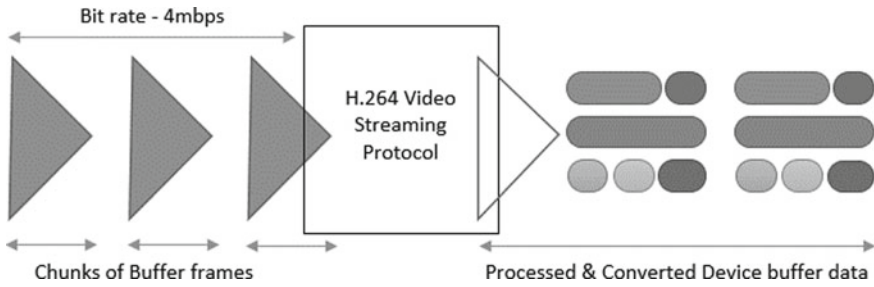


Fig. 5 Buffer chunk management of H.264 streams

Here ‘*a*’ is buffer size at ‘*n*’ number of times buffer chunk streamed and ‘*d*’ is the buffer size.

$$a_n = a_1 + (n - 1)d$$

In case of continuous video buffer streaming, then the total amount of buffer size streamed using the buffer management concept has been depicted below.

$$f(x) = a_0 + \sum_{i=1}^n i = \frac{nx}{2} [2a_i + (n - 1)d]$$

where ‘*a*’ is first buffer chunk, ‘*d*’ is the buffer size for periodic data frame and ‘*n*’ is number of chunk data which is near to infinite.

The framework uses H264 baseline profile for continuous video buffer streaming [8]. This is the simplest profile used mostly for the low-power, low cost devices, including some video conferencing and mobile applications. Even though this baseline profiles, the compression ratio can be achieved about 1000:1 (i.e.) a stream of 1 Gbps (Gigabits per second) can be compressed to about 1 Mbps (Megabits per second). The proposed framework supports multi device parallel live mobile screen streaming with orientation changes supported.

3.2 Streaming User Speech Live to Remote Device

The important aspect of video streaming is Audio associated with the mobile application and the need of feeding the user real speech into the system to stream remotely associated with the video rendered on the desktop client.

There are several use cases associated with the requirements of Audio (speech) transmission, in current scenario this concept is used to test voice enabled solutions and voice assistance and related features embedded into mobile applications (e.g.

Bixby, Alexa and Google Assistant) remotely with real time user voice streaming to the remote mobile devices.

In case of the speech transmission to the voice based solutions and assistance, the another important aspect is to retrieve the dialogue response (spoken text) from the voice assistant back to the remote user system as a live stream using audio decoding and buffer management techniques as shown in Fig. 6. The response audio buffer has been captured through mobile client as a separate service monitoring the response audio and decodes and stream back to the desktop client and it plays back at the remote system.

The proposed framework allows the bi-directional communication between client and server [9], by which client has Socket.IO [10] in the desktop client, and a server has also integrated the Socket.IO pkg, while audio buffer (PCM) can be sent in a byte format. To establish the connection, and to exchange data between client and server, Socket.IO uses Engine.IO. This is a lower-level implementation used under the hood. Engine.IO is used for the server implementation and Engine.IO-client is used for the client. By using this advanced technology [11] assuring no audio data loss, while real time streaming with our own buffer management as mentioned below

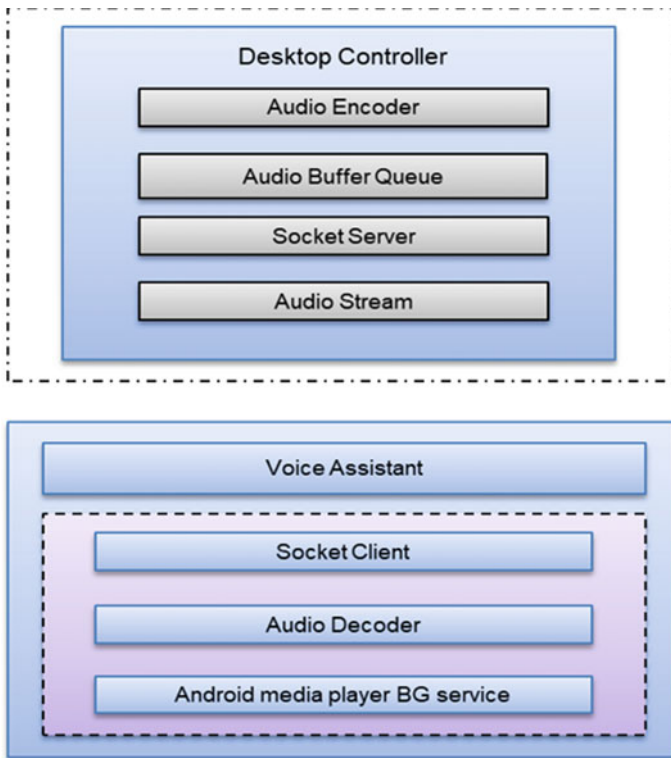


Fig. 6 Audio streaming and encoding design flow

using 4 bit Pulse code modulation scheme as shown in Fig. 7. This helps to stream the live audio from desktop to mobile device and from mobile device to desktop seamlessly.

Buffer calculation formula:

$$\begin{aligned} \text{Sampling rate } S[r] &= 16,000 \text{ samples /seconds frame duration} \\ &= 20 \text{ ms sample size} = 2 \text{ bytes channels} = \text{mono} \end{aligned}$$

So, each frame contains $16,000 \text{ samples/seconds} * 0.02 \text{ s} * 2 \text{ bytes/sample/channel} * 1 \text{ channels} = 640 \text{ bytes}$.

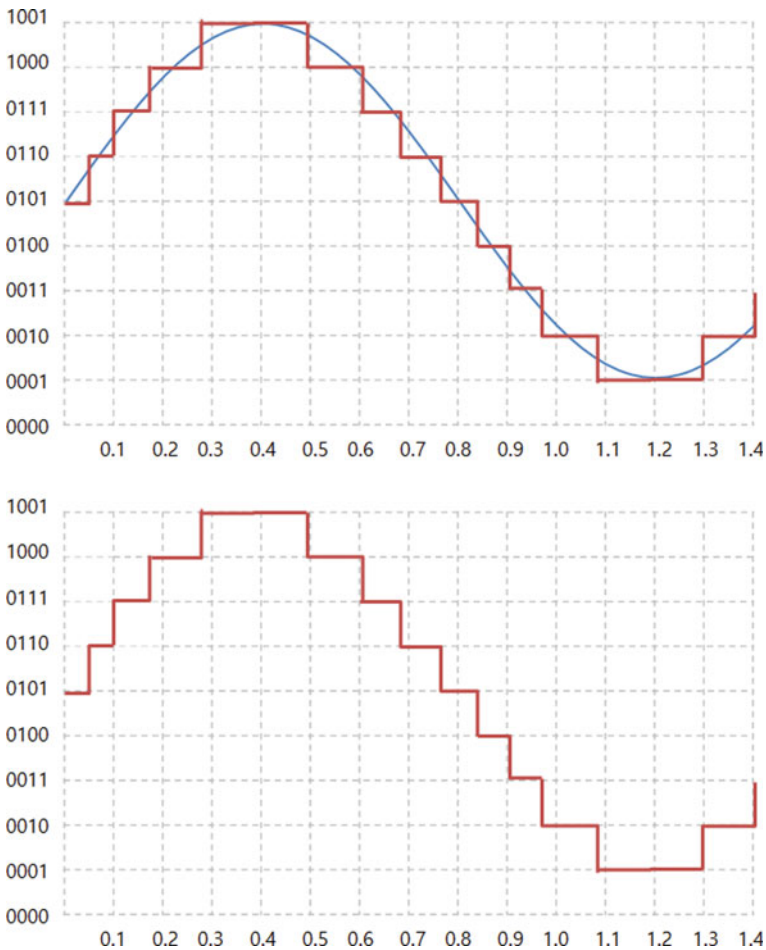


Fig. 7 Sampling and quantization of a signal for 4-bit LPCM

3.3 Faster UI Element Properties Detection with JRPC

This feature is currently used for android automation script creation. The traditional approach of script recording is carried out with invoking child process of instrumentation which takes 2–3 s(s) whereas with current approach created JRPC [12] server within instrumentation framework and connect the server from desktop client and this has been experimented with integration to the script creation modules for the evidence and research analysis to verify the quality of the latency improvements.

In this framework, Json RPC concept has been used for communication of device screen layout element information from the mobile devices to desktop client. JsonRpc comes with a streaming server and client to support apps of all types and not just HTTP. The JsonRpcClient and JsonRpcServer have methods that take InputStreams and OutputStreams, so that it is very flexible in communicating the data over the channels very efficiently in real-time [13].

As part of this module there is separate service running to monitor the user actions on the device to know the user scenario under action and based on that the actual screen element will be identified with layout parsing at real-time and communicated to desktop client for the script creation as and when the user performs the scenarios as shown in Fig. 8.

Usually in traditional methods of element fetching, the shell commands has been used which takes a lot of time in initializing the android debug bridge which is the communication channel for the android shell operations and hence overall delay introduced is huge and thus can't make bigger impact on the user experience in real-time element identification. Here we invoke the instrumentation only one time and reuse the instance for further operations.

Using this current approach we are able to fetch the android element properties in milliseconds and not exceeding 1 s as shown in Fig. 9 against the competitor data that has been analyzed for benchmark.

Overall it improves the productivity of automation script development and testers can create automation scripts in less time as shown in result evaluation in Fig. 10 and cover more scenarios without errors in ease manner.

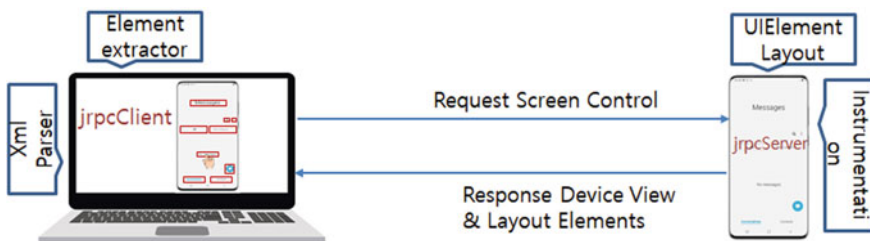


Fig. 8 Json RPC approach to retrieve the Layout elements

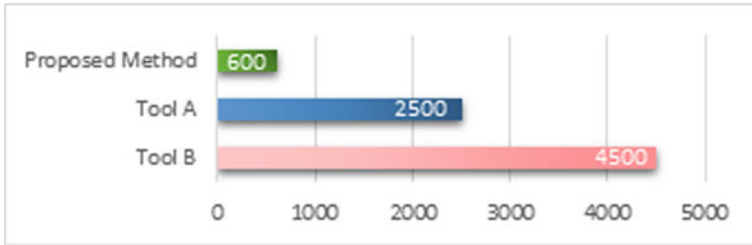


Fig. 9 Comparison of element detection latency

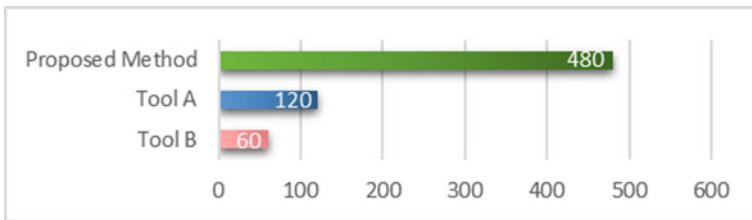


Fig. 10 Comparison of tools Avg. script creation per day

3.4 Remote Screen Sharing with Peer-To-Peer Mode

In many instances it is required for the developer or tester to observe the application behaviour on the devices of peer member or co-workers to analyze any problems and to bring it to conclusion in easy way. But if the members are decentralized and located at different geographical locations, then it is very hard to share the devices physically and only options to leverage the benefits of the software platforms and our frameworks able to achieve the same.

In this proposed approach, it has been addressed to solve the above problem while common team is decentralized at various locations by adapting various features [14] proposed as part of this research with peer-to-peer sharing of mobile application data over the network. It is the very unique concept which we introduced to our framework without having centralized central and instead the data is communicated over the IP based network.

Here our framework acts as both client and server. If user A required specific device to test/debug for defined duration from user B by without sharing the physical device, it can be shared remotely for the particular device screen to user A directly using Socket.IO peer to peer [15] sharing concept (IP based network communication). User A should give a device access request to user B, once user B approve the request as shown in the Fig. 11. User A can view and control the shared device in real time [16] with screen and element properties of that particular device in remote system. There is no existing tools or framework available to achieve this and hence this proposed

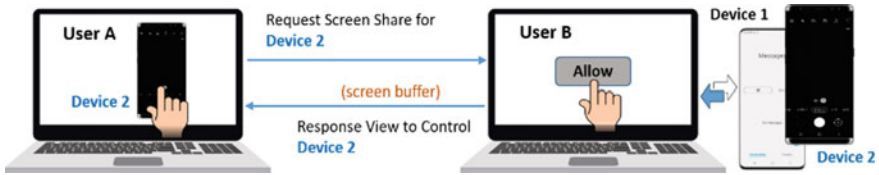


Fig. 11 Peer-to-peer device screen sharing and control

approach is unique and widely useful when teams are located across geographical locations or when team is working in remote conditions and lab environment.

This feature in our framework helps to overcome device shortage in testing and development and also it helps in avoiding device not reaching to real user after the usage.

3.5 Global Script Creation for Android Automation

In the world of fast growing android platform community, it is important to meet all quality criteria for all android application to meet global customer demands. The automated testing approach plays a significant role in the quality assessment of android applications to overcome above mentioned challenges.

Nowadays android automation testing tools are very common and several frameworks available built from scratch to test applications separately. The main intention behind for any android automation tools to reduce human effort is to increase productivity and coverage in short span of time. But it is possible only by considering the common framework to achieve with less time. Hence it has been analyzed and studied deeply on generalizing the tool as a global framework to android automation community so that any android automation tool developers can integrate our framework and make android automation tools specific use cases alone with less effort with minimal duration. This proposed and implemented framework alone covers 80% requirement of android automation like live screen rendering, fastest android element detection, Audio speech streaming with noiseless coding techniques [17] and live log parser. Only 20% of custom specific use cases and log parsing techniques will vary tool specific based on the projects and application requirements.

In automation framework, the scenario executions are carried out with set of actions or use cases on android devices based on the predefined set of instructions been given to the tool in terms of automation scripts. Scripts can be recorded automatically when user does any action to the devices based on start and stop record feature. Every tool which supports android script record automatically has its own defined syntaxes. The current approach designed to generate tool specific scripts syntax based on the static pre-defined template as represented in Tables 2 and 3, so that testers need not be familiar of script syntax for each tools with different syntax and formats whenever they use different android automation tools. This framework

Table 2 Generic execution API syntax

Android generic actions (Fixed)	Custom API's (Tool specific)
Touch	Touch[x, y]
Long Touch	LongTouch[x, y]
Double Touch	DoubleTouch[x, y]
Swipe up	SwipeUp[x1, y1, x2, y2]
Swipe down	SwipeDown[x1, y1, x2, y2]
Swipe left	SwipeLeft[x1, y1, x2, y2]
Swipe right	SwipeRight[x1, y1, x2, y2]
Type text	EnterText[data]

Table 3 Generic validation API syntax

Android validations (Fixed)	Custom API's (Tool specific)
Element exist	Verify[gui element]
Element not exist	VerifyNot[gui element]
Element partial exist	VerifyPartial[gui element]
Element partial not exist	VerifyNotPartial[gui element]

has the solution for the above practices followed by android automation developers. We have analyzed that what and all actions a user can do on the android phone in terms of action and validation of the screen using Jrpc client server concepts described in earlier *section C*.

In general device actions possible are Touch, Long Touch, Swipe up, Swipe down, Swipe left, Swipe right, Type Text and possible validations are element exist, not exist, partial exist, partial not exist (with combination of gate operators AND and OR). These are all primarily blocks of any android automation tool [18] consists for any application testing in terms of visual user interface. This set of action and verification concept can be re-used from this proposed research areas and can dynamically create the scripts [18] for quick development instead of developing the script creation module from scratch.

3.6 *Dynamic Multiple Device Screen Rendering*

In android devices one of the important testing method is benchmarking with different devices in terms of latency and sluggishness of applications. Most of the time this test has been carried out manually to understand the visual behaviour of the applications in a live manner and observations are monitored to conclude the quality of each key parameters. But fact is that the difference in user behaviour completely vary from tester to tester due to change in execution speed and the dynamics of user gestures to compare the feature behaviour and all varies in few milliseconds, so most of the time



Fig. 12 Multiple devices execution and Rendering

it is difficult to conclude calculation of KPI's (Key Performance Indicator) and due to this existing challenge we came up with reliable test approach with our proposed framework where multiple devices (within the system limit) are connected to system and this framework would render the live screens and display it in our web layout GUI same time in responsive manner [19]. So any actions performed on the device in an automated way would reflect at a time to all devices and observe the real latency difference live in single place as shown in the Fig. 12.

In order to achieve this we use encapsulation method with live streaming [4] of H.264 baseline profile [20] from android device with USB port forwarding, once the buffer reach to desktop using USB port forwarding, the dynamic data structure queue has been created to save the byte buffer and once the enough buffer is ready for the frame creation, it would concatenate it and decode the H.264 base line profile as described in earlier *section A*. Using JMuxer [7] non-standard decoder, the decoded buffer is been played using native player. This is very useful for any benchmarking in android side and identify the inferior part of our application.

4 Results and Impact

This framework integrated with Samsung projects which helps to reduce huge manual effort as shown in Table 4. This framework featured with (device screen access in Remote PC, device access through remote audio, flashing and device share) used across the projects Bixby, SNAP, VizInsight, SmartThings App, voice benchmark of competitor (Alexa, GA) devices.

Overall effectiveness is shown in Table 5.

Table 4 Effort comparison of proposed approach against manual approach

Manual	Automation (Proposed framework)
Feature benchmarking between mobile devices (S10, M30, Fold)—2 days per device/engineer	24 * 7 automation execution
Bixby E2E script development time—100 per day/engineer	100% automated system (400 per day)
IoT—SmartThings E2E script development time—80 per day/engineer	100% automated system (320 per day)
Device sharing between users	100% automated system
Live audio streaming input to remote mobile	100% automated system
Manual log collection of features and report generation	90% manual effort reduction across the devices
Dynamic device log parser for feature debugging	100% automated system and real-time devices logs in single screen

Table 5 Concept impact and effectiveness results

S. No.	Projects	No. of scripts	Man days saved
1	Bixby	15,000	62.5
2	IoT	3000	12.5

5 Conclusion

In recent mobiles software innovation and development is to market embrace opportunities, challenges and innovative automation framework to assure best quality and meet customer expectations on time. This framework evidence with proven deployment across the Samsung projects and brought huge value add in terms of efficiency and test coverage enhancements. The framework has potential to become open source global community tool which has common framework to access and control all android devices in market in more efficient way.

References

1. Denis, Sorccu & Koral. Smartphone test farm—control and manage android devices from your browser. 31 May 2020. <https://github.com/openstf/stf>
2. Google. Android UI testing framework for cross-app functional testing. <https://developer.android.com/training/testing/ui-automator>
3. Video coding for low bit rate communication, ITU-T Recommendation H.263, Version 2, Feb. 1998
4. Liu T, Choudhary C (2007) Scalable coding and wireless streaming of lecture videos for mobile learning. Adv Technol Learn 4(2)
5. OpenJS Foundation. Building NW.js. May 27, 2009. <https://nwjs.io/>

6. LeGall DJ (1991) MPEG: a video compression standard for multimedia applications. *Commun ACM* 34:46–58
7. Das SK (2018) jMuxer-javascript mp4 muxer. <https://www.npmjs.com/package/jmuxer>
8. Conklin GJ (2001) Video coding for streaming media delivery on the internet. *IEEE Trans Circuits Syst Video Technol* 11(3)
9. Kalita L (2014) Socket programming. *Int J Comput Sci Inform Technol (IJCSIT)* 5(3):4802–4807
10. Rauch G (2016) Socket.IO-Realtime application framework (Node.JS server). <https://github.com/socketio/socket.io>
11. Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s, ITU-T Recommendation G.723.1, Mar. 1996
12. JSON-Remote Procedure Call (2005) <https://en.wikipedia.org/wiki/JSON-RPC>
13. JSON-RPC Working Group. JSON-RPC 2.0 Specification. 26 March 2010. <https://www.jsonrpc.org/specification>
14. Motta G, Storer J, Carpentieri B (2000) Improving scene cut quality for real-time video decoding. In: *Proceedings of the IEEE data compression Conference, DCC'00*, pp 470–479, Mar 26–30, 2000
15. Ripean M (2001) Peer-to-peer architecture case study: Gnutella network. In: *Proceedings of the first international conference on peer-to-peer computing*, 27–29 Aug 2001, pp 99–100
16. Wyner A, Ziv J (1976) The rate distortion function for the source coding with side information at the receiver. *IEEE Trans Inform Theory* IT-22:1–11
17. Slepian D, Wolf JK (1973) Noiseless coding of correlated information sources. *IEEE Trans Inform Theory* IT-19:471–480
18. Filipe. NodeJS wrapper for UiAutomator. 8 Dec 2020. <https://www.npmjs.com/package/uiautomator-wrapper>
19. Frain B (2013) Responsive web design with HTML5 and CSS3
20. Greenbaum GS. Remarks on the H.26L project: streaming video requirements for next generation video compression standards. In: *ITU-TSG16 (Q15), Doc. Q15-G-11*, Monterey, CA

Optimization of Algorithms for Simple Polygonizations



Maksim Kovalchuk, Vasyi Tereshchenko , and Yaroslav Tereshchenko 

Abstract In this paper, we propose modifications of some known efficient algorithms for solving the problem of MAP (finding a simple polygon of minimum area). We propose an algorithm that somewhat expands the computational capabilities of the considered algorithms and has the complexity $(n^2 \log(n))$ using $O(n)$ memory. Also, we have proposed an original postprocessing that evaluates the performance of other algorithms by optimizing the configuration of local points within the specified complexity estimate $(O(n^2), O(n))$. Experimental results show that it is advisable to use the algorithm in combination with postprocessing taking into account time. Unless there are strict time limits, postprocessing can be useful for randomized algorithms.

Keywords Polygon generation · Minimum area polygon · Divide-and-conquer technique · Postprocessing

1 Introduction

One of the important and useful tasks of computational geometry and discrete optimization is the construction of a polygon of minimum area on an arbitrary set of input data (points). We offer various solutions for both deeply theoretical research and applied software. One type of problem is constructing different shapes optimizing parameters like perimeter and area. The paper considers the problem of finding the minimal area polygon (MAP), vertices of which are all points from the given set. In practice, this problem arises in the context of geo-sensor networks [1, 2] and GIS systems [3, 4]. Like traveling salesman problem, MAP is NP-hard, which was proven in [5]. It makes it almost impossible to solve the problem provably optimal for big enough data sets. Most related works focus on approximation of MAP, rather than finding the best possible polygonizations [6, 7].

M. Kovalchuk · V. Tereshchenko (✉) · Y. Tereshchenko
Taras Shevchenko National University of Kyiv, Kyiv 01601, Ukraine
e-mail: vtereshch@gmail.com

Analysis of existing approaches. The simplest algorithm for finding the optimal MAP is MAP_Permute-And-Reject (MAP_PAR) with complexity $O(n!)$. MAP_PAR brute-force checks all permutations of each unique simple polygon by the minimum area. In [8, 9], MAP was formulated as the IP problem. This approach works faster than MAP_PAR, without losing optimality. The MAP for 24 points was calculated in approximately 90,000 s [9].

In [10], the authors proposed the MAP_Greedy algorithm. The idea of the algorithm is that starting from some edge of the convex hull for a given set of points, and we choose a point that forms with this edge the maximum empty triangle. Next, the current edge is removed and replaced by the found edges of the triangle. This procedure is performed for the next edge of the convex hull. The processing ends by the exhaustion of all isolated points of the given set. The complexity of the algorithm is $(O(n^3), O(n^2))$. The MAP_DAC algorithm with complexity $O(n^2)$ is described in [11]. At the division stage, the authors recursively divide the current set of points into two equal of the subset and perform this procedure for smaller sets. At the merging stage of the algorithm, they recursively merge the found minimum polygons, obtaining the resulting polygon.

Randomized algorithms were proposed in [12, 13]. MAP_RS [13] consist of six strategies how to choose the initial triangle, next point, and next edge. We believe that complexity of MAP_RS is $O(qn^3)$, where q is the number of trials and n is the number of points. MAP_RAND [12] starts with the triangle. On each step, it chooses a random point and connects it with the edge that minimizes the polygon area. MAP_RAND is the only algorithm that processes both inside the polygon and outside the polygon types of points. In [12], MAP_RAND described with $O(qn^2 \log(n))$ complexity, but we believe that complexity can be improved to $O(qn^2)$ using polygon visibility algorithm [14]. In [13], algorithm based on ant colony optimization was proposed.

In Sect. 2, we formulate the MAP problem, describe a modification of the MAP_DAC algorithm, and propose a simple postprocessing technique. In Sect. 3, we analyze the complexity of the described algorithms and their properties. In Sect. 4, we present implementation details, a comparison of MAP_DAC2 with existing approximate MAP algorithms, and data processing results from other algorithms.

2 Improvement of Existing Algorithms for Solving the MAP Problem

Let's formulate a mathematical formulation of the **MAP** problem.

Problem A set S of n points is randomly generated on the plane. It is necessary for each generation S to construct a simple polygon of the smallest area, which would cover a given set of points and whose vertices are all points of the set S .

2.1 Modification of the MAP_DAC Algorithm

The main steps of the modified MAP_DAC [11] algorithm is following.

1. If the number of points is less than six, apply the algorithm MAP_Permute-And-Reject, otherwise:
2. **Preprocessing.** A given set S of n points on a plane **are presented as an** ordered array of points is created in the form of a list $U = \{P_{ij}, i, j = 1, n\}$, where i and j are indices specifying the order of a point in the array along the coordinates x and y , respectively.
3. **Partition.** At each recursion step, we split (by the median l_x) this set of points (as an indexed list U) into two subsets S_{1x} and S_{2x} of equal cardinalities. The dividing process ends when the subsets have such a number of points for which the given problem is solved trivially.
4. **Merger I.** At each step of this stage, we recursively construct a minimal polygon, by merging the minimal polygons P_1 and P_2 obtained in the previous step. In the last step of this stage, we obtain the required minimum polygon. Denote it by P_x .

Similarly, we can construct a simple polygon of minimum area by coordinate y . In this case, we recursively split (by the median l_y (horizontal)) U , using index j , into two subsets S_{1y} and S_{2y} of equal cardinalities. And at the merging stage, we will construct a minimal polygon P_y . In principle, we can apply this procedure to any direction and obtain a minimal polygon. The search time P_y will be $O(n^2 \log n)$ in the general case.

5. **Merger II.** At this stage, we merge the polygons P_x and P_y , obtaining the minimum polygon P_{xy} .
6. **Postprocessing.** At this stage, we perform additional processing P_{xy} and thus obtain the final result. We will describe in detail the postprocessing in the next section.

We propose to maintain two solutions for each point subset, one for vertical separation and other for horizontal separation. In such a way, we bring more local optimality without much loss of execution time efficiency. Our algorithm MAP_DAC2 performs the next steps.

- (1) If the number of points is less than six, return MAP using MAP_Permute-And-Reject;
- (2) Suppose we have two subsets S^1_x and S^2_x of equal cardinalities, sorted by the X -coordinate;
- (3) Solve recursively (from step 1) for S^1_x and S^2_x getting polygons $P^{1;1}_x, P^{1;2}_x, P^{2;1}_x$, and $P^{2;2}_x$
- (4) Consider polygons PM^1_x, PM^2_x, PM^3_x , and PM^4_x received by merging pairs $\{P^{1;1}_x, P^{2;1}_x\}, \{P^{1;1}_x, P^{2;2}_x\}, \{P^{1;2}_x, P^{2;1}_x\}$, and $\{P^{1;2}_x, P^{2;2}_x\}$, respectively, Fig. 1a–d;
- (5) Let P_x be one of the PM^1_x, PM^2_x, PM^3_x , and PM^4_x with the minimal area;

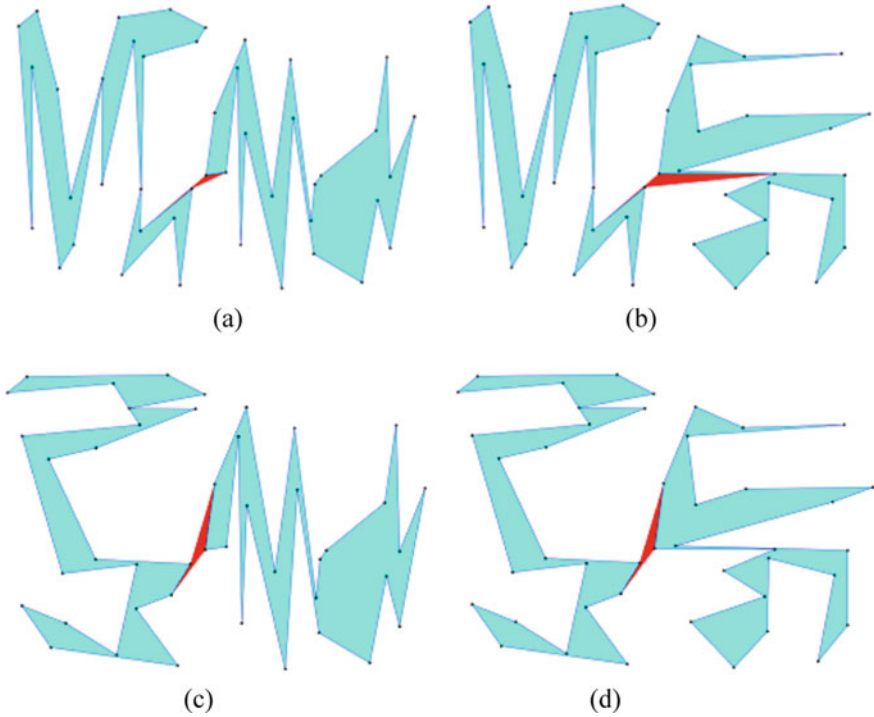


Fig. 1 Eight different polygons considered in MAP_DAC2 (blue—polygons received recursively, red—merging quadrilateral), where **a** PM^1_x , **b** PM^2_x , **c** PM^3_x , and **d** PM^4_x

- (6) Suppose we have two subsets S^1_y and S^2_y of equal cardinalities, sorted by the Y -coordinate;
- (7) Solve recursively (from step 1) for S^1_y and S^2_y getting polygons $P^{1:1}_y, P^{1:2}_y, P^{2:1}_y$, and $P^{2:2}_y$;
- (8) Consider polygons PM^1_y, PM^2_y, PM^3_y , and PM^4_y received by merging pairs $\{P^{1:1}_y, P^{2:1}_y\}$, $\{P^{1:1}_y, P^{2:2}_y\}$, $\{P^{1:2}_y, P^{2:1}_y\}$, and $\{P^{1:2}_y, P^{2:2}_y\}$, respectively, Fig. 2a–d;
- (9) Let P_y be one of the PM^1_y, PM^2_y, PM^3_y , and PM^4_y with the minimal area;
- (10) Return pair $\{P_x, P_y\}$.

All steps of MAP_DAC2 except four and eight are straightforward. Step 4 is described in the original paper for MAP_DAC [11] and can be done in the next way.

- (1) Build polygon E enclosing area between polygons A and B . It consists of two chains, one from each of given polygons, and two tangents that connect border points of chains;
- (2) For all points of the first chain find all visible points on the second chain [14];

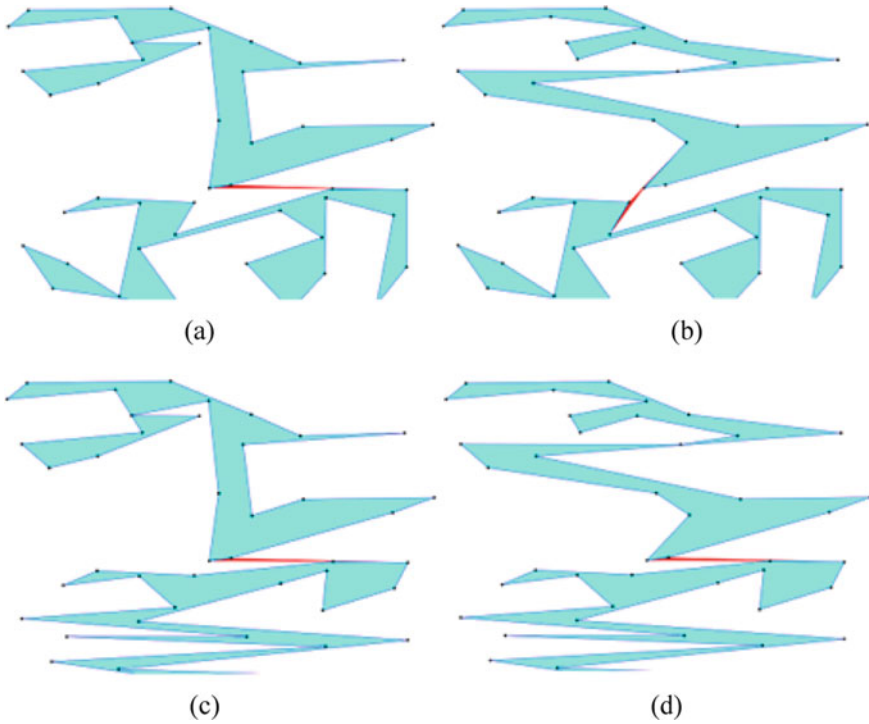


Fig. 2 Eight different polygons considered in MAP_DAC2 (blue–polygons received recursively, red–merging quadrilateral), where **a** PM^1_y , **b** PM^2_y , **c** PM^3_y , and **d**. PM^4_y

- (3) For all consecutive point pairs of the first chain find all visible consecutive point pairs on the second chain that form simple quadrilateral. Choose the smallest quadrilateral.

Step 8 is almost the same as 4, but with horizontal separation. In case, implementation is hardcoded for vertical separation, it is possible to interchange X- and Y-coordinates of points to bring step 8 to 4. Eight polygons considered on steps 4 and 8 are shown in Figs. 1 and 2.

2.2 Postprocessing of MAP Solution

It is typical for MAP approximation algorithms to use heuristics that greedily choose point, edge, quadrilateral, etc. Such heuristics may be optimal on the current step but can be no optimal in the future. We propose a simple postprocessing technique that removes one point at a time and tries to add it in the more optimal position. Postprocessing has the following steps.

Let the considered polygon be presented in the form of a linked list (a cyclically ordered set of vertices) $P = P_1, P_2, \dots, P_n$.

1. Consider sequentially all points P_i of polygon P .
2. If we can remove P_i so that $P = \{P_1, \dots, P_{i-1}, P_{i+1}, \dots, P_n\}$ is a simple polygon, and the segment $(P_{i-1}, P_{i+1}) \in P$ is visible with P_i (we can use the polygon visibility [14]), then remove P_i , else go to step 1 and consider the next point.
3. If P_i is inside P , then we find the edge (P_{j-1}, P_j) of $*$, which is visible from P_i and maximizes the area of the triangle $\{P_{j-1}, P_i, P_j\}$.
4. If P_i is outside P , then we find the edge (P_{j-1}, P_j) of $*$, which is visible from P_i and minimizes the area of the triangle $\{P_{j-1}, P_i, P_j\}$.
5. Insert P_i in P as a new value of the polygon and go to step 1, taking into account the next point.

To check whether the point is inside or outside the polygon, we can use the windmill algorithm. To find the edge (P_{j-1}, P_j) in points three and four, you can use the algorithm of visibility of the polygon, computed during verification [14].

In practice, we ran several trials of postprocessing until no more area minimizing points rearrangements are possible. Example of postprocessing optimization is shown in Fig. 3.

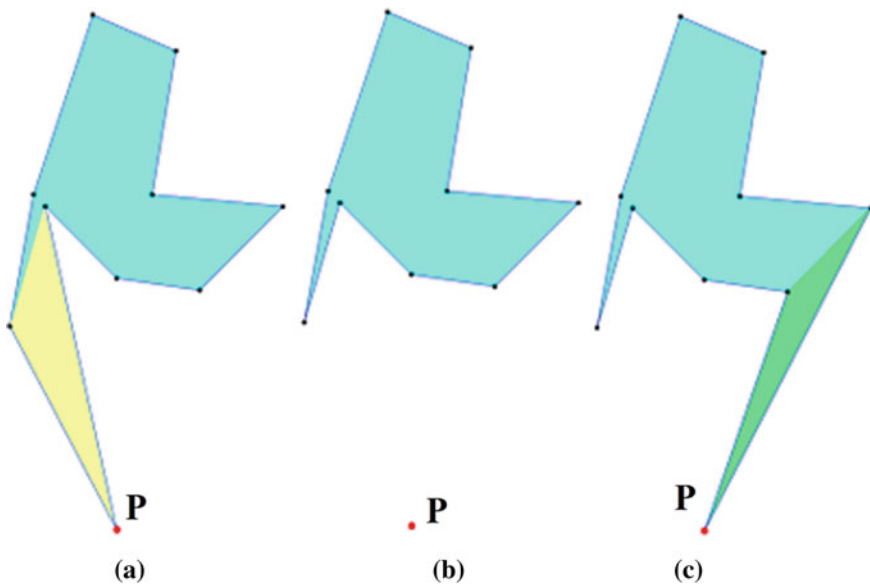


Fig. 3 Example of postprocessing (red point P is postprocessed, yellow—removed triangle, green—added triangle), where **a** initial polygon, **b** P removed from polygon, and **c** P inserted in better position

3 Complexity and Correctness

3.1 Complexity

Theorem 1 *The MAP problem on a random sample of the set S from n points can be solved by a modified algorithm MAP_DAC2 in $(n^2 \log(n))$ time.*

Proof MAP_DAC2 is recursive algorithm. During the one iteration, it sorts out points twice, calls merging function eight times, and calls itself recursively four times with two times smaller point set. Complexity of sorting is $O(n \log(n))$. Complexity of merging function is $O(n^2)$ [11]. That is, we have recurrence $T(n) = 4 \cdot T(n/2) + 8 \cdot O(n^2) + 2 \cdot O(n \log(n))$. After simplification, we get $T(n) = 4 \cdot T(n/2) + O(n^2)$. Last recurrence can be solved using master theorem, and the solution is $O(n^2 \log(n))$.

Theorem 2 Postprocessing can be implemented with complexity $O(n^2)$.

Proof We can check the test in steps 3 and 4 in $O(n)$ time. The complexity of the windmill algorithm is also (n) . If during the check the visibility polygon has already constructed, then we can find the edge (P_{j-1}, P_j) of step 4 by $O(n)$. Therefore, we can process one point in $O(n)$ time, and n points in $O(n^2)$. Testing, we ran postprocessing several times until the polygon area converged. Maximal number of postprocessing trials was ten. Thus, the execution time postprocessing will be also $O(n^2)$.

3.2 Correctness of MAP_DAC2

Correctness of MAP_DAC2 consists in the existence of quadrilateral merging two polygons on steps 4 and 9. MAP_DAC2 tries to merge four pairs of polygons on both step 4 and step 9. During the experimental study, we found pairs that cannot be merged since a quadrilateral do not exist. Despite it, at least one of four pairs always could be combined when running MAP_DAC2 on our data set. We do not know whether it holds in the general case. In the implementation, we assign infinity area to the incorrect polygonization, leading the algorithm to choose one with a smaller area.

3.3 Correctness of Postprocessing

In [12], the authors introduce the concept of “trap region.”

Definition 1 A trap region of polygon P is the area from which no edges of P are completely visible (Fig. 4).

Definition 2 An algorithm that bypasses trap region we will call a *no-trap algorithm*; otherwise, we will call such an algorithm a *trap algorithm*.

Therefore, the strategy of the postprocessing algorithm is to bypass the trap regions.

To have a no-trap algorithm, it is necessary to not lead input points into the trap region or somehow deal with the trap region otherwise. Peethambaran et al. [12] proposes to run MAP_RANDOM with different random samples in case when a point into the trap region. The same can be done in MAP_RS. So, we can consider randomized algorithms to be no-trap algorithms. Since MAP_Greedy [10] is a deterministic algorithm, we claim it to be trap algorithm in the general case.

Theorem 3 *The postprocessing algorithm never places removed points into the trap region of P^* . The postprocessing algorithm either improves the area of the given polygon or does not change it.*

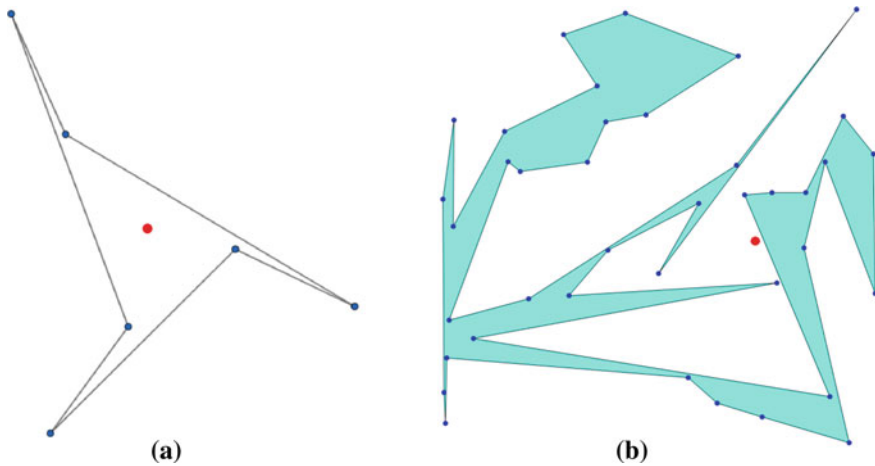


Fig. 4 Red point is inside the trap region, where **a** point inside the polygon and **b** point outside the polygon

Proof For every point there possible three distinct cases.

- (1) (+) At least one of steps 3 or 4 does not hold and the point is not removed;
- (2) (++) Point is removed and added such that old and new values of P are the same;
- (3) (+++) Point is removed and added such that old and new values of P are not the same.

In case (+), we do not remove points, therefore, correctness is held. Otherwise (cases (++) , (+++)) point is removed. In case (+++) not only correctness is held but also area is improved. If both (+) and (+++) are not met then from (+), we have guarantee of (**). That is, edge (P_{i-1}, P_{i+1}) of P^* is visible from P_i . So, removed point P_i is not in the trap region but overall area remains the same as before (case (++)).

4 Experiments

To test described algorithms and compare them with previously proposed algorithms, we have implemented MAP_DAC2, MAP_DAC, MAP_Greedy, MAP_RAND, MAP_RS, and postprocessing. We write in C++ using the SFML library for a graphical interface (Fig. 5). There are two ways of specifying the data—reading points from the file and adding or deleting points via clicking on the drawing panel.

To compare different algorithms, 30 random point sets are generated. There are three types of point sets, each type consists of ten sets of different sizes. The first two

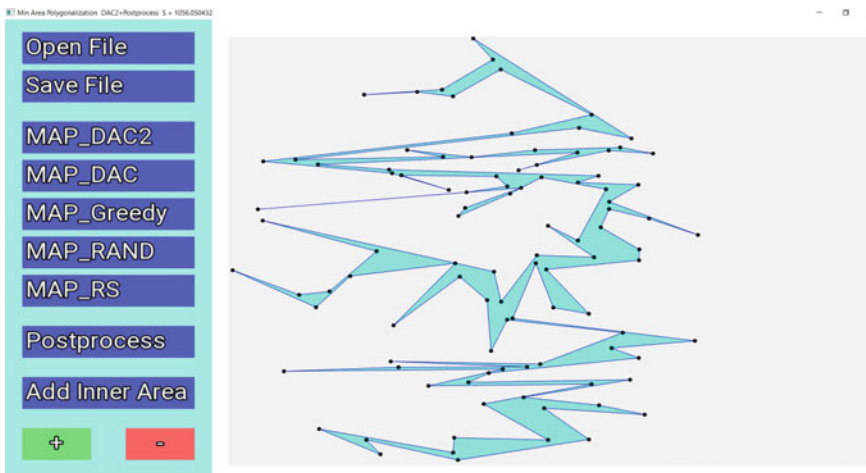


Fig. 5 Interface of the developed program

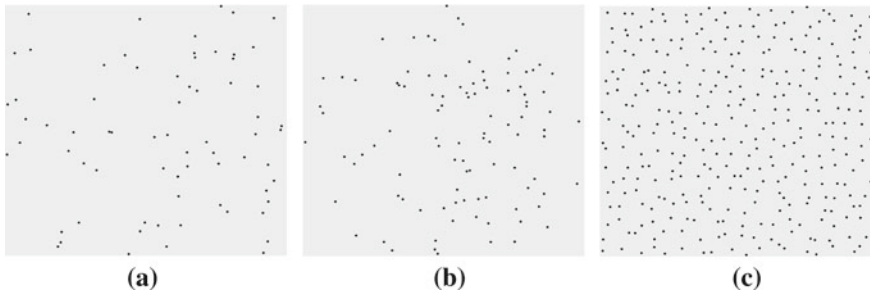


Fig. 6 **a** 1 square.txt, **b** 1 circle.txt, and **c** 1 grid.txt

types are random points in rectangle and circle, named * square.txt and * circle.txt, respectively. Points from the third type form a grid with some random shift from the exact position, named * grid.txt. Different types are shown in Fig. 6.

We ran all five algorithms on described point sets (Fig. 7a–e). For randomized algorithms MAP_RAND and MAP_RS, we set $q = 200$ trials and take the best produced polygonization. Table 1 shows the results. As MAP_Greedy was discovered to be a trap algorithm, our implementation of this algorithm ignores points that are in the trap region. In 17 out of 30 cases, the best result was computed by MAP_DAC2. In the other 13 cases, the best polygonization was computed by MAP_RS. Regardless of MAP_RS can outperform MAP_DAC2 in terms of polygon area, MAP_RS is quite slow relative to MAP_DAC2 as their complexities are $O(qn^3)$ and $O(n^2 \log(n))$, respectively.

We also ran postprocessing on output from all algorithms on all point sets (Fig. 7f). We emphasize that postprocessing allows decreasing the area of solutions up to 10–25%. Table 2 shows results after postprocessing. The result is improved by 18.7% on average. MAP_DAC2_P outputs the best result in 17 cases, MAP_RS_P in 11 cases, and MAP_RAND_P in two cases. We apply postprocessing only to the best output of MAP_RS and MAP_RAND. It is expected to get even better results if postprocessing is applied after every trial of randomized algorithms.

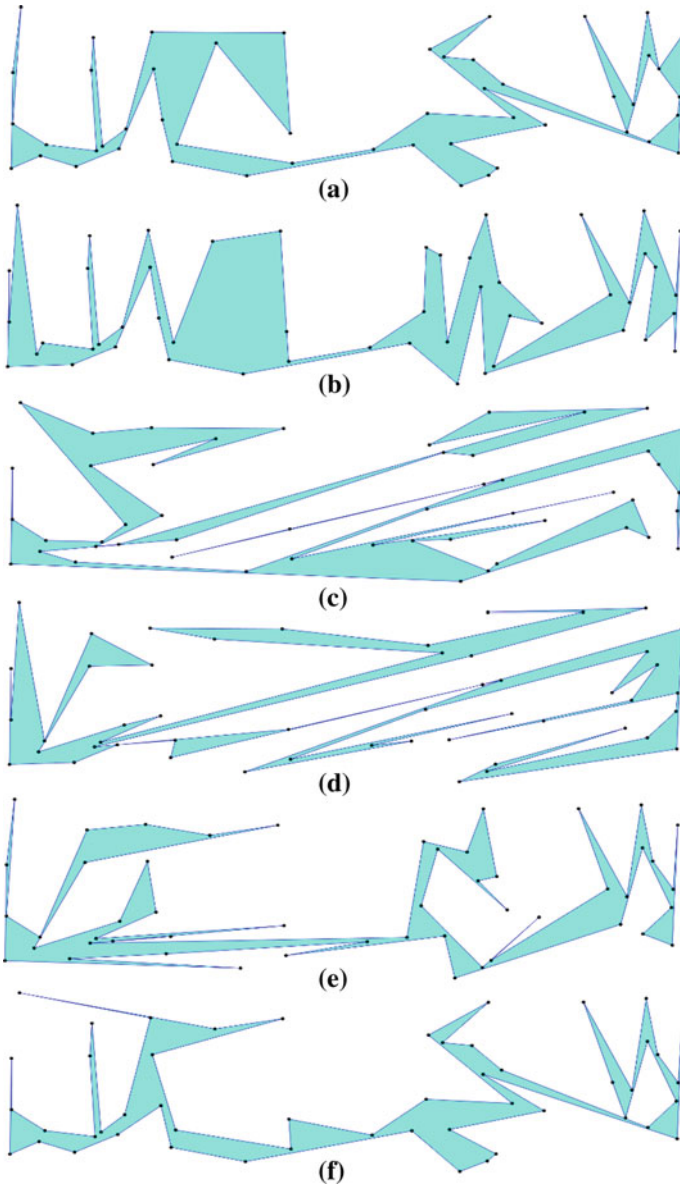


Fig.7 Polygonization of five square.txt. **a** MAP_DAC2 **b** MAP_DAC **c** MAP_Greedy **d** MAP_RAND **e** MAP_RS, and **f** MAP_DAC2 P

Table 1 Comparison of results obtained from MAP_DAC2, MAP_DAC, MAP_Greedy, MAP_RAND, and MAP_RS

File name	Number of points	MAP_DAC2	MAP_DAC	MAP_Greedy	MAP_RAND	MAP_RS
0_circle.txt	50	1546.95	2205.22	2163.97	1421.74	1337.32
0_grid.txt	225	5073.6	7552.84	5893.04	6450.58	5136.01
0_square.txt	20	1649.18	2569.88	1783.53	1347.51	1275.03
1_circle.txt	100	1407.81	2478.42	1613.12	1285.63	1245.65
1_grid.txt	324	7509.8	10,717.5	9285.85	9478.32	7556.45
1_square.txt	70	2029.32	2206.15	2844.91	1834.54	1613.95
2_circle.txt	150	1459.24	2173.04	2132.49	1606.34	1402.67
2_grid.txt	441	10,196.1	14,775.8	12,771.9	12,862.2	10,229.6
2_square.txt	220	2022.32	3182.49	2937.25*	2242.83	1869.77
3_circle.txt	200	1565.39	2702.54	1597.23	1674.86	1398.37
3_grid.txt	576	13,397.4	19,157.1	18,039.6	16,774.6	13,640.4
3_square.txt	470	1851.76	2916.31	2669.09	2294.82	1903.7
4_circle.txt	250	1534.87	2405.29	1873.09	1704.48	1466.64
4_grid.txt	729	16,209.4	22,362.3	20,766.8*	21,420.7	17,387.4
4_square.txt	820	1881.42	2910.44	2708.03	2452.73	2018.38
5_circle.txt	300	1533.45	2192.7	2027.85	1805.44	1479.12
5_grid.txt	900	19,876	28,519.2	26,701.5	26,726.3	21,612.8
5_square.txt	50	589.507	797.318	681.784	594.001	497.325
6_circle.txt	350	1441.19	2430.5	1901.1	1792.82	1475.15
6_grid.txt	330	6667.67	10,850.5	9933.22*	8969.63	7247.62
6_square.txt	140	560.736	875.54	680.984	597.184	530.764
7_circle.txt	400	1397.47	2552.57	1923.42	1809.85	1492.82
7_grid.txt	360	7818.05	12,676.4	10,598.4*	9783.51	8104.49
7_square.txt	290	564.116	933.543	741.509	675.447	561.196
8_circle.txt	450	1485.48	2288.47	1966.1	1695.7	1515.07
8_grid.txt	390	7870.21	13,452.7	11,478.4	10,362.4	8182.76
8_square.txt	500	548.995	973.862	772.515	701.326	573.593
9_circle.txt	500	1536.36	2544.76	2084.82	1819.55	1591.75
9_grid.txt	420	9318.83	13,578.1	12,283.1	11,080.6	8967.92
9_square.txt	770	561.227	945.21	826.275*	718.896	567.384

Table 2 The polygons area after refinement relative to Table 1

File name	MAP_DAC2_P	MAP_DAC_P	MAP_Greedy_P	MAP_RAND_P	MAP_RS_P
0_circle.txt	83.2882	68.8231	74.8971	95.2749	86.6654
0_grid.txt	78.4822	60.7262	93.39	83.7179	85.1955
0_square.txt	86.2036	56.9215	91.4348	100	98.5454
1_circle.txt	75.0139	53.6432	92.613	74.8542	92.1643
1_grid.txt	77.4988	60.4907	89.2939	79.3326	75.3776
1_square.txt	84.4191	81.2336	88.9499	82.773	95.9746
2_circle.txt	84.6908	68.3225	82.913	83.7697	82.2322
2_grid.txt	76.7843	59.1727	91.3193	81.1598	83.3493
2_square.txt	77.272	60.4619	89.8424*	71.8307	91.8665
3_circle.txt	82.1769	55.8258	94.2239	77.7306	85.2177
3_grid.txt	74.7865	55.3249	88.3407	80.2147	83.075
3_square.txt	79.7649	59.1584	85.9197	74.7312	87.1332
4_circle.txt	77.9657	62.6643	92.1431	82.6544	83.4875
4_grid.txt	81.1176	62.3924	83.0463*	76.909	81.0698
4_square.txt	79.6647	59.2644	83.1866	76.7587	77.6908
5_circle.txt	81.8064	63.7366	84.6136	84.7903	78.679
5_grid.txt	78.9024	59.4176	85.437	80.5203	79.2287
5_square.txt	82.5412	70.5668	93.033	85.7604	82.9386
6_circle.txt	85.8389	60.589	85.0262	77.9972	83.1042
6_grid.txt	77.2097	66.6462	87.6776*	78.2857	85.3425
6_square.txt	83.7059	64.268	78.2077	85.5597	86.3094
7_circle.txt	80.6722	62.6201	82.9018	77.7518	80.1766
7_grid.txt	76.3475	65.5676	83.5795*	80.737	81.5681
7_square.txt	82.0708	57.3291	86.3043	83.8413	83.6514
8_circle.txt	81.367	63.3459	79.5047	72.7502	79.4567
8_grid.txt	75.0734	66.7353	86.8286	77.9401	80.2878
8_square.txt	80.2742	56.8465	82.3947	80.5178	85.0199
9_circle.txt	77.5403	57.7584	89.2339	74.1333	82.8734
9_grid.txt	81.2925	64.3985	80.572	79.5746	81.9839
9_square.txt	77.0793	61.8968	81.2972*	74.8826	82.044

5 Conclusion

In this paper, we present a modification of some minimum area polygon search algorithms on any randomly generated set of points. In particular, we propose modifications of MAP_DAC2 algorithm using the divide-and-conquer technique for constructing an approximate solution of the MAP problem. The algorithm computes slightly more cases compared to the unmodified version but shows significant improvement in the polygon area. The complexity of the algorithm is $O(n^2 \log(n))$ using $O(n)$ memory. We also propose the algorithm for postprocessing for others algorithms that improves local optimality via local permutations of sequential point selection. The complexity of postprocessing is $O(n^2)$ using $O(n)$ memory. We conduct experimental study to compare existing algorithms. MAP_DAC2 turns out to be competitive in terms of polygonization area while working much faster than the best of existing algorithms. We also compare results before and after postprocessing. Postprocessing improved best polygonizations by 18% on average which is a new qualitative result. During the research of existing and new MAP algorithms, we considered three tactics—producing new polygonization, postprocessing of polygonization, and constructing new better polygonizations via combining other polygonizations. Producing new polygonization is relatively well studied in the literature. We propose the first MAP postprocessing technique in this paper. Polygon combining is left for future research. We believe that all three tactics deserve further investigation.

References

1. Worboys MF, Duckham M (2006) Monitoring qualitative spatiotemporal change for geosensor networks. *Int J Geogr Inform Sci* 20(10):1087–1108
2. Galton A (2005) Dynamic collectives and their collective dynamics. *COSIT, LNCS 3693*:300–315
3. Miller HJ, Han J (eds) *Geographic data mining and knowledge discovery*. CRC Press
4. Galton A, Duckham M (2006) What is the region occupied by a set of points? *GIScience 2006, LNCS 4197*:81–98
5. Fekete SP (2000) *On simple polygonalizations with optimal area*. *Discrete and computational geometry*. Springer, Berlin
6. Velho L (1996) Simple and efficient polygonization of implicit surfaces. *J Graph Tools* 1(2):5–24
7. Zhang H, Zhao Q, Yu L, Generation of simple polygons from ordered points using an iterative insertion algorithm. <https://doi.org/10.1371/journal.pone.0230342>
8. Fekete S, Friedrichs S, Hemmer M, Papenberg M, Schmidt A, Troegel J (2015) Area- and boundary-optimal polygonalization of planar point sets. In: *EuroCG 2015, Ljubljana, Slovenia, March 16–18*, pp 133–136
9. Fekete S, Friedrichs S, Hemmer M, Papenberg M, Schmidt A, Troegel J (2020) Computing area-optimal simple polygonalizations. In: *36th European workshop on computational geometry, Würzburg, Germany, March 16–18*, pp 20:1–20:8
10. Tereshchenko V, Muravitskiy V (2011) Constructing a simple polygonalizations. *J World Acad Sci Eng Technol* 77:668–671

11. Osiponok M, Tereshchenko V (2019) The “Divide and Conquer” technique to solve the minimum area polygonalization problem. In: 2019 IEEE international conference on advanced trends in information theory, proceedings, pp 336–339
12. Peethambaran J, Parakkat AD, Muthuganapathy R (2016) An empirical study on randomized optimal area polygonization of planar point sets. *J Exp Algorithmics* 21(1):A:1-A:25
13. Taranilla MT, Gagliardi EO, Peñalver GH (2011) Approaching minimum area polygonization. *Universidad Nacional de La Plata*, pp 161–170
14. Joe B, Simpson RB (1987) Corrections to lee’s visibility polygon algorithm. *BIT Numer Math* 27(4):458–473

A Super Ensembled and Traditional Models for the Prediction of Rainfall: An Experimental Evaluation of DT Versus DDT Versus RF



Sheikh Amir Fayaz, Majid Zaman, and Muheet Ahmed Butt

Abstract The main purpose of the current study is to use three traditional and ensemble machine learning approaches namely Decision tree (DT), Distributed Decision tree (DDT) and Random Forest (RF), and a classification tree model- Iterative Dicotomizer 3 (ID3) to predict the rainfall based on the historical data. A hard-voting classifier was used to check the accuracy performance in DDT and RF models. In this study, comparative performance of all the three techniques was assigned, and overall, all the three techniques perform reasonably well. Furthermore, it was observed that the DT model on the original dataset produces an accuracy of 81.70% followed by an accuracy of 81.41% in the case of the RF model. The overall performance in case of the DDT model got reduced to 78.46%. Thus, the obtained results predict that DT outperforms all other approaches with the highest accuracy measure and high susceptibility rate in rainfall prediction.

Keywords Decision tree · Distributed decision tree · Random forest · Hard-voting · Geographical data

1 Introduction

Weather forecasting is considered as one of the approaches which are used to check the state of the atmosphere in the future at a specified location. In the early century when there was no technology that can predict whether the man first looked to the skies, he was comforted by an ever-changing atmosphere that defined his understanding. To explain whether he did the best he could with what he had, various observations from the foundation form whether proverbs are the man's first attempts to predict the weather. A surprising number of them have proved to be correct. After

S. A. Fayaz · M. A. Butt
Department of Computer Sciences, University of Kashmir, Srinagar, India

M. Zaman (✉)
Directorate of IT & SS, University of Kashmir, Srinagar, India
e-mail: zamanmajid@gmail.com

some improvements in technology, the properties of the atmosphere became measurable quantities, and meteorology (the study of the atmosphere) became science. With the progress in the physical sciences, the understanding of the atmosphere and weather increased steadily. With time many technological developments have made modern meteorology one of the most complex and far-reaching of all the physical sciences [1]. Today, weather prediction has become increasingly accurate in answering a question that has challenged man for centuries but accurate and timely weather forecasting still remains a challenge for scientists. This study of weather and weather forecasting is one subject that almost everyone is interested in. Thus, in this era where machine learning techniques are used in every field, weather forecasting can also be predicted using various suitable machine learning approaches [2–5]. In this paper, we have used traditional and ensemble techniques which includes Decision tree, Random forests, and Distributed Decision Trees for the prediction of rainfall based on the weather data of Kashmir province at three different locations. Various performance parameters are evaluated in which precision, recall; Cohen’s Kappa, Accuracy, and error are calculated [6–9]. This paper is structured as follows: this section (Sect. 1) gives the introduction about weather forecasting and briefly describes the three techniques (Decision tree, Random Forest, and Distributed Decision tree) which are implemented in this paper. Section 2 describes the review of literature in using various machine learning algorithms. Section 3 briefly analyzes the data on which the experiment has been carried out. Section 4 describes the methodology and the flow process of the technique used in this study. Section 5 gives the experimental implementation of Decision trees (DT), Random forests (RF), and Distributed Decision trees (DDTs). In Sect. 6 a voting strategy has been defined followed that Sect. 7 checks the performance of the algorithms and its brief comparative analysis. In last, Sect. 8 concludes the paper.

1.1 Decision Trees (DT)

A decision tree is a type of classification algorithm which comes under the supervised learning technique. It is one of the basic supervised learning algorithms that are used in the tree-structured classification and regression problem. A decision tree is a tree-based classifier where data is classified based on a tree structure. A decision tree is represented by a tree-shaped structure that includes decision nodes (leaf nodes) and parent nodes (internal nodes) [10–12]. Each node is split into a number of possible branches which determines the course of action. There are many approaches in which the nodes of the decision tree are splitted. The selection of the splitting criteria is mainly based on the splitting measures. The splitting criteria include univariate splitting and multivariate splitting criteria. The univariate splitting criteria internally include impurity-based splitting criteria, Information Gain, GINI index, DKM criteria, Gain ratio, Binary Criteria, Twoing criteria, orthogonal (ORT) Criterion, Kolmogorov–Smirnov Criterion, etc. [13–15]. In this study, the implementation of the decision tree is based on the information gain criteria and GINI Index

criteria. In the case of information gain criteria, the attribute which has the highest information gain is treated as the splitting node. This repetitive process stops until all the nodes of the tree are processed. The formula to calculate the information gain of the node is (1):

$$\text{informationGain}(a_i, S) = \text{Entropy}(y, S) - \sum_{v,j}^i \frac{|\sigma a_i = v_{i,j} S|}{|S|} \cdot \text{Entropy}(y, \sigma a_i = v_{i,j} S) \quad (1)$$

where:

$$\text{Entropy}(y, S) = \sum_{C_j \in \text{dom}(y)} - \frac{|\sigma_y = C_j S|}{|S|} \cdot \log_2 \frac{|\sigma_y = C_j S|}{|S|} \quad (2)$$

Consequently, the Gini index works on the impurity-based approach and the node which has minimum impurity will be chosen as the splitting node. The formula to calculate the GINI index of the node is shown (3):

$$\text{Gini}(y, S) = 1 - \sum_{C_j \in \text{dom}(y)} \left(\frac{|\sigma_y = C_j S|}{|S|} \right)^2 \quad (3)$$

Here, in this paper the splitting is based on information gain where the decision tree is constructed on the historical geographical data of Kashmir Province.

1.2 Random Forest (RF)

Random forest is a collection of multiple random decision trees and it's much less sensitive to the training data as compared to decision trees. This first step to build random forest is to build new datasets from our original data. The process of creating new datasets from the original dataset is called bootstrapping. The generalized approach and working of random forest are shown (Fig. 1) where the original dataset is splitted into 3 decision trees, each having different set of features [16, 17].

The prediction in case of random forest depends in the majority hard voting approach where each individual subtree predictions are analyzed and the resultant output prediction will be the first output of the random forest. This process of combining results from multiple models is called aggregation. So, in random forest we first perform bootstrapping and then aggregation in combination and this process is called bagging [18].

In random forest classifier two random processes are used namely: bootstrapping and random feature selection. The motivation behind bootstrapping and feature

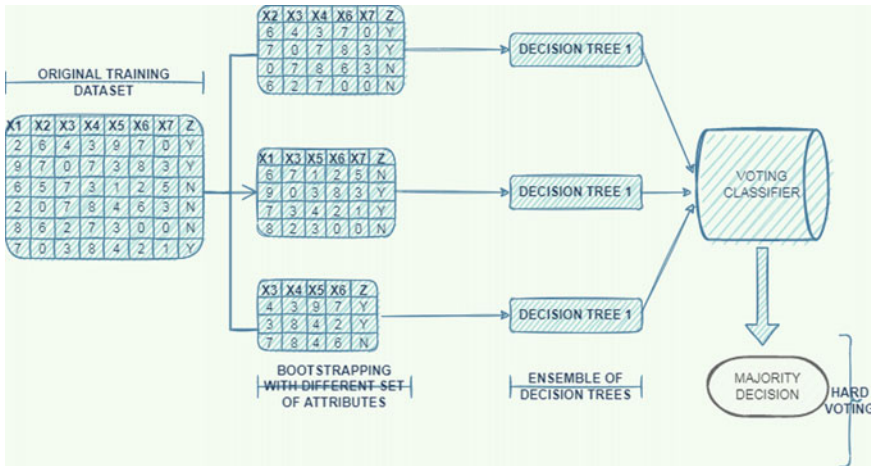


Fig. 1 Random forest model

selection is that the same data has not been used for every tree. By this the model becomes less sensitive to the original training data. The random forest selection helps to reduce correlation between the trees [19, 20].

1.3 Distributed Decision Tree (DDT)

Distributed decision tree is a supervised classification algorithm where the original dataset is divided into multiple subset datasets based on the particular attribute. The working of distributed decision tree is same as random forest where bootstrapping and aggregation processes takes place [6, 21]. In Distributed decision tree approach the bootstrapping is done based on the same set of attributes. i.e., if the original dataset contains 5 attributes, then the resultant subsets contain the same set of attributes for the operation. The generalized and graphical approach and working of Distributed Decision tree is shown (Fig. 2) where the original dataset is splitted into 3 decision trees, each having same set of features [22, 23].

Suppose if there are 1000 entries present in the original dataset and the data is splitted into 5 partitions i.e., 5 decision trees will be created and if the target attribute is binary classified [0,1] and every decision tree either predicts 0 or 1. Then the final output will be the majority count of 0 or 1 from all the sub trees.

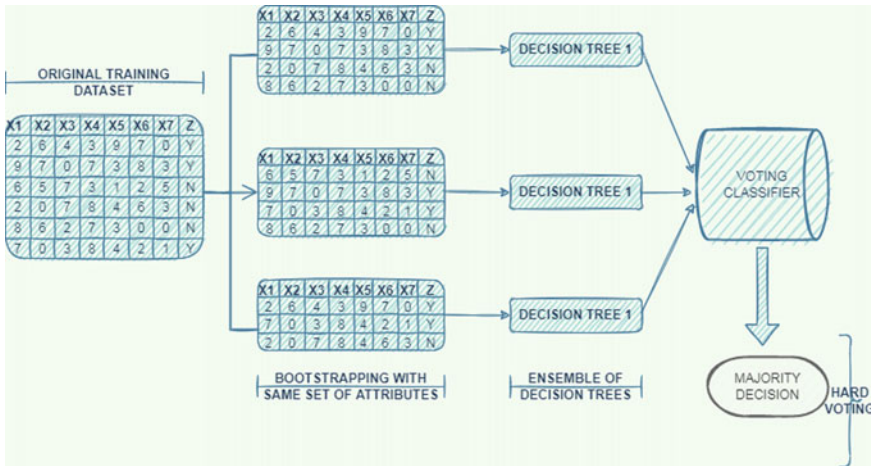


Fig. 2 Distributed decision tree model

2 State of Knowledge

In this study our main focus will be on previous research based on some traditional and ensemble machine learning approaches which are used in the rainfall prediction. These approaches include decision trees (DT), random forests (RF) and distributed decision trees (DDT) [24].

Based on the historical data obtained from Tamil Nadu government of Tamil Nadu region in India, including 7 attributes used in prediction, Aswini et al. [25] proposed a statistical technique to predict the rainfall. The algorithms used in the implementation were Naïve Bayes, KNN, Decision Trees, Fuzzy Logic and Neural Network. The experimental results showed that the Decision trees and KNN are performing better results in terms of accuracy. Hemalatha [26] implements decision tree technique using C4.5 and Iterative Dicotomizer 3 (ID3) algorithm on 20–30 instances of previous data and Petre [27] implements Decision tree technique using CART algorithm on 48 instances of data available from 2002 2005 and Ji et al. [28] implements Decision tree technique using C4.5, CART algorithms on previous data available consisting of 26,280 instances. It was observed that the Decision tree in individual performs better accuracy of 93–99% when compared with other algorithms [29, 30].

A distributed representation was proposed by Irsoy et al. [31] in which multiple paths are combined that can be crossed at the same time using multiple layer perceptron. In this study distributed decision trees and budding trees were compared and the results depict that in classification and regression techniques distributed decision trees perform much better than in case of budding trees. This study also concludes with that, the distributed trees can be considered as alternative layers of perceptron's

for deep learning by implementing the hierarchical distributed representation of the input at the various levels.

Based on the accuracies and the algorithms used (DT & Random forests), the comparative performances on the meteorological data are summarized in Table 1.

3 Data Collection and Analysis

In this paper the data used is of Kashmir region of India and has been collected from the Indian Meteorological department Pune. There are 9 parameters present in the dataset including humidity at 12 AM (humid12) and 3 PM (humid3), Maximum Temperature (tmax), Minimum Temperature (tmin), Station ID (station_id), date (dt), month (mnth), year and the rainfall (rfall) as the target parameter. These parameters are present in two comma separated files which consists of around 12,000 instances of humidity values measured at two-time intervals every day at 12 AM and 3 PM and 6000 instances of rest of the above defined parameters every day. This data was collected from the year 2012–2017 for all the three stations.

The two files are then integrated into a single file where all the missing values were removed and all the inconsistencies were resolved and cleaned. The below snapshot (Table 2) shows the integrated file in which all the necessary attributes are merged for all the three stations of the Kashmir province [37, 38].

Thus the same continuous dataset has been discretized [1] using GINI index and was used for the implementation purposes. The snapshot of the final labeled data is shown (Table 3).

4 Methodology

In this study, we present how the model is implemented on the historical geographical data using three traditional and ensemble approaches which includes Decision tree (DT), Distributed Decision tree (DDT) and Random Forest (RF). In these algorithms the datasets has been divided into (70–30) % training and test set respectively. This splitting of the data was implemented in python using sklearn split model as shown in below code snippet Fig. 3:

Since, in this study we have used three different approaches to check the accuracy on the same set of data and each approach follows the same methodology which is defined below.

Table 1 Statistical measures of ensembled and traditional algorithms for rainfall prediction

Author, Year	Region	Attributes used	Data	Techniques used	Accuracy	References
Aswini R., 2018	Tamil Nadu, India	7 different rainfall parameters were used	26,280 samples of Data	Decision tree	Decision Tree (93–99)% accurate	[25]
P. Hemalatha, 2013	Not mentioned	Different rainfall parameters	20–30 instances of historical data	Decision tree (ID3, C4.5)	DT performs better	[26]
Elia Georgiana Petre, 2009	Not mentioned	Basic rainfall parameters	48 instances (2002–2005)	Decision tree (CART)	DT performs better results	[27]
Soo-Yeon Ji, 2012	Not mentioned	Not mentioned	Historical data with 26,280 instances	Decision tree (CART & C 4.5)	DT performs better results	[28]
Suhaila Zainudin, 2016	Malaysia	Temperature, rainfall, relative humidity, water level	Historical data from (2010–2014)	Decision tree, random forest, KNN etc	DT and random forest perform better as compared to other techniques	[32]
A. Geetha, 2014	Trivandrum, India	20 parameters out of which only 12 were used	(2013–2014)	Decision tree	80.67%	[33]
Zeyi Chao, 2018	Wuhan area, China	General	General	Comparative analysis between MEMS sensors/LSTM/BPNN/RF/ARMA/SVM	Satisfactory results in which Prediction based sensors and RF perform good results	[34]

(continued)

Table 1 (continued)

Author, Year	Region	Attributes used	Data	Techniques used	Accuracy	References
Razeef Mohd, 2018	Srinagar, J&K, India	9 parameters out of which only 5 were used	540 instances (2015–2016)	Comparative analysis of decision tree, J48, random forest, Naïve Bayes	Random Forests performs good results with an accuracy of 87.76%	[35]
Sheikh Amir, 2021	J&K, India	5 parameters	5900 instances (2012–2017)	Distributed decision tree (DDT)	Accuracy 78.7%	[36]

Table 2 Integrated data

Station_id	Year	Mnth	Dt	Tmax	Tmin	Rfall	Humid3	Humid12
42026	2012	2	5	5.5	-8	0	100	100
42026	2013	2	8	5.4	-7.6	0	80	100
42027	2012	6	4	4.2	-8.1	0.6	96	90
42044	2014	5	7	6.5	-4.3	1.2	86	95
42024	2014	6	7	5.3	-6.5	1.8	97	100
42026	2012	5	6	5.7	-8.2	1.02	88	97
42026	2013	4	3	4.5	-4.6	0.8	75	100
42026	2014	6	7	5.8	-7.4	0.8	77	86
42027	2017	6	8	1.6	-4.5	2.6	94	98
42027	2013	6	8	0.3	-6.5	1.65	95	93
42024	2014	6	7	5.3	-6.5	1.8	97	100
42026	2014	6	7	5.8	-7.4	0.8	77	86
42026	2012	2	5	5.5	-8	0	100	100
42027	2012	6	4	4.2	-8.1	0.6	96	90
42026	2013	5	4	3.2	-2.5	0	65	86

Table 3 Processed and labeled data

Season	Tmax	Tmin	Humid3	Humid12	Rfall
Summer	H2	L1	U2	T2	N
Summer	H2	L2	U1	T2	N
Winter	H1	L2	U1	T1	Y
Autumn	H2	L1	U1	T2	Y
Spring	H1	L2	U2	T1	N
Autumn	H1	L2	U2	T1	Y
Summer	H2	L2	U1	T2	Y
Winter	H1	L1	U1	T1	N
Winter	H2	L1	U2	T2	Y
Spring	H2	L2	U2	T2	Y
Autumn	H1	L1	U2	T1	N
Summer	H2	L2	U2	T2	Y
Summer	H1	L2	U1	T1	N
Autumn	H2	L1	U2	T2	Y
Winter	H1	L2	U1	T1	N

```
In [1]: import pandas as pd
df = pd.read_csv('Data.csv')
df.head()
from sklearn.model_selection import train_test_split
X_train, X_test, Y_train, Y_test = train_test_split(X,Y, test_size=0.3)
```

Fig. 3 Train test split model

4.1 Methodology: Workflow Process

All the classifiers are set on the particular training dataset and test sets. The result obtained from the training set will be tested on the test dataset and the performance will be analyzed. The flow process is shown (Fig. 4) in which the partitioning of the data takes place into test and training data. An iterative Dicotomizer ID3 algorithm has been used in this study. This is the supervised classification approach in which the preprocessed dataset will be classified based on the parameters/attributes to check the overall performance of the prediction [39].

This method involves the analytical and theoretical implementation of the various splitting thresholds which include information gain, entropy and GINI coefficients. Varying decisions and PMML rules will be generated and based on these generated rules we can predict the future weather for various regions.

This flow process (Fig. 4) is the base for all the three approaches i.e. for Decision tree, Random forests and Distributed Decision Tree in which the data is divided into training and test ratio. In case of Random Forest (RF) and Distributed Decision Tree

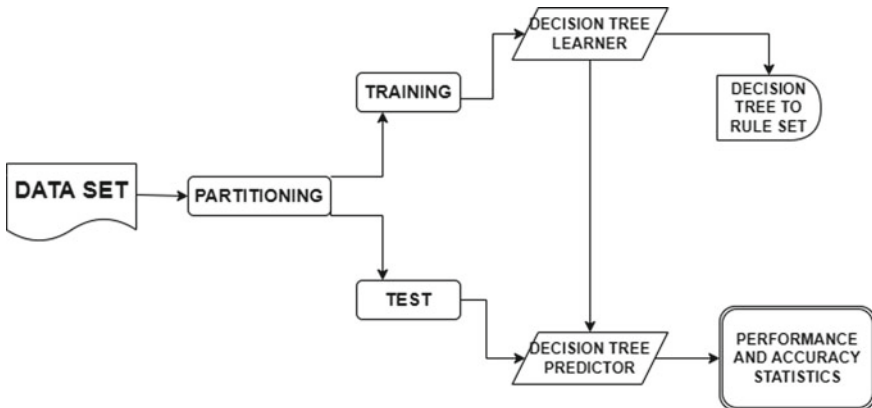


Fig. 4 Workflow process

(DDT), a mechanism is used called as “Hold the strategy” mechanism in which a hard voting approach is used to calculate the overall accuracy of the model [40, 41, 42].

5 Implementation and Experimental Evaluation

In this paper, an Iterative Dicotomizer algorithm (ID3) was implemented in the construction of decision tree. The implementation was carried out on the historical geographical data of Kashmir province. The dataset was divided into 70% as training and 30% as testing ratio. We have used the information gain for the splitting criteria and the splitting node has been chosen which has maximum information gain. Also in construction of Distributed Decision tree the dataset was divided into 3 subsets based on the station Id. After the subdivision of the dataset the same approach of ID3 has been carried out.

Furthermore, after the implementation process, the accuracy measure was calculated and the overall performance of the original decision tree was calculated and PMML rules were also generated in the decision tree implementation. Since, after the implementation of distributed decision trees and random forests using same approach was performed with the hard voting strategy in which the majority of the predicted values from the individual decision trees have been implemented and it has been taken as the final prediction value.

6 Voting Strategy

A Voting Classifier is basically a Machine Learning (ML) model that trains on an ensemble of numerous models and predicts the overall output based on their majority values given by the individual models. It is a wrapper for set of different ones that are trained and valuated in parallel in order to exploit the different individualities of each algorithm. We can train data using different set or same set of algorithms and ensemble them to predict the final output. According to the hard voting strategy the final output is based on the majority value predicted by the individual models [36, 43].

The code snippet shown below implemented in java defines the hard voting concept where the subsets of the original data are provided as the inputs to the classifier. The count variable is used as a container which counts the highest number of votes based on the different string values and accordingly the databases values are updated.

Code Snippet – Voting Classifier

```

while (rs1.next ())
{
int rno = rs1.getInt ("rno");
String d26 = rs1.getString ("d26");
String d27 = rs1.getString ("d27");
String d44 = rs1.getString ("d44");
Statement st6 = con.createStatement ();
System.out.println (rno+" "+d26+" "+d27+" "+d44);
if (d26.equals (d27))
{
cnt1++;
String fqry="update ftest_set set rf='"+d26+"' where rno='"+rno+"';";
St6.executeUpdate (fqry);
}
else if (d26.equals (d44))
{
cnt2++;
String fqry="update ftest_set set rf='"+d26+"' where rno='"+rno+"';";
St6.executeUpdate (fqry);
}
if (d44.equals (d27))
{
cnt3++;
String fqry="update ftest_set set rf='"+d44+"' where rno='"+rno+"';";
St6.executeUpdate (fqry);
}
}

```

The output of the above code implemented in which we combine three classifiers that classify a training sample as Y or N is shown (Table 4).

Here rf field is the outcome of hard voting strategy which checks the values from field d26, d27, d44 and outputs the majority value and crfall field denotes the original prediction what it was actually. In this study we have implemented hard voting strategy in Random Forest and Distributed decision tree to check the overall accuracy based on the same set of data.

Table 4 Hard voting strategy output

R_No.	Season	Ctmax	Ctmin	Chumid12	Chumid3	Crfall	D26	D27	D44	rf
1	Spring	H1	L1	T2	U1	Y	N	N	Y	N
2	Winter	H2	L2	T1	U1	Y	Y	N	Y	Y
3	Autumn	H2	L2	T1	U1	Y	Y	N	Y	Y
4	Spring	H1	L2	T1	U2	Y	Y	Y	Y	Y
5	summer	H2	L2	T2	U1	Y	N	N	Y	N
6	winter	H1	L2	T2	U1	Y	Y	N	Y	Y
7	Spring	H2	L2	T1	U1	Y	Y	N	Y	Y
8	summer	H2	L1	T2	U1	Y	Y	N	N	N

7 Performance Evaluation

In this study, the researchers employed state-of-the-art technologies on geographical dataset with the prime purpose was to check the algorithm with the best overall performance and accuracy. After multiple partitioning of the data based on the station id's (zones of the Kashmir province), hard voting technique was implemented and it was observed that there was no substantial improvement in prediction accuracy in predicting the class labels when distributed decision tree was taken into consideration. Furthermore, after checking the accuracy measure of the random forest it was observed that there is some improvement in the performance as compared to Distributed decision tree but the accuracy measure still remains below the accuracy measure of the original decision tree.

Table 5 shows a snapshot of results which includes accuracy, precision, recall values and many other calculations. The overall accuracy of the original tree is 81.70% in predicting the outcome class and approximately same accuracy is shown by 2 zones (zone 42027, 42044). The accuracy of zone 42026 is 70.29% which results in the overall degradation of the performance in case of Distributed decision tree and Random Forest. Moreover, after implementing the distributed decision tree and random forest the accuracy is 78.46% and 81.41% respectively and it was observed that the decision tree and random forest shows better results and performance as compared to Distributed Decision tree technique.

Table 5 Prediction performance of original decision tree, different zones, distributed decision tree and random forest

Classifier (ID3)	Original Tree	Zone 42026	Zone 42027	Zone 42044	Distributed Decision Tree (DDT)	Random Forest (RF)
Test	1786 (30%)	579 (30%)	612 (30%)	597 (30%)	1786 (30%)	1786 (30%)
Training	4165 (70%)	1347 (70%)	1424 (70%)	1392 (70%)	4165 (70%)	4165 (70%)
Correctly Classified	1459	407	497	485	1401	1441
Wrong Classified	327	172	115	112	385	345
Accuracy	81.70%	70.29%	81.20%	81.23%	78.46%	81.41%
Error	18.30%	29.70%	18.79%	18.76%	21.54%	18.59%
Cohen Kappa	0.502	0.343	0.436	0.494	0.461	0.494
Precision	79.00%	79.80%	70.10%	82.10%	–	82.42%
Recall	94.60%	94.10%	95.80%	94.00%	–	94.32%
F-measure	88.00%	80.30%	88.40%	87.50%	–	87.96%
Specificity	94.60%	94.10%	95.80%	94.00%	–	78.18%

The columnar (Fig. 5) and graphical (Fig. 6) representation of geographical data of Kashmir province is shown above, which helps in easy visualization of the results performed. Other methods are also very efficient method but they need a large portion of training data to train in order to predict very small portion of test data.

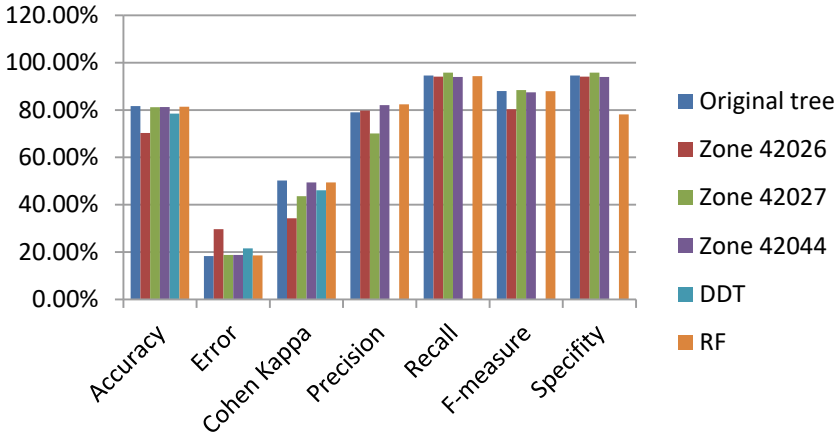


Fig. 5 Columnar graph: defines the accuracy statistics of each algorithm

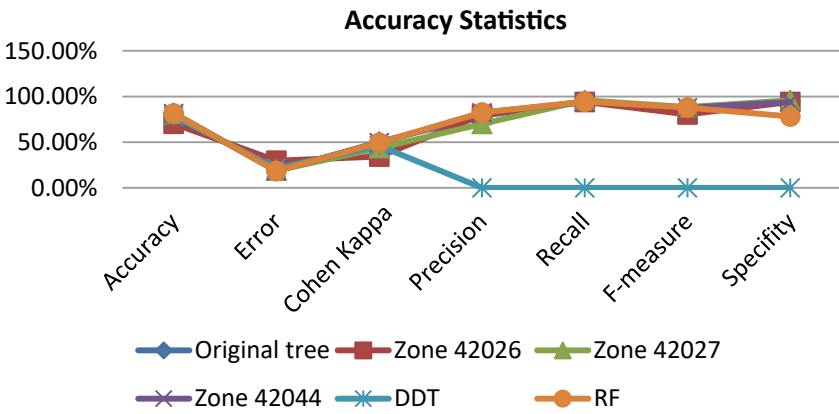


Fig. 6 Line graph: defines the accuracy statistics of each algorithm

8 Conclusion

In this study we have implemented three different techniques including DT, DDT and RF on the historical geographical data of Kashmir province. All the three implementations are systematically analyzed and are compared for the prediction of rainfall. The final result depicts that the Decision tree and Random Forest shows better accuracy and other measures like precision, recall etc. Table 5 shows the snapshot of results obtained which includes accuracy, precision, recall and many other calculations. The overall accuracies produced by Distributed Decision tree, Random Forest and Decision tree are 78.46%, 81.41% and 81.70% respectively. By this we can conclude that Distributed Decision tree doesn't show any substantial improvement in predicting the class labels whereas, DT and RF shows head-to-head accuracy and prediction and thus can be considered as the promising techniques for the prediction of rainfall in temperate zones like Kashmir province.

References

1. Zaman M, Kaul S, Ahmed M (2020) Analytical comparison between the information gain and Gini index using historical geographical data. *Int J Adv Comput Sci Appl (IJACSA)* 11(5):429–440
2. Zamani NW, Khairi SSM (2018, October) A comparative study on data mining techniques for rainfall prediction in Subang. In: *AIP conference proceedings*, vol 2013, No 1. AIP Publishing LLC, p. 020042
3. Safavin SR, Langrebe D (1991) A survey of decision tree classifier methodology. *IEEE Trans Syst Man Cybern* 21(3):660–674
4. Kapás Z, Lefkovičs L, Szilágyi L (2016) Automatic detection and segmentation of brain tumor using random forest approach. In: *Modeling decisions for artificial intelligence*, Springer, Berlin, pp 301–312
5. Rokach L, Maimon O (2008) *Data mining with decision trees: theory and applications*. World Scientific Pub Co Inc.
6. Ashraf M, Zaman M, Ahmed M (2020) An intelligent prediction system for educational data mining based on ensemble and filtering approaches. *Proc Comput Sci* 167:1471–1483
7. Zaman M, Quadri SMK, Butt MA (2012) Information translation: a practitioners approach. In: *Proceedings of the World Congress on engineering and computer science*, vol 1
8. Quinlan J (1999) Simplifying decision trees. *Int J Hum Comput Stud* 1999(51):497–491
9. Zainudin S, Jasim DS, Bakar AA (2016) Comparative analysis of data mining techniques for Malaysian rainfall prediction. *Int J Adv Sci Eng Inform Technol* 6(6):1148–1153
10. Misra S, Li H (2019) Noninvasive fracture characterization based on the classification of sonic wave travel times. In: *Machine learning for subsurface characterization*, pp 243–287
11. Ashraf M, Ahmad SM, Ganai NA, Shah RA, Zaman M, Khan SA, Shah AA (2020) Prediction of cardiovascular disease through cutting-edge deep learning technologies: an empirical study based on Tensorflow, Pytorch and Keras. In: *International conference on innovative computing and communications*. Springer, Singapore, pp 239–255
12. Sarapardeh AH, Larestani A, Menad NA, Hajirezaie S (2020) *Applications of artificial intelligence techniques in the petroleum industry*. Gulf Professional Publishing
13. Shuja M, Mittal S, Zaman M (2020) Effective prediction of type II diabetes mellitus using data mining classifiers and SMOTE. In: *Advances in computing and intelligent systems*. Springer, Singapore, pp 195–211

14. Zaman EM, Quadri SMK, Butt EMA (2012) Information integration for heterogeneous data sources. *IOSR J Eng* 2(4):640–643
15. Butt EMA, Quadri SMK, Zaman EM (2012) Star schema implementation for automation of examination records. In: Proceedings of the international conference on frontiers in education: computer science and computer engineering (FECS), p 1. The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp)
16. Chen J, Wang T, Abbey R, Pinget J (2016) A distributed decision tree algorithm and its implementation on big data platforms. In: 2016 IEEE international conference on data science and advanced analytics (DSAA). IEEE, New York, pp 752–761
17. Desai A, Chaudhary S (2016) Distributed decision tree. In: Proceedings of the 9th Annual ACM India conference, pp 43–50
18. Bhaduri K, Wolff R, Giannella C, Kargupta H (2008) Distributed decision-tree induction in peer-to-peer systems. *Stat Anal Data Min: ASA Data Sci J* 1(2):85–103
19. Altaf I, Butt MA, Zaman M (2021) A pragmatic comparison of supervised machine learning classifiers for disease diagnosis. In: 2021 Third international conference on inventive research in computing applications (ICIRCA). IEEE, New York, pp 1515–1520
20. Altaf I, Butt MA, Zaman M (2022) Disease detection and prediction using the liver function test data: a review of machine learning algorithms. In: International conference on innovative computing and communications. Springer, Singapore, pp 785–800
21. Ashraf M, Zaman M, Ahmed M (2019) To ameliorate classification accuracy using ensemble vote approach and base classifiers. In: Emerging technologies in data mining and information security. Springer, Singapore, pp 321–334
22. Zaman M, Butt MA (2012) Information translation: a practitioners approach. In: World Congress on engineering and computer science (WCECS), San Francisco, USA. October, 2012
23. Butt MA, Quadri SMK, Zaman M (2012) Data warehouse implementation of examination databases. *Int J Comput Appl* 44(5):18–23
24. Fayaz SA, Altaf I, Khan AN, Wani ZH (2019) A possible solution to grid security issue using authentication: an overview. *J Web Eng Technol* 5(3):10–14
25. Aswini R, Kamali D, Jayalakshmi S, Rajesh R (2018) Predicting rainfall and forecast weather sensitivity using data mining techniques. *Int J Pure Appl Math* 119(14):843–847 ISSN: 1314-3395 (on-line version) <http://www.ijpam.eu> Special Issue
26. Hemalatha P (2013) Implementation of data mining techniques for weather report guidance for ships using global positioning system. *Int J Comput Eng Res* 3(3)
27. Petre EG, A decision tree for weather prediction. *BULETINUL Universităţii Petrol – Gaze din Ploieşti*, vol LXI No. 1/2009 77 - 82 Seria Matematică - Informatică – Fizică
28. Ji S-Y, Sharma S, Yu B, Jeong DH (2012) Designing a rule-based hourly rainfall prediction model. In: 2012 IEEE 13th international conference on data analysis information reuse and integration (IRI); August 2012
29. Ashraf M, Zaman M, Ahmed M (2018) Performance analysis and different subject combinations: an empirical and analytical discourse of educational data mining. In: 2018 8th International conference on cloud computing, data science & engineering (confluence). IEEE, New York, pp 287–292
30. Mohd R, Butt MA, Baba MZ (2020) GWLM–NARX. *Data technologies and applications*
31. Irsoy O, Alpaydin E (2014) Distributed decision trees. arXiv preprint arXiv: 1412.6388
32. Zainudin S, Jasim DS, Bakar AA (2016) Comparative analysis of data mining techniques for Malaysian rainfall prediction. *Int J Adv Sci Eng Inform Technol* 6(6) ISSN: 2088-5334
33. Geetha A, Nasira GM (2014, December) Data mining for meteorological applications: decision trees for modeling rainfall prediction. In: 2014 IEEE international conference on computational intelligence and computing research. IEEE, New York, pp 1–4
34. Chao Z, Pu F, Yin Y, Han B, Chen X (2018) Research on real-time local rainfall prediction based on MEMS sensors. *Hindawi J Sens* 2018, Article ID 6184713, 9 p. <https://doi.org/10.1155/2018/6184713>

35. Mohd R, Butt MA, Baba MZ (2018) SALM-NARX: Self adaptive LM-based NARX model for the prediction of rainfall. In: 2018 2nd international conference on I-SMAC (IoT in social, mobile, analytics and cloud) (I-SMAC). IEEE, New York, pp 580–585
36. Fayaz SA, Zaman M, Butt MA (2021) To ameliorate classification accuracy using ensemble distributed decision tree (DDT) vote approach: an empirical discourse of geographical data mining. *Proc Comput Sci* 184:935–940
37. Ashraf M, Zaman M, Ahmed M (2018) Using ensemble Stacking method and base classifiers to ameliorate prediction accuracy of pedagogical data. *Proc Comput Sci* 132:1021–1040
38. Fayaz SA, Zaman M, Butt MA (2021) An application of logistic model tree (LMT) algorithm to ameliorate Prediction accuracy of meteorological data. *Int J Adv Technol Eng Explor* 8(84):1424–1440. <https://doi.org/10.19101/IJATEE.2021.874586>
39. Butt MA, Zaman M (2013) Assessment model based data warehouse: a qualitative approach. *Int J Comput Appl* 62(10). 16/j.scs.2020.102399
40. Fayaz SA, Zaman M, Butt MA (2022) Knowledge discovery in geographical sciences—a systematic survey of various machine learning algorithms for rainfall prediction. In: International conference on innovative computing and communications. Springer, Singapore, pp 593–608
41. Fayaz SA, Zaman M, Butt MA (2022) Performance evaluation of GINI index and information gain criteria on geographical data: an empirical study based on JAVA and python. In: International conference on innovative computing and communications. Springer, Singapore, pp 249–265
42. Fayaz SA, Zaman M, Butt MA (2022) Numerical and Experimental Investigation of Meteorological Data Using Adaptive Linear M5 Model Tree for the Prediction of Rainfall. *Rev Comput Eng Res* 9(1):1–12
43. Kaul S, Fayaz SA, Zaman M, Butt MA (2022) Is decision tree obsolete in its original form? A burning debate. *Artif Intell Rev* 36:105–113

Novel User Association Scheme Deployed for the Downlink NOMA Systems



Sunkaraboina Sreenu and Kalpana Naidu

Abstract Non-orthogonal multiple access (NOMA) is perceived as a crucial methodology used in 5G cellular networks on account of its enhanced spectrum efficiency and the ability to serve multi-users simultaneously. This paper proposes an unprecedented user pairing algorithm to improve the system's sum rate, by ensuring the significant channel gain difference even in between user pairs having low channel gains. To achieve this, the proposed algorithm first divides the users into two equal number of user groups according to their channel conditions. Then, these two groups are portioned into subgroups based on each user index (even and odd users). Finally, the pairing technique is performed between subgroups. Later, the 'fragmented power allotment method' distributes the power among users on each sub-channel with closed-form solutions. As a consequence of achieving these, simulation results validate that the proposed user pairing algorithm strengthens the data rate of the system when compared to orthogonal multiple access (OMA) and the conventional user pairing algorithms.

Keywords Sum rate · NOMA · User pair · Power allotment

1 Introduction

The rapid proliferation of Internet of Things (IoT) data transmissions and the widespread use of intelligent terminals have raised the demands for future cellular systems [1]. Thus, the fifth-generation (5G) networks endeavor to increase system capacity so as to satisfy the emerging modern market requirements [2, 3]. However, the traditional multiple access approaches struggle to meet the current demand in the presence of highly scarce spectrum resources. Consequently, in 5G, a new form of multiple access mode is introduced, i.e., non-orthogonal multiple access (NOMA)

S. Sreenu · K. Naidu (✉)
Department of ECE, National Institute of Technology, Warangal, India
e-mail: kalpana@nitw.ac.in

S. Sreenu
e-mail: sreenus@student.nitw.ac.in

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
H. Sharma et al. (eds.), *Communication and Intelligent Systems*, Lecture Notes
in Networks and Systems 461, https://doi.org/10.1007/978-981-19-2130-8_49

637

[4–7]. Additionally, NOMA tremendously proliferates system throughput and also supports many users simultaneously.

In conventional orthogonal multiple access (OMA) [8–19], only one user is assigned with single radio resource block (RB), with each RB accommodating time slots and/or frequency segments. This, in turn, may lead to the scarcity of resources (i.e., time slots and frequency bands) for some OMA users. To surmount this difficulty, NOMA offers a power domain that can assign a single RB to several users [20, 21]. Moreover, NOMA maintains user fairness by giving more powers to users with lower channel gains and also by providing smaller powers to users who retain higher channel gains [22].

Besides, in the downlink scenario, the transmitter superposes all user signals in a single waveform through superposition coding (SC). As a result, every user performs the successive interference cancellation (SIC) in order to interpret the respective user's signal correctly. Nevertheless, the superposition of all the users' information in the single RB may lead to error propagation and decoding latency [23]. Accordingly, to circumvent this limitation, it is required to limit the multiplexed users on the single RB. This, in turn, can be done by user pairing. Therefore, systematic user pairing and power allotment methods play a crucial role in reducing inter-user interference and improving system capacity.

In the recent literature, various user pairing techniques for NOMA systems have been presented. The first and straightforward user pairing approach is random pairing (RP) [24], in which BS selects users at random to form pairs. Despite its simplicity, this method does not perform well since it ignores the user's channel conditions. Analogously, the influence of user pairing is described in [25] on the performance of NOMA systems. In [25], the authors suggested two user pairing methods: conventional user pairing NOMA (C-NOMA) with invariant power distribution and cognitive radio-pertained NOMA. Additionally, [26, 27] deals with the uniform channel gain gap among paired users to overcome user pairing issues in the traditional near-far user pairing algorithm. Furthermore, while studying imperfect SIC, the proposed pairing approach in [26, 27] increases overall system capacity. Likewise, [28] provides the user grouping problem with a sub-optimal solution in a multi-user NOMA.

Analogously, [29] reduces the probability of outage. But, when the number of users increases in [29], SIC Performance will be reduced. However, this drawback is circumvented (or SIC performance is enhanced) by proposing a novel user pairing scheme in this article by considering the significant channel gain gap between user pairs.

Additionally, [30] presents the method of partitioning of user pairs, that have larger channel gain difference (or D-NLUPA of [30]), so as to ensure minimal sum capacity for each pair. Even though the division (of user pairs) is practiced in [30], user pairs having smaller channel gain gaps still do not profit to the greatest extent possible from being wholly shielded from the zero channel gain gap. Nevertheless, to get over the issues in the prevailing implementations, this article presents a novel user pairing approach to enhance the system sum rate further. Also, the intended method in this article augments the performance of user pairs, even though they have a smaller channel gain gap. Besides, the proposed method presumes that users are distributed

uniformly across the cell, thereby allowing equal preference to stronger channel gain users as well as weaker gain users. In particular, the objective of the proposed method is to pair all users by using the user’s index grouping, which helps to prevent the deterioration of the performance of low channel gain gap pairs. To accomplish this, the cell users are split into subgroups. Thereafter, user pairs are coupled, one from each of the subgroups (odd indexed users subgroup and even indexed users subgroup). It is done to preserve the channel gain difference between pairs of weak users as high as possible. Thus, compared to the aforementioned prior contributions, the uniqueness of the proposed user pairing is that all users (irrespective of channel gains of the users) benefit from the significant channel gain gap introduced by the proposed algorithm. Consequently, the SIC performance of all NOMA pairs will be enhanced. Finally, it is demonstrated that in comparison with traditional pairing and D-NLUPA schemes, the proposed user pairing technique is more efficient.

The remaining article is structured accordingly: The downlink NOMA system model is introduced in Sect. 2. Section 3 discusses the proposed user pairing and power allocation algorithms. Section 4 displays the results of the simulations. Ultimately, in Sect. 5, the article is concluded.

2 The System Model

Assume that the downlink NOMA system presented in Fig. 1. Here, the base station (BS) radiates signals to K users who are randomly deployed within the cell’s radius ($=R$) over S sub-channels. Further, the bandwidth available at BS is B , and each sub-channel bandwidth is $B_s = \frac{B}{S}$. Moreover, the overall power available at BS is P_T .

However, it is supposed that BS has complete channel state information (CSI) for all sub-channels (through feedback mechanism). Subsequently, the transmitted symbol on sub-channel ‘s’ is achieved by the sum of each user’s symbols. Consequently, the transmitted symbol is given as

$$x_s = \sum_{i=1}^{K_s} \sqrt{p_{i,s}} x_{i,s} \tag{1}$$

where $p_{i,s}$ denotes assigned power for the i th user on ‘s’ sub-channel. Additionally, $x_{i,s}$ represents the i th user’s information on sub-channel ‘s’. Besides, $x_{i,s}$ follows $E \left[|x_{i,s}|^2 \right] = 1$. Moreover, K_s is the number of multiplexing users on every sub-channel. Along with these, if the total power sent across all sub-channels is the same, then it follows

$$\sum_{i=1}^{K_s} p_{i,s} = P_s \tag{2}$$

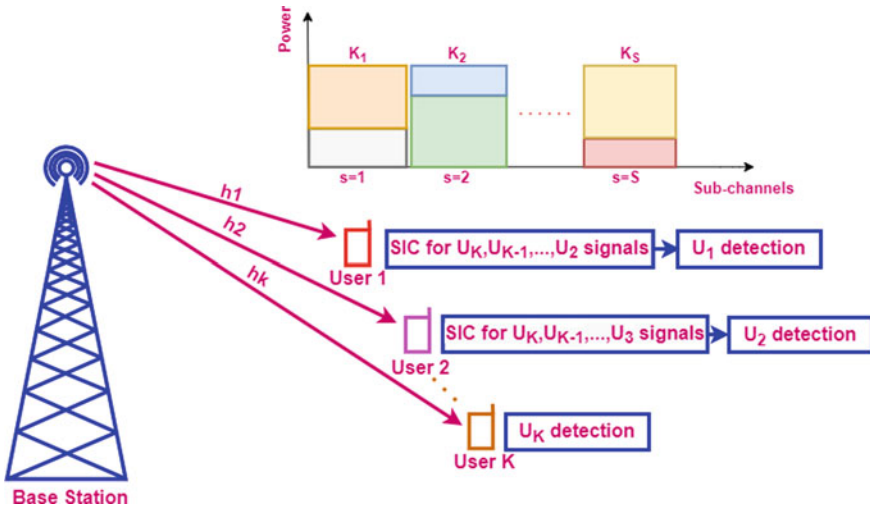


Fig. 1 Downlink NOMA system model

$$\text{and } \sum_{s=1}^S P_s = P_t \tag{3}$$

Anyway, zth user received signal is represented with

$$y_{z,s} = h_{z,s}x_s + n_{z,s} \tag{4}$$

$$\Rightarrow y_{z,s} = \underbrace{\sqrt{p_{z,s}}h_{z,s}x_z}_{\text{intended signal}} + \underbrace{\sum_{i=1, i \neq z}^{K_s} \sqrt{p_{i,s}}h_{i,s}x_i}_{\text{Inter-user interference}} + \underbrace{n_{z,s}}_{\text{noise}} \tag{5}$$

In (5), $h_{z,s}$ is channel gain from BS to zth user on sub-channel ‘s’. In addition, $n_{z,s}$ is the Gaussian noise along with interference from other cells. Also, the noise power spectral density of $n_{z,s}$ is represented with N_o , which is assumed to be constant over all sub-channels. Further, without loss of generality, presume that users allotted with the same sub-channel are sorted according to channel gains as $h_{1,s} \geq h_{2,s} \geq h_{3,s} \geq \dots \geq h_{K_s,s}$.

Meanwhile, the basic principle of NOMA is assigning greater power to the lowest channel gain user (weak user) and lower power to the highest channel gain user (strong user) [31]. Thus, each user’s power allocation is purely dependent on CSI. That is, considering two users operating on the ‘s’ sub-channel and $h_{1,s} \geq h_{2,s}$; then, BS will allot the transmission powers as $p_{1,s} \leq p_{2,s}$. Because many users have been assigned to the same sub-channel, the SIC mechanism can isolate the multiplexed signal at each user as follows: a) The weak user decodes its signal from the received

signal by considering the robust user signal as noise. b) Additionally, the vigorous user conducts SIC, in which weak user signal is decoded first and then deducts it from the entire received signal to get its specific signal [32].

As a result of using SIC, the achievable throughput of k th user on the sub-channel 's' is obtained as

$$R_{k,s} = B_s \log_2 \left(1 + \frac{p_{k,s} |h_{k,s}|^2}{\sum_{i=1}^{k-1} p_{i,s} |h_{k,s}|^2 + N_o B_s} \right) \quad (6)$$

Consequently, the NOMA system's sum rate is achieved by summing the respective user rates, and it is given by

$$R_{\text{sum}} = \sum_{s=1}^S \sum_{k=1}^{K_s} R_{k,s} \quad (7)$$

On the other hand, the OMA user rate is determined by

$$R_k^{\text{OMA}} = \frac{1}{K} B \log_2 \left(1 + \frac{P_t |h_k|^2}{N_o B} \right) \quad (8)$$

where the factor $\frac{1}{K}$ is because of the multiplexing degradation in the conventional OMA. Equivalently, from (8), the data rate of the OMA system is ascertained as

$$R_{\text{sum}}^{\text{OMA}} = \sum_{k=1}^K R_k^{\text{OMA}} \quad (9)$$

3 Proposed Methods for User Pairing and Power Allotment

The intended method follows two strategic processes: (a) Initially, user pairing is accomplished such that even the user having the lower channel gain gets its required capacity. (b) After that, powers are apportioned to the resolved user pairs essentially to enhance the total rate of the system specified in (7). These two techniques are detailed hereinafter.

3.1 Proposed User Pairing Algorithm

In the conventional user pairing technique, user pairs are formed by coupling weak users and strong users to maintain the highest channel gain disparity between paired users. But, cell mid-user pairs have significantly less channel gain difference because of this pairing, leading to severe interference at the receiver. Conversely, to ensure a minimum channel gain difference for low channel gain user pairs, the dividing step is introduced in D-NLUPA. However, in a massive users scenario (where channel gains are slightly similar), low channel gain users do not benefit fully even though they employed the division step before pairing. So as to increase the performance of low channel gain gap user pairs and the overall sum throughput of the system, users are formed into subgroups based on each users' index (even and odd) in the proposed algorithm. The basic concept of the proposed user pairing procedure is as follows:

In the first step, all users are indexed based on the decreasing order of their channel conditions. Then next, indexed users are divided into two equal number of user groups: strong (high channel gain) users group ($U_1; U_2; \dots; U_{\frac{K}{2}}$) and weak (low channel gain) users group ($U_{\frac{K}{2}+1}; U_{\frac{K}{2}+2}; \dots; U_K$). This division is performed to preserve the particular gap among the user's channel gains in a pair. After that, each group in the above step is partitioned into two subgroups: odd and even indexed strong user subgroups and similarly odd and even indexed weak user subgroups. Next, two distinct sets are developed. Out of these two sets, one set consists of a combination of odd indexed strong users and even indexed weak users subgroups. Similarly, another set contains remaining even and odd indexed user subgroups. Finally, the pairing method is performed on each set whereby the first best channel gain user of every subgroup paired with the foremost best channel gain user of the other subgroup, then the next-best channel user of every subgroup paired with the next-best channel user of another subgroup, and this pairing is repeated till the end.

To make the understanding easier, Fig. 2 displays the process of our proposed user pairing approach with 16 users randomly located with uniform distribution in the cell. The user organization in each group is not regular, as shown in Fig. 2. As previously stated, the 16 sorted users are divided into two groups first, and then, these two groups are separated into four subgroups depending on the user's index. Each subgroup is made up of four users. Then, the odd indexed strong users subgroup establishes a set with even indexed weak users subgroup and even indexed strong users subgroup forms a set with odd indexed weak users subgroup. Finally, the pairing is implemented within each set. Algorithm 1 describes the proposed user pairing algorithm in further depth.

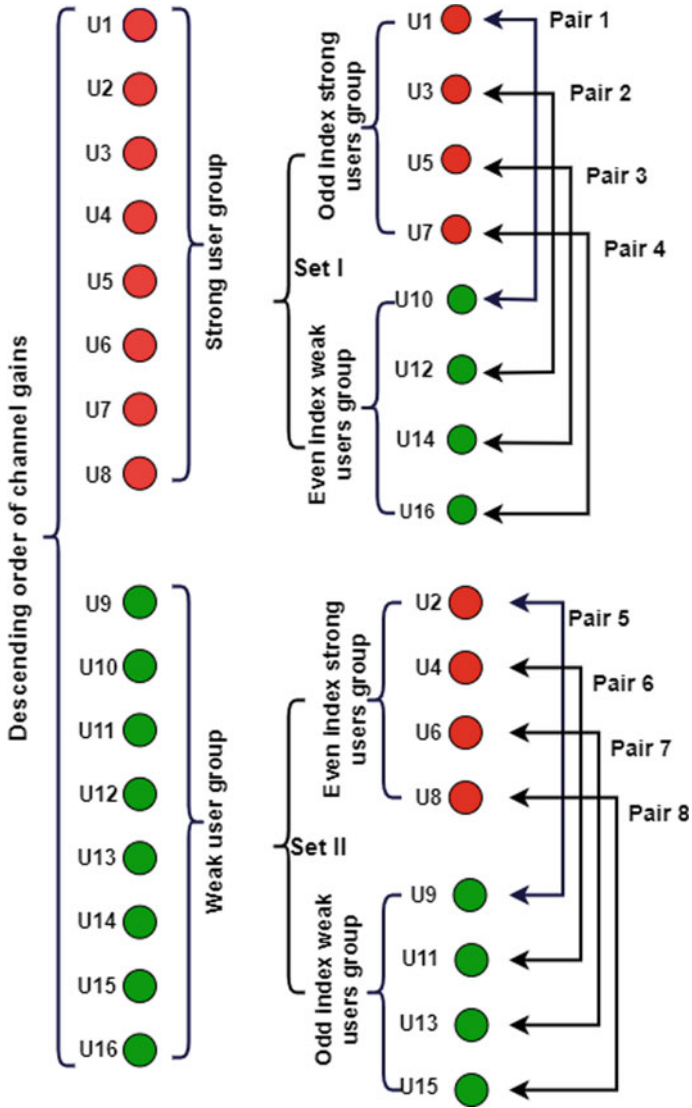


Fig. 2 Proposed user pairing scheme

Algorithm 1: Proposed user pairing scheme

Input: Users' distribution in the cell

Output: Collection of user pairs generated by odd and even indexed users subgroups

1 : Initialization.

2 : Sort all K users according to their channel gains in descending order.

3 : K users division into two groups.

4 : Each group forms the odd index users and even index users subgroups.

5 : **For users of odd index strong users group and even index weak users group**

6 : Pair 1 : $\left(U_1; U_{\frac{K}{2}+2} \right)$

7 : Pair 2 : $\left(U_3; U_{\frac{K}{2}+4} \right)$

8 : \vdots

9 : \vdots

10 : Pair $\frac{K}{4}$: $\left(U_{\frac{K}{2}-1}; U_{\frac{K}{2}+(\frac{K}{2}-1)+1} \right)$

11 : **For users of even index strong users group and odd index weak users group**

12 : Pair $\frac{K}{4} + 1$: $\left(U_2; U_{\frac{K}{2}+1} \right)$

13 : Pair $\frac{K}{4} + 2$: $\left(U_4; U_{\frac{K}{2}+3} \right)$

14 : \vdots

15 : \vdots

16 : Pair $\frac{K}{2}$: $\left(U_{\frac{K}{2}}; U_{\frac{K}{2}+(\frac{K}{2}-1)} \right)$

3.2 Power Allotment Method

After carrying out the user pairing, the fragmented power allotment (FPA) method proposed in [33] is exercised to assign powers for paired users on every sub-channel. FPA is used here because of the following reasons :

- (i) FPA is a widely used low-complex method to enhance the sum throughput of the system.
- (ii) Also, the FPA method is identical to long-term evolution (LTE) uplink power control [34, 35].

Thus, in this scheme, the power per sub-channel is allocated to its paired users based on users' channel conditions, and each paired user power on the sub-channel 's' can be represented by [33],

$$p_{s,i} = P_s \times \frac{\left(|h_{s,i}|^2 \right)^{-\xi}}{\left(|h_{s,i}|^2 \right)^{-\xi} + \left(|h_{s,j}|^2 \right)^{-\xi}} \quad (10)$$

P_s is 's' sub-channel assigned power, and ξ is the power distribution decay factor, which lies in between 0 and 1. If $\xi = 0$, each user gets identical power in the user pair. Moreover, by increasing ξ , the weak users obtain more power, and ξ remains constant throughout all sub-channels and is assessed using computer simulations. Besides, the value of ξ is set to 0.4 during the simulation to attain the maximum sum rate and maintain weak user fairness.

4 Simulation Results

Simulations are conducted in MATLAB. Besides, the effectiveness of the proposed user pairing approach is studied in this section. A single-cell downlink NOMA scheme is taken in the simulations, with K users distributed at random. Further, it is assumed that the BS is situated at the middle of the cell, which has perfect CSI information. Moreover, Each sub-channel is multiplexed with only two users, so as to minimize the receiver complexity. Although the proposed user pairing algorithm works for all scenarios, one specific example is elucidated below to make the functioning of the intended user pairing technique apparent.

Here, in the considered scenario, 12 users are deployed in the cell. For simplicity, presume that channel gains of users are $H_i = [4, 49, 93, 53, 75, 3, 1, 69, 59, 7, 89, 24]$ dB, here $H_i = \frac{|h_i|^2}{N_o}$, $\forall i \in K$ and transmit power is 30 dBm. According to the Algorithm 1, users are indexed based on descending order of their channel conditions i.e., $[93, 89, 75, 69, 59, 53, 49, 24, 7, 4, 3, 1]$ dB. Then, indexed users are separated into two groups: strong user group ($U_1; U_2; U_3; U_4; U_5; U_6$) and weak user group ($U_7; U_8; U_9; U_{10}; U_{11}; U_{12}$). After that, two subgroups are formed from each group in the above step based on user index, which are $g1 = (U_1; U_3; U_5)$, $g2 = (U_2; U_4; U_6)$, $g3 = (U_7; U_9; U_{11})$ and $g4 = (U_8; U_{10}; U_{12})$. Next, user coupling is performed between subgroups $g1$ and $g4$, $g2$ and $g3$ to form user pairs; as a result, user pairs are $(U_1; U_8)$, $(U_3; U_{10})$, $(U_5; U_{12})$, $(U_2; U_7)$, $(U_4; U_9)$, and $(U_6; U_{11})$. Finally, the total system rate is obtained by using equation (7). In this case, the sum rates of the proposed method, existing D-NLUPA, prevalent C-NOMA, and prevailing OMA schemes respectively are 5.5751, 5.5618, 5.5178, and 4.6331 bps. Hence, the proposed system sum rate is better than the existing user pairing schemes and the prevalent OMA. In addition, the performance of each NOMA pair with a small channel gain difference has increased.

For convenience, the BS evenly allocates the total transmission power among sub-channels. Then, a sub-channel power is distributed between their paired users using the FPA algorithm. We carried out 10000-channel accomplishments in our evaluations. The necessary simulation specifications are presented in Table 1.

Figure 3 shows the comparison between the sum rate and total power available at the BS for $K = 20$ users. Accordingly, it is comprehensible from Fig. 3 that the sum rate rises as the transmission power varies. The influence of the different number of users on the sum rate of the system is studied in Fig. 4. Besides, the transmission

Table 1 Parameters chosen for simulations

Parameter name	Value
No. of users (K)	20
No. of sub-channels (S)	10
Cell radius (R)	500 m
Transmission power (P_t)	30 dBm
System bandwidth (B)	5 MHz
Channel model	Rayleigh fading channel
Noise power density (N_o)	-174 dBm/Hz
Path loss exponent (ν)	4
FPA decay factor (ξ)	0.4

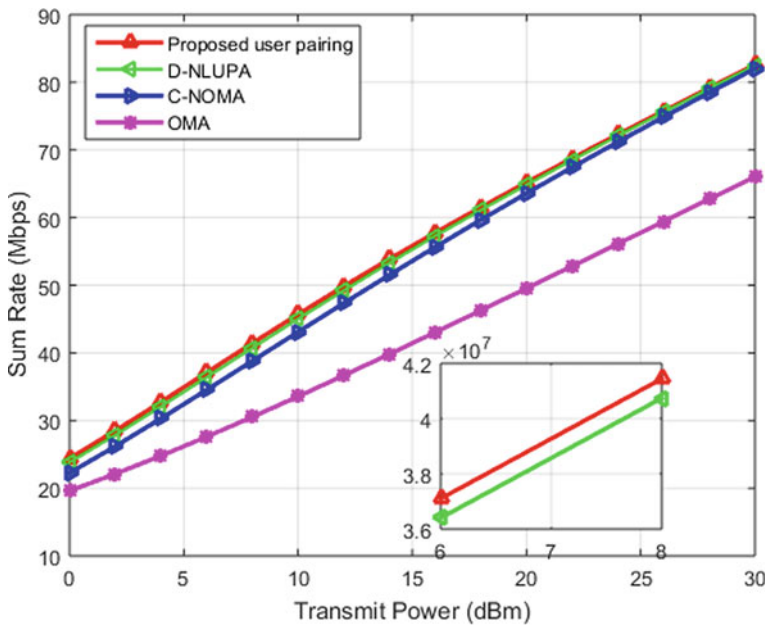


Fig. 3 Comparison of system’s total rate with BS transmission power

power is fixed at 30 dBm in Fig. 4. Further, it is shown that the sum rate of the system improves with the increase in the number of users. Therefore, Figs. 3 and 4 exhibit that the performance of the presented method is superior to the existing algorithms C-NOMA, D-NLUPA, and conventional OMA.

It is substantiated here that the proposed user pairing NOMA system performs better than the existing pairing techniques. Hence, Table 2 reveals the sum rate improvement of the propounded approach over the other algorithms and OMA systems for various number of users and $P_t = 20$ dBm.

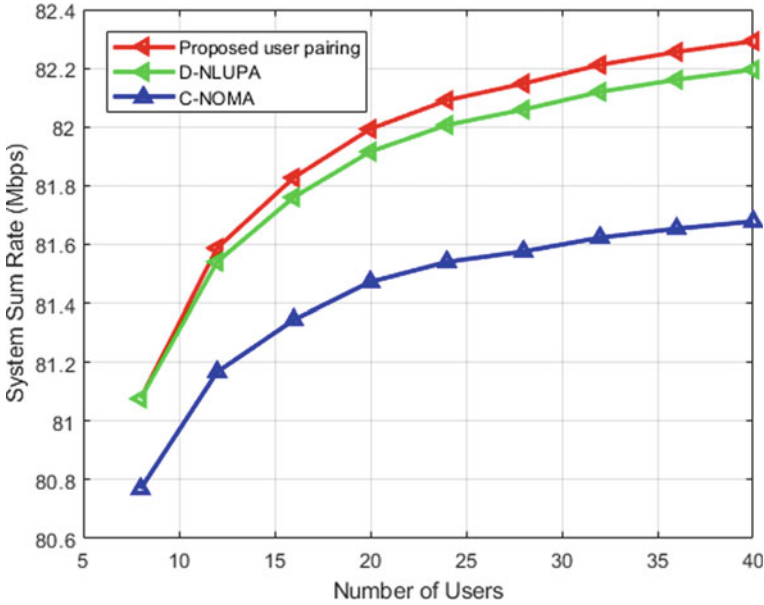


Fig. 4 Comparison of system’s total rate with different number of users per cell at $P_t = 30$ dBm

Table 2 Comparison of the proposed algorithm with existing methods in terms of sum rate for various number of users and $P_t = 20$ dBm

Algorithm	Overall system sum rate (Mbps)		
	At $K = 20$ users	At $K = 40$ users	At $K = 60$ users
OMA	54.15	54.18	54.21
C-NOMA	69.36	69.51	69.58
D-NLUPA	71.23	71.60	71.74
Proposed method	71.69	72.18	72.35

Figure 5 depicts the sum rate per user pair versus the index of the user pairing. Here, in Fig. 5, sum data rates are ordered in ascending order. It is noticed here that 50% of the user pairs gain from greater data rates in comparison with the C-NOMA method, while others are benefited from the minimum required data rates. The D-NLUPA system also yields the same outcomes. However, as compared to the D-NLUPA, 25% of the proposed scheme pairs in each set (of smaller channel gain gap pairs) benefit from the higher data rates. As a result, near-zero performance was avoided for smaller gain difference users in the proposed method, and also sum rate of the system is improved in the proposed method.

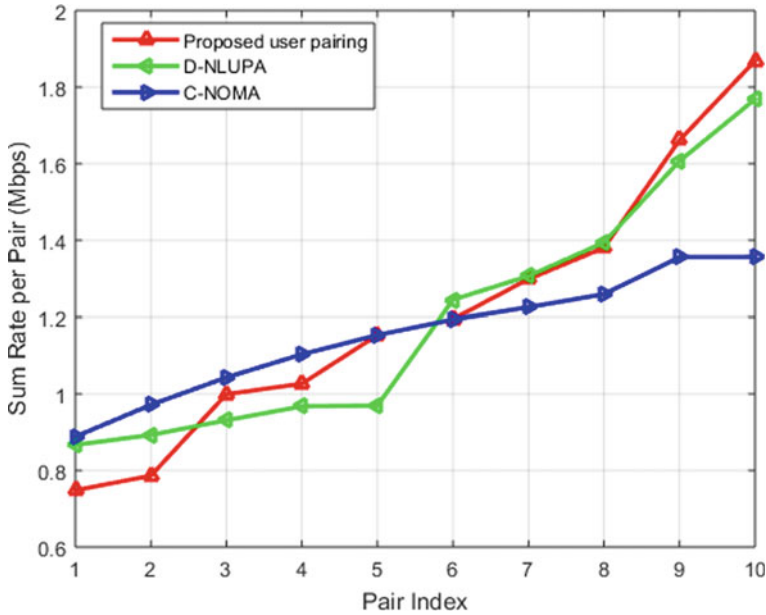


Fig. 5 Comparison of system's sum rate per pair with index of the pair

5 Conclusion

In this article, user pairing for the downlink NOMA is essentially inferred from the channel gain variations of the user pairs. Correspondingly, a novel user pairing approach is proposed according to the user's index grouping, intending to increase the system's sum rate. Later on, the FPA technique is used to assign the powers for paired users on each sub-channel. As a result, the proposed technique guarantees significant data rates (or sum rates) even for user pairs having small channel gain gap. Therefore, this method consistently avoids the deterioration in system capacities. Moreover, the simulation results revealed that the user pairing system presented is more efficient with regard to the system's sum rate than D-NLUPA, C-NOMA, and OMA methods.

References

1. Liu Li Y et al (2019) Energy-efficient power allocation for millimeter wave beamspace MIMO-NOMA systems. *IEEE Access* 7:114582–114592
2. Dai L, Wang B, Ding Z, Wang Z, Chen S, Hanzo L (2018) A survey of non-orthogonal multiple access for 5G. *IEEE Commun Surveys Tutorials* 20(3):2294–2323

3. Baghani M, Parsaeefard S, Derakhshani M, Saad W (2019) Dynamic non-orthogonal multiple access and orthogonal multiple access in 5G wireless networks. *IEEE Trans Commun* 67(9):6360–6373
4. Liu Y, Qin Z, Elkashlan M, Ding Z, Nallanathan A, Hanzo L (2017) Non-orthogonal multiple access for 5G and beyond. In: *Proceedings of the IEEE*, vol 105(12), pp 2347–2381
5. Haci H, Zhu H, Wang J (2017) Performance of non-orthogonal multiple access with a novel asynchronous interference cancellation technique. *IEEE Trans Commun* 65(3):1319–1335
6. Zhang X, Wang F (2018) Resource allocation for wireless power transmission over full-duplex OFDMA/NOMA mobile wireless networks. *IEEE J Sel Areas Commun* 37(2):327–344
7. Saraereh OA, Alsaraira A, Khan I, Uthansakul P (2019) An efficient resource allocation algorithm for OFDM-based NOMA in 5G systems. *Electronics* 8(12):1399
8. Naidu K, Khan ZA (2015) Fast computation of generalized water filling problems. *IEEE Sign Process Lett* 22(11):1884–1887
9. Naidu K, Zafar AK, Hanzo L (2016) An efficient direct solution of cave-filling problems. *IEEE Trans Commun* 64(7):3064–3077
10. Naidu K et al (2018) Quick resource allocation in heterogeneous networks. *Wireless Networks* 24(8):3171–3188
11. Naidu K et al (2018) Swift resource allocation in wireless networks. *IEEE Trans Veh Technol* 67(7):5965–5979
12. Naidu K et al (2018) Quicker solution for interference reduction in wireless networks. *IET Commun* 12(14):1661–1670
13. Naidu K et al (2021) Efficient allotment of resources in heterogeneous communication. *Wireless Networks*, pp 1–23
14. Naidu K, Kumar AR, Vikas V (2017, January) The fastest possible solution to the weighted water-filling problems. In: *IEEE 7th International advance computing conference (IACC)*. IEEE, New York, pp 614–618
15. Naidu K, Khan MZA (2016, September) Fast algorithm for solving cave-filling problems. In: *IEEE 84th vehicular technology conference (VTC-Fall)*. IEEE, New York, pp 1–5
16. Naidu K, Khan MZA (2015, May) Weighted water-filling algorithm with reduced computational complexity. In: *Proceedings of ICCIT Conference*
17. Naidu K, Khan MZA, Desai UB (2011, December) Optimal power allocation for secondary users in CR networks. In: *Fifth IEEE International conference on advanced telecommunication systems and networks (ANTS)*. IEEE, New York, pp 1–6
18. Naidu K et al (2019) Optimal resource allocation based on particle swarm optimization. In: *Advances in communications, signal processing, and VLSI: select proceedings of IC2SV*, 199
19. Naidu K (2019, May) Simple solution to reduce interference in cognitive radio networks. In: *TEQIP III sponsored international conference on microwave integrated circuits, photonics and wireless networks (IMICPW)*. IEEE, New York, pp 199–203
20. Benjebbour A, Saito Y, Kishiyama Y, Li A, Harada A, Nakamura T (2013, November) Concept and practical considerations of non-orthogonal multiple access (NOMA) for future radio access. In: *International symposium on intelligent signal processing and communication systems*. IEEE, New York, pp 770–774
21. Mounchili S, Hamouda S (2020, June) New user grouping scheme for better user pairing in NOMA systems. In: *International wireless communications and mobile computing (IWCMC)*. IEEE, New York, pp 820–825
22. Al-Abbasi ZQ, So DKC (2017) Resource allocation in non-orthogonal and hybrid multiple access system with proportional rate constraint. *IEEE Trans Wireless Commun* 16(10):6309–6320
23. Agiwal M, Roy A, Saxena N (2016) Next generation 5G wireless networks: a comprehensive survey. *IEEE Commun Surveys Tutorials* 18(3):1617–1655
24. Zhang H, Zhang D, Meng W, Li C (2016, May) User pairing algorithm with SIC in non-orthogonal multiple access system. In: *IEEE international conference on communications (ICC)*. IEEE, New York, pp 1–6

25. Ding Z, Fan P, Poor HV (2015) Impact of user pairing on 5G nonorthogonal multiple-access downlink transmissions. *IEEE Trans Veh Technol* 65(8):6010–6023
26. Shahab MB, Irfan M, Kader MF, Young Shin S (2016) User pairing schemes for capacity maximization in non-orthogonal multiple access systems. *Wireless Commun Mob Comput* 16(17):2884–2894
27. Ahmad H, Ali DM, Muhamad WNW (2020, August) The effect of user pairing strategies on the downlink throughput of NOMA. In: 11th IEEE control and system graduate research colloquium (ICSGRC). IEEE, New York, pp 43–48
28. Ali MS, Tabassum H, Hossain E (2016) Dynamic user clustering and power allocation for uplink and downlink non-orthogonal multiple access (NOMA) systems. *IEEE Access* 4:6325–6343
29. Xu P, Yuan Y, Ding Z, Dai X, Schober R (2016) On the outage performance of non-orthogonal multiple access with 1-bit feedback. *IEEE Trans Wireless Commun* 15(10):6716–6730
30. Islam SMR, Zeng M, Dobre OA, Kwak K (2018) Resource allocation for downlink NOMA systems: key techniques and open issues. *IEEE Wireless Commun* 25(2):40–47
31. Akbar A, Jangsher S, Bhatti FA (2021) NOMA and 5G emerging technologies: a survey on issues and solution techniques. *Comput Networks* 190:107950
32. Lan Y, Benjebbour A, Li A, Harada A (2014, May) Efficient and dynamic fractional frequency reuse for downlink non-orthogonal multiple access. In: IEEE 79th vehicular technology conference (VTC Spring). IEEE, New York, pp 1–5
33. Benjebbovu A, Li A, Saito Y, Kishiyama Y, Harada A, Nakamura T (2013, December) System-level performance of downlink NOMA for future LTE enhancements. In: IEEE Globecom workshops (GC Wkshps). IEEE, New York, pp 66–70
34. Endo Y, Kishiyama Y, Higuchi K (2012, August) Uplink non-orthogonal access with MMSE-SIC in the presence of inter-cell interference. In: International symposium on wireless communication systems (ISWCS). IEEE, pp 261–265
35. Sharma P, Kumar A, Bansal M (2021) Performance analysis for user selection-based downlink non-orthogonal multiple access system over generalized fading channels. *Trans Emerg Telecommun Technol*

Analyzing the Performance of a Digital Shadow for a Mixed-Model Stochastic System



Philane Tshabalala  and Rangith B. Kuriakose 

Abstract The industry 4.0 era has brought about many trend breaking strategies in the manufacturing structure. These new approaches have resulted in ensuring the production process being more autonomous and efficient. Digital transformation is one such strategy introduced to improve product efficiency in the manufacturing industry. Digital twins and digital shadows are the two popular tools used for implementing digital transformation. This article discusses the design of a Digital Shadow created using MATLAB/Simulink for analyzing the performance and operation of a Mixed-model stochastic system. The research uses the case study of a water bottling plant capable of producing multiple variants of water bottles. The article describes how the digital shadow of the plant is able to simulate a specific input to determine possible bottlenecks, use of raw materials and the production efficiency.

Keywords Industry 4.0 · Assembly line balancing · Digital shadow · Mixed-model stochastic assembly lines

1 Introduction

The Fourth Industrial Revolution (also known as Industry 4.0) has brought about major changes in the manufacturing industry [1]. These changes are made to improve productivity and product efficiency [2]. One noticeable change is the shift from traditional mass production to customized production [3], as the current market demands faster deliveries and personalized products [4]. The manufacturing firms that are capable of responding to market fluctuations and customer demand in a short period of time have the advantage of staying competitive in the global market [5].

The two universal operational modes used in the manufacturing industry to satisfy the current market are Make-to-Stock (MTS) and Make-to-Order (MTO) [5]. The MTS systems are used for large, stable and continuous orders [6], as they produce products in volume and stock them as a supply of finished goods [7]. However,

P. Tshabalala (✉) · R. B. Kuriakose
Central University of Technology, Bloemfontein, Free State, South Africa
e-mail: philanetshabalala@cut.ac.za

customization is limited in the MTS systems, as their focus is on future customer demands. Unlike MTS, in the MTO systems the production orders are not predictable and production only starts after the customer's order is placed and received [8].

The MTO systems are stochastic in nature [3], as they need to provide product variety. Product variety is achieved through the introduction of Multi/Mixed Model Assembly Lines (MMALs), used with stochastic task times [9], hence the focus of this research is on Mixed-Model Stochastic (MMS) assembly lines. Line balancing is one of the main problems in these type of assembly lines [10] and the reduction of bottlenecks in production is one of the main objectives of assembly line balancing [3].

Digital shadows are a great tool for predicting possible bottlenecks and reduction of cycle-time, as their ability to link the physical and virtual product [11] enables them to run tests and simulations in order to improve product efficiency, flexibility, reduce maintenance and reduce time to market [12]. Cycle-time is the amount of time it takes a single workstation to complete all the tasks assigned to it before the product can move on to the next workstation [13].

This article analyzes the operation and performance of a digital shadow in a MTO water bottling plant and shows how a digital shadow is a possible solution in reduction of cycle-time in the manufacturing setup. The article is divided into four sections. Section 2 discusses assembly lines, line balancing, digital shadow definitions and the water bottling plant case study. Section 3 details the methodology for designing a digital shadow. Section 4 discusses the results and analysis of the digital shadow created.

2 Background

2.1 *Assembly Lines*

An assembly line is one of the most valuable components in the manufacturing setup [10]. An assembly line can be defined as a production line with multiple work stations where products flow from the first station to the last station and a number of tasks are assigned to each of these stations [14]. The purpose of assembly lines is to assign these tasks to the operators in each station in order to enhance production [15].

Assembly lines are broadly divided into three categories [14], Single-model assembly lines, Multi/Mixed-model assembly lines [16]. Single model assembly lines are those assembly lines whereby only a single product is produced at a time [17]. Multi-model assembly lines are designed to produce multiple products in single variants [18]. Mixed-model assembly lines produce a variety of products at the same time [3]. The competitiveness of the current market makes it a necessity for manufacturing firms to switch to mixed-model assembly lines in order to meet the customer demand [19].

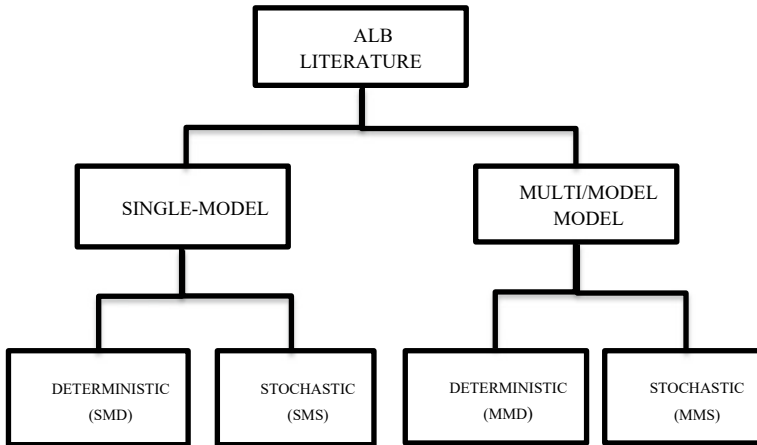


Fig. 1 Classification of assembly line balancing literature [21]

2.2 Assembly Line Balancing

Line balancing is defined as the leveling of the tasks in each workstation, assigning these tasks to the operators in such a way that they all have the same number of tasks [17]. The decision making of assigning these tasks is referred to as Assembly Line Balancing (ALB) [20]. The classification of ALB is shown in Fig. 1 [21]. The single model assembly lines and multi/mixed model are divided into deterministic and stochastic processes [21].

The time taken by each workstation to complete the task assigned to it is referred to as the Task/Processing time [3]. Task times can either be deterministic or stochastic depending on the goal of an assembly line. Deterministic task times refers to assembly lines whereby the inputs are predetermined and stochastic task times are assembly lines whereby the inputs are not predictable [3].

This research and the results focus on a Mixed Model Assembly Line with stochastic task times. This is affected by using the case study of a water bottling plant. Clients place customized orders, pertaining to their requirement through a cloud server. The orders get processed by an optimization model according to the customer requirements and time of delivery. A 3-dimensional model of the plant is shown in Fig. 2.

As depicted in Fig. 2, the plant is subdivided into three sections. Section A depicts the water filling unit, Section B, the capping unit and Section C is the packaging unit. Three Smart Manufacturing Units (SMUs) driven by PLCs are used to complete this process in the physical plant. The water bottling plant can bottle, cap and store two variants (the 300 ml and the 500 ml bottles) of bottled water. The function and execution of each SMU is detailed in Table 1.

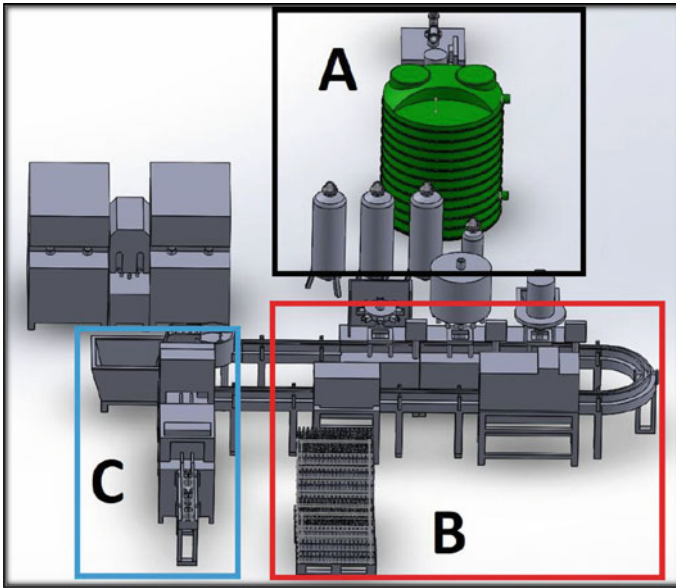


Fig. 2 3D model of the water bottling plant

Table 1 SMU function and execution in water bottling plant

SMU	Function	Execution
SMU 1	Filling water	Siemens S7 PLC
SMU 2	Capping	Kuka Robot
SMU 3	Packaging	Festo Gripper

3 Aim and Methodology

The aim of this article is to design a Digital Shadow (DS) to analyze the performance and the operation of the water bottling plant. A digital shadow is a model in which the flow of data is one way, from the physical product to the virtual product and any change in the state of a physical product will automatically result to a change in the state of a virtual product but not vice versa [22].

The direction of data flow (between the physical product and the virtual product) is what distinguishes a digital shadow from a digital twin [23]. The advantage of a digital shadow is that they are simulation based models [24]. This allows the plant manager to simulate all possible outcomes of a specific operation and predetermine completion time and possible bottlenecks. However, as the digital shadows are not physically connected to the plant, they are not able to receive or relate with real-time data.

For the purposes of this article, a digital shadow of the water bottling plant comprising of the three Smart Manufacturing Units was developed using MATLAB

/Simulink. The input to the plant was the actual customer inputs. The simulation outputs were designed to monitor the water level in the tank, the time it takes to fill the bottles, the time it takes to cap the bottles and the overall production rate. The Simulink model used for the design is shown in Fig. 3.

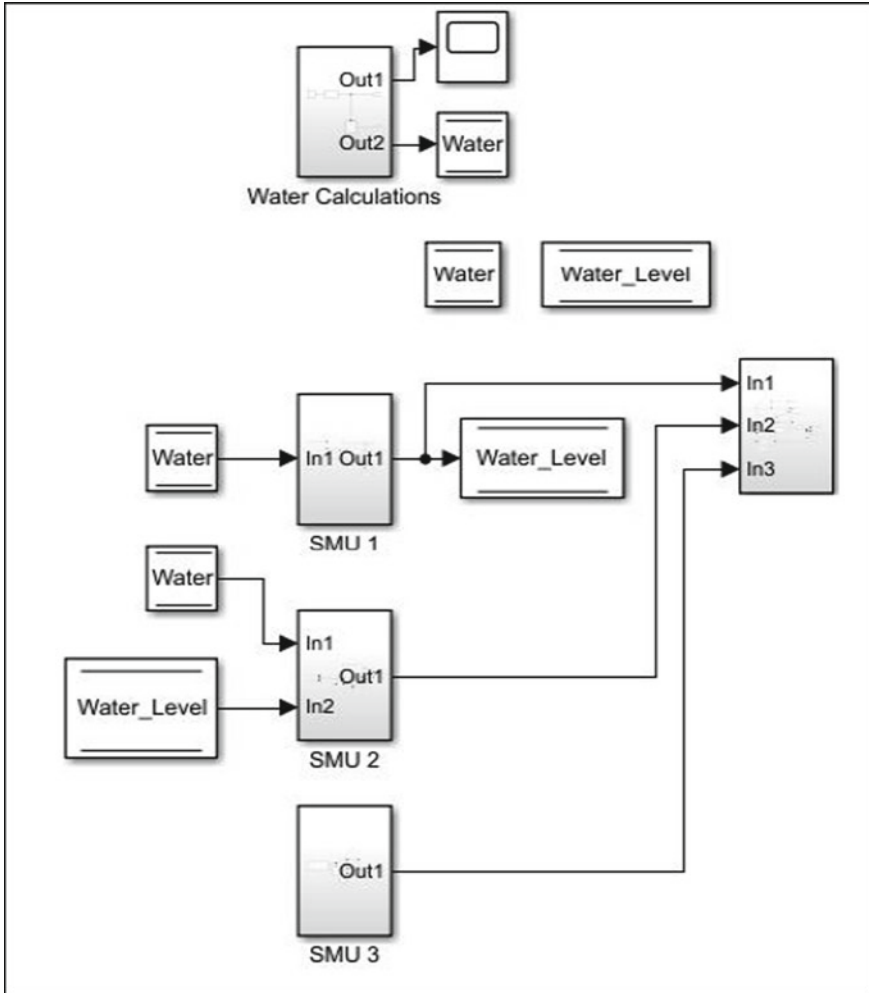


Fig. 3 Simulink model for digital shadow of water bottling plant

4 Results and Analysis

As explained in the previous section, the aspects monitored in this section are water level in the tank, the time it takes to fill and cap the bottles and the overall production rate. For analysis purposes, the tests are done on an input for 500 ml bottles only. The first aspect to be tested was the level of water in the tank as the process starts. This is depicted in Fig. 4.

As it can be seen in Fig. 4, water level, which is a factor governed by SMU 1, is constantly decreasing with the gradual increase in time. As this is the start of the production process, the tank is full and poses no threats in the form of bottlenecks. However, with the output from the digital shadow, a prediction can be made as to when the tank needs to be refilled, before it affects production.

Water level in the tank is inversely proportional to the number of bottles that get filled. The bottles getting filled is an aspect that is controlled by SMU 2. The results of the water bottles getting filled is shown in Fig. 5. As seen from Fig. 5, there is a steady rise in the number of 500 ml bottles that get filled.

The next aspect that needs to be analyzed is the rate of capping bottles that are filled with water. This analysis is particularly significant as bottle capping is comparatively slower than water filling, hence it is an area that can possibly give rise to bottlenecks in the production. The bottle capping is shown in Fig. 6.

As expected, the bottle capping process is slower than the filling process, hence this will have an impact on the production efficiency, as seen in Fig. 7. From the analysis of Fig. 7, it can be seen that the production efficiency rises to 72% during the initial stages of the production.

However, as the number of bottles increases, the production efficiency decreases below 70%. This is due to the bottlenecks that are caused by the bottle capping process. It can be seen that once the bottleneck is relieved, the production efficiency picks up and settles into a steady state value of about 70%.

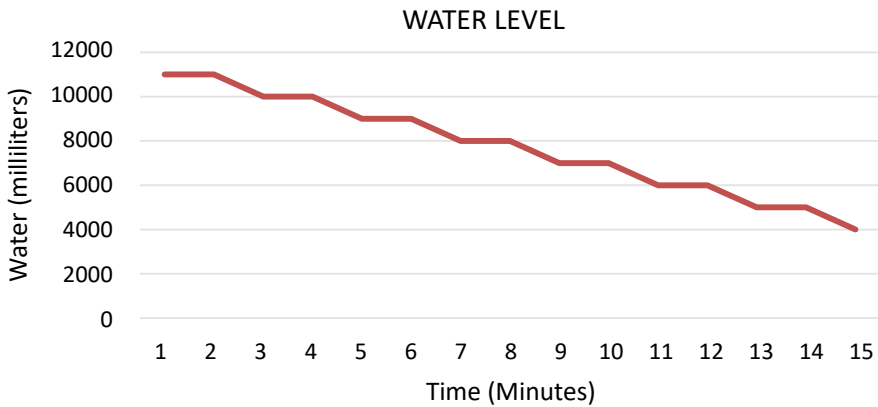


Fig. 4 Results of water level monitoring in the digital shadow

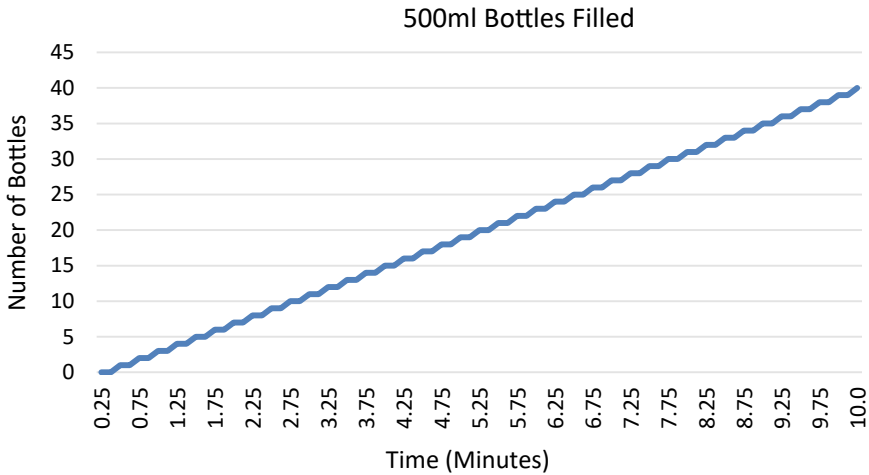


Fig. 5 Results of bottle filling in the digital shadow

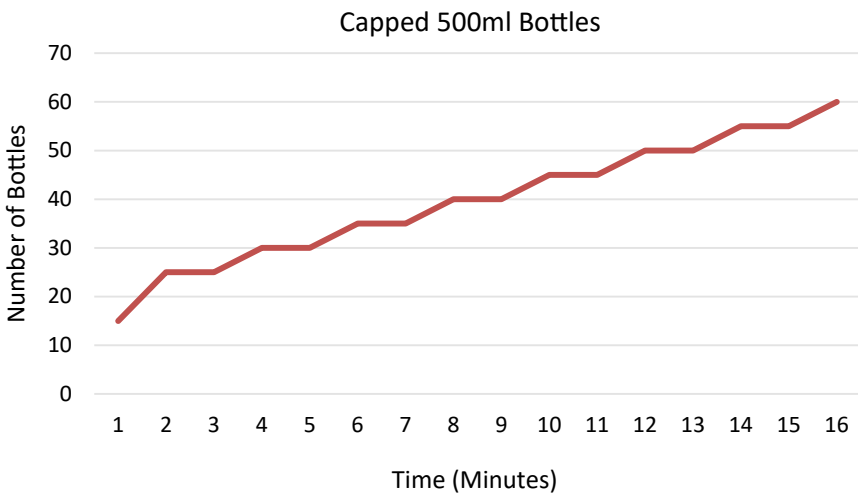


Fig. 6 Results of bottle capping in the digital shadow

Analyzing the results of this study with other similar studies highlight the following. McMullen and Tarasewich proposed a solution for reducing the number of workstations, using Ant Colony techniques [25]. McMullen and Frazier also proposed a workstation reduction solution and focused on line balancing in mixed-model stochastic assembly lines, using simulated annealing [3]. Xu and Xiao proposed a solution for the reduction of cycle time, using the fuzzy algorithm [26]. Matanachai and Yano researched on positive drift but on mixed-model assembly lines with various production systems [27].

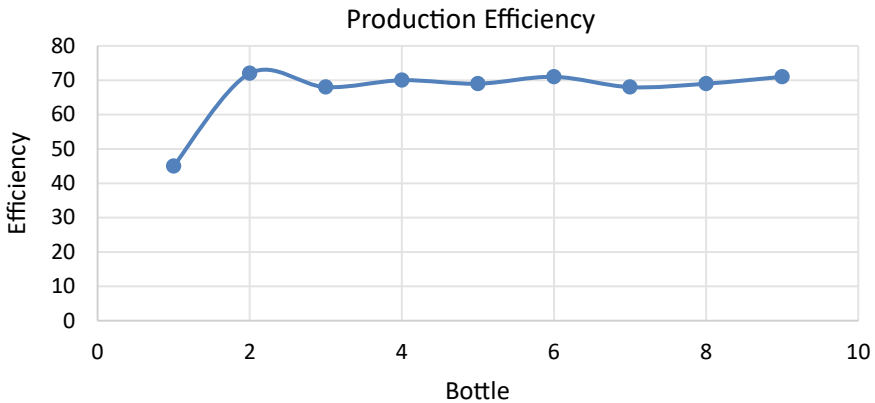


Fig. 7 Production efficiency of water bottling plant as determined by the digital shadow

All these studies focused on problems other than cycle-time except for Xu and Xiao. However, the solution proposed also use optimal solutions. The limitation of optimal solutions is that they have a limited number of variables and constraints that can be used in the software that solve models, therefore cannot solve large-scale problems being posed by mixed-model assembly lines.

5 Conclusion

This article showcased the impact of having a digital shadow in a water bottling plant. The water bottling plant was chosen as it represents a Mixed-Model Stochastic system. With the aid of the digital shadow, various factors such as the flow rate and rate of water filling can be varied to see the impact that it has on the production efficiency of the plant for a specific input. This will assist the plant manager in determining the performance and efficiency for various “what-if” scenarios that might occur during the production.

However, it should be noted that that the digital shadow is not physically connected to the water bottling plant and therefore not able to get real-time data from the plant during the actual operation. As part of the recommendation from this project will aim to construct a digital twin for the water bottling plant with inputs from the different SMU’s sourced using various sensors. It is hypothesized that the real-time data will further improve the efficiency of the plant.

A Digital Twin (DT) can be defined as a mirror image of a real world object presented in a digital format [11]. Unlike in digital shadows, the data flow in digital twins is two-way which means a change in the state of a physical product gets automatically updated to the virtual product and vice versa [11]. Digital Twins are for both before and after the physical system is built, before the system is built they

are there to estimate the cost of implications and the resources that are required [28], after the physical system is built they are used to foresee future faults and manage production [28].

References

1. Kuriakose RB, Vermaak HJ, Customized mixed model stochastic assembly line modelling using simulink, pp 2–7. <https://doi.org/10.5013/IJSSST.a.20.S1.06>
2. Liu Q et al (2021) Digital twin-based designing of the configuration, motion, control, and optimization model of a flow-type smart manufacturing system. *J Manuf Syst* 58(PB):52–64. <https://doi.org/10.1016/j.jmsy.2020.04.012>
3. Kuriakose R (2019) Optimization of a real time multi-mixed make-to-order assembly line to reduce positive drift. Central University
4. Zenisek J, Wild N, Wolfartsberger J, Zenisek J, Wild N, Wolfartsberger J (2021) Investigating the potential of smart manufacturing technologies investigating the potential of smart manufacturing technologies. *Proc Comput Sci* 180(2019):507–516. <https://doi.org/10.1016/j.procs.2021.01.269>
5. Wicaksono H, Ni T (2020) An automated information system for medium to short-term manpower capacity planning in make-to-order. *Proc Manuf* 52(2019):319–324. <https://doi.org/10.1016/j.promfg.2020.11.053>
6. Kuthambalayan TS, Bera S (2020) Managing product variety with mixed make-to-stock/make-to-order production strategy and guaranteed delivery time under stochastic demand. *Comput Ind Eng* 147:106603. <https://doi.org/10.1016/j.cie.2020.106603>
7. Zaremba LS, Smoleński WH (2000) Make-to-order versus make-to-stock in a production-inventory system with general production times. *Ann Oper Res* 97(1–4):131–141. <https://doi.org/10.1023/A>
8. Gupta D, Benjaafar S (2004) Make-to-order, make-to-stock, or delay product differentiation? A common framework for modeling and analysis. *IIE Trans Inst Ind Eng* 36(6):529–546. <https://doi.org/10.1080/07408170490438519>
9. Kuriakose R et al (2019) Optimization of a real time web enabled mixed model stochastic assembly line to reduce production time
10. Liu X, Yang X, Lei M (2021) Optimisation of mixed-model assembly line balancing problem under uncertain demand. *J Manuf Syst* 59(March):214–227. <https://doi.org/10.1016/j.jmsy.2021.02.019>
11. Singh M, Fuenmayor E, Hinchy EP, Qiao Y, Murray N, Devine D (2021) Digital twin: origin to future. *Appl Syst Innov* 4(2):1–19. <https://doi.org/10.3390/asi4020036>
12. Kuehn W (2018) Digital twins for decision making in complex production and logistic enterprises. *Int J Des Nat Ecodyn* 13(3):260–271. <https://doi.org/10.2495/DNE-V13-N3-260-271>
13. Raj ASV, Mathew J, Jose P, Sivan G (2016) Optimization of cycle time in an assembly line balancing problem. *Proc Technol* 25:1146–1153. <https://doi.org/10.1016/j.protecy.2016.08.231>
14. Dolgui A, Petroodi SEH, Kovalev S, Kovalyov MY, Thevenin S (2019) Workforce planning and assignment in mixed-model assembly lines as a factor of line reconfigurability: state of the art. *IFAC-PapersOnLine* 52(13):2746–2751. <https://doi.org/10.1016/j.ifacol.2019.11.623>
15. Kuriakose RB, Vermaak HJ (2018) A review of the literature on assembly line balancing problems, the methods used to meet these challenges and the future scope of study. *Adv Sci Lett* 24(11):8846–8850. <https://doi.org/10.1166/asl.2018.12359>
16. Azizoğlu M, İmat S (2017) Workload smoothing in simple assembly line balancing. *Comput Oper Res* 89:51–57. <https://doi.org/10.1016/j.cor.2017.08.006>
17. RashmiSarmah R (2019) A review on assembly line balancing. *Int J Adv Res* 7(9):465–470. <https://doi.org/10.21474/ijar01/9685>

18. El Ahmadi SEA, El Abbadi L, Belghiti MT (2019) A review paper on algorithms used for simple assembly line balancing problems in the automotive industry. In: Proceedings of the international conference on industrial engineering and operations management, no. July, pp 1840–1846
19. Sivasankaran P, Shahabudeen P (2017) Comparison of single model and multi-model assembly line balancing solutions. *Int J Comput Intell Res* 13(8):1829–1850. Available: <http://www.ripublication.com>
20. Leiber D, Hammerstingl V, Weiß F, Reinhart G (2019) Automated design of multi-station assembly lines. *Proc CIRP* 79:137–142. <https://doi.org/10.1016/j.procir.2019.02.029>
21. Kriengkarakot N, Pianthong N (2015) The assembly line balancing problem : review articles. *The Assembly Line Balancing Problem*
22. Schraknepper D, Stief P, Dantan J, Etienne A, Siadat A (2021) A new methodology to analyze the of functional and physical architecture existing products for an assembly oriented product family the concept of digital twin and digital shadow in manufacturing the concept Thomas. *Proc CIRP* 101:81–84. <https://doi.org/10.1016/j.procir.2021.02.010>
23. Mashaly M (2021) Connecting the twins: a review on digital twin technology & its networking requirements. *Proc Comput Sci* 184:299–305. <https://doi.org/10.1016/j.procs.2021.03.039>
24. Schwarz A, Ralph BJ, Stockinger M (2021) Planning and implementation of a digital shadow for the friction factor quantification of the ECAP process using a grey box modeling approach and finite element analysis. *Proc CIRP* 99:237–241. <https://doi.org/10.1016/j.procir.2021.03.035>
25. Tarasewich P (2003) Using ant techniques to solve the assembly line balancing problem parallel workstations, stochastic task durations, and mixed-models, pp 605–617. <https://doi.org/10.1080/07408170390214473>
26. Weida X, Xiao T (2008) Mixed model assembly line balancing problem with fuzzy operation times and drifting operations. In: Proceedings - Winter simulation conference, no. December 2008, pp 1752–1760. <https://doi.org/10.1109/WSC.2008.4736263>
27. Matanachai S, Yano CA (2001) Balancing mixed-model assembly lines to reduce work overload. *IIE Trans Inst Ind Eng* 33(1):29–42. <https://doi.org/10.1080/07408170108936804>
28. Gericke G, Kuriakose R, Vermaak H (2019) Design of digital twins for optimization of a water bottling plant

Fuzzy Logic-Based Cluster Head Selection an Underwater Wireless Sensor Network: A Survey



Hetal Panchal  and Sachin Gajjar 

Abstract Underwater wireless sensor network (UWSN) is a network of underwater sensor nodes (USN), surface buoy, anchor nodes, and autonomous utility vehicles (AUVs) or unmanned utility vehicles (UUVs) that sense the parameters of the underwater environment cooperatively. Underwater sensor network finds very important and useful applications in assisted navigation, offshore exploration, oceanographic data collection, pollution monitoring, disaster prevention, and tactical surveillance applications. UWSN faces many challenges like limited power supply, connectivity, computational constraints, storage, and underwater communication. To divide the network in the cluster and selection of cluster heads (CHs) is the feasible solution to deal with challenges. This paper provides a survey of fuzzy logic (FL)-based cluster head selection to deal with the challenges of UWSN. More specifically, aspects like clustering and routing, data aggregation, coverage, and security are dealt extensively. Different cluster head selection protocols are discussed with their advantages, disadvantages, and results.

Keywords Underwater wireless sensor network · Cluster head selection · Fuzzy logic

1 Introduction

Oceans occupy 75% portion of the earth [1]. In history, exploration of the underwater was performed manually. Real-time monitoring was not possible by this method [2]. An underwater wireless sensor network is formed by sensor nodes, anchor nodes, surface buoys, offshore base stations, and AUVs or UUVs that sense the parameters of the underwater environment like pressure, depth, temperature, etc. The data collected is sent to the surface buoys, which then send it to the offshore base station. [1]. Base station analyzes the data and takes appropriate actions if required. UWSNs are envisioned to enable applications for offshore exploration, oceanographic data collection, disaster prevention, pollution monitoring, assisted navigation, and tactical

H. Panchal (✉) · S. Gajjar
Department of Electronics and Communication Engineering, Institute of Technology, Nirma University, Ahmedabad, India
e-mail: 20ftphde47@nirmauni.ac.in

surveillance [3]. UWSN faces many challenges like low bandwidth, high bit error rate, large propagation delay, multi-path fading, attenuation, scattering, and Doppler effect due to the presence of water in the environment of communication [1]. Acoustic communication is preferable due to low attenuation in water.

However, UWSN nodes are constrained with limited power supply, limited storage, low communication bandwidth, and processing capabilities. So, UWSN faces challenges related to network lifetime and energy-efficient communication. Cluster-based routing is an efficient solution to the mentioned challenges [4]. Sensor nodes are divided into groups called clusters and cluster heads are selected to accumulate the data sensed by its cluster members (CMs). CHs collect, process, and pass on the data to the surface sink node. Computational intelligence (CI)-based techniques provide a better solution for CH selection. This paper surveys fuzzy logic to select CH.

Fuzzy logic is a mathematical tool that deals with the input uncertainties. In FL input parameters are fuzzy. The object of the fuzzy sets is a partial member of a set. The combination of fuzzy rules and fuzzy sets makes a knowledge base rule for a fuzzy logic system (FLS). The block diagram of the FLS is shown in Fig. 1.

Fuzzification is the process of mapping non-fuzzy inputs to their fuzzy representation. This includes the use of membership functions like trapezoidal, triangular, Gaussian, and so on. Knowledge base defines a relation between input and output with the If-Then rule with If antecedent, then consequent. In the inference process, fuzzified inputs are mapped to the rule base that produces a fuzzy output. In the defuzzification process, output of a fuzzy rule is converted into a crisp, non-fuzzy form.

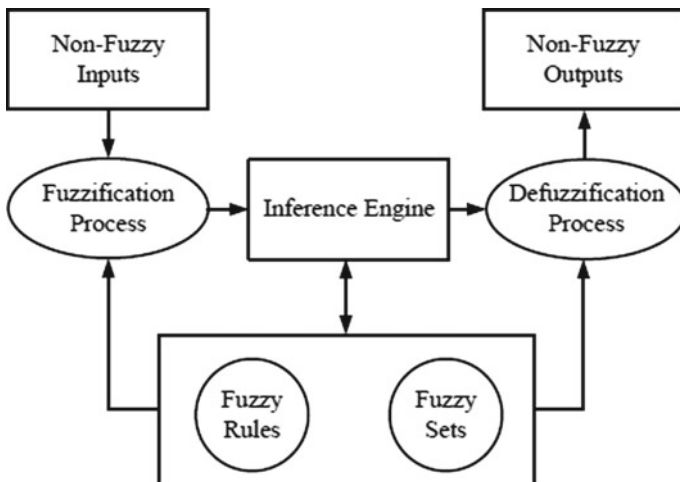


Fig. 1 Block diagram of the fuzzy logic system

The rest of this paper is organized as follows: CH selection protocols based on Fuzzy Logic in UWSN are presented in Sect. 2. Finally, conclusions and future work are discussed in Sect. 3.

2 Fuzzy Logic for Cluster Head Selection in Underwater Wireless Sensor Networks

Capabilities of FL are used in both technical and non-technical sectors such as image, voice, and signal processing; robotics, aerospace, and embedded system industries; and non-technical fields such as sales, business, and marketing. Here, FL is used for CH selection in UWSN and CH selection algorithms are discussed below [7].

In [8], FL is used to select the CH and cluster size. CH acts as a data aggregator node. The goal of this algorithm is to balance the power consumption of the node and extend the lifetime of the network. New fuzzy input parameters like a load on a node and link quality are introduced with distance to sink, residual energy, and node density. Data is sensed by CMs and sent to the CH. CH combines the data and pass on it to the surface sink node. The proposed protocol is compared with UCFIA [9]. In [9], the protocol deals with balancing the Energy Consumption (EC) of the WSN. Fuzzy input parameters are the energy level of the node, local density, and node's distance to the BS. Inter-cluster communication is done by ant colony optimization. The proposed protocol [8] has minimized delay, energy consumption and improved the throughput and delivery ratio. However, the Mobility of the nodes should be considered because underwater network topology is dynamic and nodes are not static due to water current.

In [10], CH selection and cluster size selection are determined by FL. Fuzzy input parameters are node density, residual energy, link quality, distance to sink and, load on a node. MARPCP [11] is used that divide the network into clusters and is used for Intra clustering communication. For inter-cluster communication, a hierarchical multi-path routing-LEACH method is utilized to determine numerous paths from source to destination. The path selection depends on the residual energy of the node, channel condition, and the number of hops. The proposed protocol is compared with MARPCP [11] and E²DTS [12]. In [12], the E²DTS technique was used to calculate both clock skew and offset. To determine its clock offset, local timestamps were transmitted. With minimal energy, high-level time synchronization precision was achieved. The synchronization error, on the other hand, is greater than the mean value. In the proposed protocol [10] energy consumption is reduced; end-to-end packet delay is reduced and packet delivery ratio (PDR) is improved. However, to construct energy-balanced networks the mobility of nodes is not considered. Packets collision and aggregation delay issue is there as effective scheduling techniques are needed.

In [13], failure of the node is considered in terms of depletion of energy of the node or node hardware failure. If CH fails, FL is used to select the backup cluster head (BCH) node. Fuzzy input parameters are distance to sink, node density, residual

energy, load on a node, and link quality. If BCH fails, any of CM is selected as BCH. In case of failure of CM, CH instructs BCH to transmit all the data of CM to the CH. Failed CM is replaced by a new node. When nodes are not transmitting goes to the sleep state and before going to the sleep state nodes send a message to CH about change in the state. PDR is improved as in case of failure of CH, BCH takes the responsibility to work as CH. So, energy consumption and delay are reduced. However, during the failure of CH and BCH, packet transmission is affected as failed CH and BCH will be no longer in the cluster as they are replaced by new CH and BCH. The proposed protocol is compared with EDETA [14], RCH [15], and SDMCGC [16]. EDETA [14] is a hierarchical protocol in which CHs are decided after clusters formation. To send the collected data from the remaining nodes to the sink node in a multi-hop fashion, The CHs form a tree structure among themselves. To give better scalability and some fault tolerance techniques, the protocol enables multiple sinks. In RCH [15], CH and BCH are selected as if CH fails BCH takes place the duties of CH. It detects the failure and initiates the recovery by identifying the reason for node failure. SDMCGC [16] is about connectivity and coverage of the final UWSN in which nodes are randomly deployed initially then by considering the connected dominating set of initial network depths of the nodes is adjusted for dominating node to minimize the coverage overlap and distance between nodes to BS. In the proposed protocol [13], mobility of the nodes should be considered because nodes are not in a static position due to water currents.

In [17], 3D UWSN architecture is used. The monitoring network is divided into layers. CH is selected in every layer. For CH selection, intuitionistic fuzzy analytic hierarchy process (AHP) and hierarchical fuzzy integration are used to have an unequal size of cluster topology. CH is selected based on service quality status, energy status, and node location status. The main criteria are again divided into two sub-criteria coverage factors, link reliability, residual energy, message cost, number of neighboring nodes, and depth of the nodes, respectively. Energy balancing routing is used for communication between clusters based on the number of neighbor nodes, the depth of the nodes, and the remaining energy of the nodes. The proposed protocol is compared with LEACH-coverage-U [18] and NULCPR [19]. In [18], the impact of sensing coverage on the network is solved by modifying the LEACH protocol and by using virtual grid topology. In [19], hierarchical coverage preserving protocol is described having unequal cluster size that forms a tree from sink node to other nodes in the downwards direction. CH is selected in each layer and it is connected to the upper layer node to continue the connectivity. The radius of CH is proportional and the density of nodes is inversely proportional to the layer number. The proposed protocol [17] improves the coverage of the network and data gathering reliability. Node mobility is considered. The number of CHs per round is stable as it depends upon network coverage and the number of neighbor nodes of CH in every layer. Again, the radius of the cluster is fixed in each layer and so it gives better stability. Computational complexity increases as CH evaluation depends upon a greater number of input parameters.

In [20], the fuzzy C means clustering (FCM) algorithm is developed for clustering purpose. CH is selected depending on the closeness of the cluster's center in the

initial round, then depending on the remaining energy of the node CH is selected. In this work, two routing algorithms are proposed: (i) single-hop fuzzy-based energy-efficient routing algorithm (SH-FEER) for UWSN. In SH-FEER, data is transmitted by CMs to CHs and CHs to UW-sink using single-hop communication. (ii) Multi-hop fuzzy-based energy-efficient routing algorithm (MH-FEER) for UWSN. Cluster formation and CH selection are similar to SH-FEER. MH-FEER transfers the data of CHs to UW-sink using multi-hop transmission and finds the shortest path. CHs are rotated in every round to balance the energy usage of the network. But, to choose CH, only residual energy is considered. With an increase in the number of rounds and a change in node density and node speed, the number of alive nodes falls and energy consumption rises as more nodes participate in packet transmission.

In [21], the fuzzy clustering algorithm is implemented for the formation of the cluster, and to select CH, particle swarm optimization (PSO) is used. Trust evaluation is performed by the FL means trusted CHs and trusted CMs are chosen using FL. The node's trust value is determined using fuzzy input parameters such as the energy, node's distance, and relative mobility. If the CH node's total trust value drops below the low trust value, the task of CH node is shifted to a node in the same cluster with a higher trust value. The block cipher design in UWASNs is precisely developed to keep packets secure with decreasing energy consumption. The lightweight XOR logical operation is used for secure transmission between CH and CMs that marks a good trade-off among overhead and privacy of the operation. Energy consumption is higher but it provides a balance among overhead and security. The proposed protocol is compared with CSLT [22] and TMCHV [23]. In [22], the authors address the development of the trust model that is security-based for localization. In this trust model, the nodes are fixed, and the transmission range is limited. In [23], the authors offer a secure model for establishing trust among sensor nodes in UWASNs that is based on trust. Calculation of trust value of each CH is equal to the combination of recommended trust and direct trust due to energy, communication, and collection of the data.

In [24], three FL systems are used. Clustering is initialized by the sink node to select CHs based on the depth of the node and the distance between the node and the sink node. The CHs employ the first FL system to determine the next eligible hop by considering the link quality and the node's residual energy. Based on energy, the distance between a node and a sink, and the depth of the node, the second FL system is used to find a CH if the energy of existing CHs falls below a threshold level. Because not all CMs are appropriate for becoming CH, the third FL is utilized to choose a relay node. The data is transferred toward the sink node through the relay node. The relay node is selected based on the node's distance from the CH, the node's average delay, the energy level of the node, and cluster density in the proximity of the candidate relay node. The proposed protocol is compared with IDACB [25]. In [25], with the help of clustering and data aggregation algorithms, as well as sleep-wake-up techniques, IDACB proposes an appropriate solution for collecting and transferring data to the base station. When a Time Division Multiple Access (TDMA)-based transmission mechanism is utilized, both intra-cluster and inter-cluster collisions are avoided in aggregated data transfer. In the proposed protocol [24] cluster is fixed and

only the cluster head is changed in every round. Node mobility is not considered. Link quality is considered for next-hop that increases the PDR.

In [26], clusters are formed using fuzzy C means clustering algorithms. The theoretical CH is located at the cluster's center. Depending upon the energy of the node as well as the distance between the theoretical CH and node, real CHs are selected. If the energy of CH is depleted, new CH is selected among the CMs of the cluster. New nodes are added to the network whenever the CM's energy is depleted. The size of the cluster is determined by probability which is a function of the radius of the cluster and distance between nodes. Reliability, flexibility, and scalability of the network are increased as new nodes can be added in case of node dies. Insertion of new nodes is a costly and biggest issue due to the unpredictable underwater environment. Single-hop intra-cluster communication and inter-cluster communication can increase energy consumption.

In [27], initial clusters are formed by K-means clustering using the expectation–maximization algorithm [28], and a mathematical model is developed to decide the probability of sensor node belongingness to form an initial cluster. The number of clusters is decided by the sum of squared parameters (SSE). CH selection is performed based on the distance and residual energy of the node in the hierarchical topology. Nodes are partitioned into layers depending on the distance between nodes and the sink node. CHs are data aggregator nodes of the network. Data transmission of the CHs is based on the Euclidian distance with similarity function [29]. CH aggregates the data from CMs called a vector. It compares two vectors and transmits only one vector if two vectors are the same. The proposed protocol is compared with LEACH [4]. It avoids the data redundancy which decreases energy usage and lengthens the network lifetime. Increased CH election can cause SSE problems, nodes are viewed as static, and QoS must be evaluated.

In [30], 3D architecture is deployed in 2D layer formation from the bottom to the surface. Each layer is subdivided into several hexagonal grids. SNs are exponentially distributed in the network. The grid size is decided depending on the range of the SN. CH is selected using FL and CH decides the cluster of multiple grids. CHs are chosen depending on distance, projection, residual energy of a node, and sensitivity. CH works as a forwarding node. Some nodes are fixed for data aggregation. CH transmits the data to aggregator nodes. Aggregator nodes transmit the data to the sink node. The proposed protocol is compared with DBR [5] and VBF [6]. Hexagon shapes of grids provide much coverage. End-to-end delay in transmission of packets is higher when the number of nodes in the network is lower.

In [31], 3D underwater architecture is considered for mobile ad hoc network for a disaster situation. Node sensing capability monitoring is done by the Boolean perceptual model. qV-ECC (qu-Vanstone-based elliptic curve cryptography) is used to register nodes to the surface sink. Based on the depth, nodes are divided into layers from the bottom surface to the sink node. Clustering of the nodes and CH selection is done by a Type-II FLS. Fuzzy input parameters are trust value, relative mobility, node buffer size, residual energy, and distance between neighbors. In the network, multiple AUVs are placed throughout and travel in an elliptical path to visit all CHs. The nearest AUV is chosen by CH for transmission of data using weight values like distance to the surface sink node, node projection, and residual energy of the node to the surface sink node. Pigeons swarm optimization (PiSO) is used for routing and it uses three parameters: distance to the depth of the ocean, node residual

energy, and relative mobility of node. Ciphertext stealing technique (CST) is used for secure packet transmission. Lightweight digital watermarking is investigated, that is an information hiding approach that costs less to compute than encryption. It is used for common data transmission by the firefly algorithm. The proposed protocol [31] is compared with HAMA [32] and ABSR [33]. In [32], Multi-AUVs follow a predetermined path in the network. The nodes closest to an AUV's trajectory deliver data directly to the AUV, while the remaining nodes transfer data to nodes nearer to the trajectory. In case of failure of an AUV, while communicating with nodes at the time of collecting data, malfunction detection and repair techniques are utilized to keep the network running smoothly. In [33], by locating wormhole robust secure neighbors and routing information through the secure path, the proposed agent-based secured routing strategy improves the quality of service. This strategy employs four agencies: routing, security, underwater vehicles, and gateways, all of which are comprised of static and mobile agents. Type-II FLS has a fuzzy membership function that effectively deals with any source of uncertainty. Computational complexity is more cumbersome and complex as it possesses a three-dimensional membership function. The mobility of SNs is not considered.

In [34], the underwater sensing network is divided into equal-sized rings depending upon the radius and diameter of the area. In each ring, underwater cluster heads (UCHs) are chosen randomly for Intra-cluster routing. For inter-cluster routing, among all the UCHs, the best UCH is selected to transmit the data from nodes to sink nodes. For that suboptimal and optimal UCHs are selected based on fuzzy input parameters like the load on a node, residual energy, link quality, node density, and distance between UCH and sink node. Suboptimal UCHs select one best optimal UCH among them using UCH rank and pass on the collected data to the sink node. UCH rank represents the number of suboptimal UCHs between the sink node and optimal UCH. For each UCH, a unique PRN is generated and broadcasted, and depending upon the PRN, UCH performs inter-cluster communication.

The proposed protocol [34] is compared with DBR [5], EEDBR [35], and MLCEE [36]. In EEDBR [35], depth-based routing is done by considering the residual energy of the sensor node. In MLCEE [36], the hotspot problem is resolved by splitting the network into layers. Clusters are formed and CHs are selected in each layer except the layer near the sink node. Multi-hop communication is established between nodes depending on the fitness of the nodes. The average end-to-end delay and energy consumption of the proposed protocol is lesser than DBR, EEDBR, and MLCEE due to the fuzzy cluster-based routing and priority-based routing. Data aggregation is done by UCH using PRN reduces the collision, increases the throughput, and increases the packet delivery ratio of the proposed protocol. Routing is done by using fuzzy logic by selecting appropriate UCH as a forwarding node increases the network lifetime of the proposed protocol. Due to the complex process of cluster head selection for data aggregation and routing the data energy consumption and overhead increase. A comparison of protocols based on the FL in UWSN is given in Table 1.

Table 1 Comparison of fuzzy-based cluster head selection in UWSN

Refs.	Input parameters to fuzzy logic	Simulation tool	Results
[8]	(i) Residual energy	NS-2	(i) PDR is 47% higher than UCFLA [9]
	(ii) Distance to sink		(ii) Delay is 5% lesser than UCFLA [9]
	(iii) Node density		(iii) Throughput is 11% higher than UCFLA [9]
	(iv) Load on a node		(iv) EC is 6% of less than UCFLA [9]
	(v) Link quality		
[10]	(i) Residual energy	NS-2	(i) PDR is 67% and 36% higher than MARPCP [11] and E ² DTS [12], respectively
	(ii) Distance to sink		(ii) Delay is 66% and 36% less than MARPCP [11] and E ² DTS [12], respectively
	(iii) Node density		(iii) EC is 67% less than MARPCP [11] and 33% less than E ² DTS [12]
	(iv) Load on a node		
	(v) Link quality		
[13]	(i) Residual energy	NS-2	(i) PDR is 17, 41 and 60% higher than EDETA [14], RCH [15] and SDMCGC [16]
	(ii) Distance to sink		(ii) Delay is 33, 69 and 63% lesser than EDETA [14], RCH [15] and SDMCGC [16]
	(iii) Node density		(iii) Packet drop is 7, 20 and 36% lesser than EDETA [14], RCH [15] and SDMCGC [16]
	(iv) Load on a node		(iv) EC is 45% lesser than EDETA [14], 66% lesser than RCH [15] and 23% than SDMCGC [16]
	(v) Link quality		
[17]	(i) Residual energy	MATLAB	(i) First node died time of DHCDGA is about 18.71% longer and last node died time of DHCDGA is about 15.38% longer than that of NULCPR [19]
	(ii) Quality of service		(ii) Number of alive nodes per round, Average remaining energy, number of cluster heads per round, network coverage ratio, and amount of data packets received of DHCDGA is higher than NULCPR [19] and leach-coverage-U [18]
	(iii) Location		

(continued)

Table 1 (continued)

Refs.	Input parameters to fuzzy logic	Simulation tool	Results
[20]	Closeness to CH	MATLAB	<p>(i) MH-FEER is 32.78 and 63.07% more efficient in terms of network lifetime compared to SH-FEER and direct transmission when nodes are static</p> <p>(ii) MH-FEER is 8.5 and 39.04% more efficient in terms of network lifetime compared to SH-FEER and direct transmission when nodes are mobile</p>
[21]	(i) Distance	MATLAB	(i) Malicious node detection ratio is higher than CSLT [22] and TMCHV [23] when malicious nodes vary in between 1 and 5 and time for simulation increases
	(ii) Energy		(ii) Number of packets delivered by the proposed protocol is higher and the delay is lower than CSLT [22] and TMCHV [23] for the number of attackers
	(iii) Relative mobility of the node		
[24]	(i) Depth of the node	NS-2	(i) Average EC of the algorithm is 7% better, the variance of remaining energy of nodes is 29% better and normalized routing is 0.47% better than IDACB [25] protocol
	(ii) Distance		(ii) Energy efficiency is 3.86% higher than IDACB [25]
	(iii) Link quality		(iii) PDR is better than IDACB [25]
	(iv) Residual energy		(iv) Number of packets lost is 12% better than IDACB [25]
	(v) Delay		
[26]	(i) Distance	MATLAB	(i) The energy consumption link between different numbers of theoretical and real clusters is evaluated. The result shows that when the number of clusters is 4 and 5, EC for real and theoretical clusters is minimum
	(ii) Residual energy		
[27]	<p>(i) Distance</p> <p>(ii) Residual energy</p>	MATLAB	(i) Number of dead nodes to the number of rounds in the proposed protocol is lesser than LEACH [4] protocol

(continued)

Table 1 (continued)

Refs.	Input parameters to fuzzy logic	Simulation tool	Results
			(ii) EC of the CHs for load using data aggregation is lesser than without data aggregation in the proposed protocol
[30]	(i) Distance	MATLAB	(i) With the increase in the number of nodes hexagon deployment requires less CH compared to the triangular and circular deployment
	(ii) Projection		(ii) PDR and energy efficiency of FBR is higher than VBF [6] and DBR [5] to an increase in the number of SNs in the network
	(iii) Sensitivity		(iii) When the number of nodes is less, the end-to-end delay of packet delivery of DBR [5] and VBF [6] is less compared to FBR but when the number of nodes increases DBR [5] and VBF [6] provides high delay than FBR
	(iv) Residual energy		
[31]	(i) Trust value	NS-3	(i) PDR versus number of nodes for E ² -SCRCP is 100%, HAMA [32] is 93% and ABSR [33] is 90%
	(ii) Distance between neighbors		(ii) EC versus the number of nodes of E ² -SCRCP is lesser than HAMA [32] and ABSR [33]
	(iii) Relative mobility		(iii) End-to-end delay versus number of nodes is 110% for E ² -SCRCP, 135% for HAMA [32], and 145% for ABSR [33]
	(iv) Node buffer size		(iv) Security strength E ² -SCRCP versus number of nodes is 80% and ABSR [33] is 56.81%
	(v) Residual energy		(v) Throughput of E ² -SCRCP versus the number of nodes is 200 kbps, HAMA [32] is 125 kbps and ABSR [33] is 150 kbps
[34]	(i) Residual energy	NS-2	(i) The average end-to-end delay and energy consumption of the proposed protocol is lesser than DBR [5], EEDBR [35], and MLCEE [36]
	(ii) Load		(ii) Average packet delivery ratio, network lifetime, and average throughput of the proposed protocol is higher than DBR [5], EEDBR [35], and MLCEE [36]
	(iii) Link quality		
	(iv) Distance		
	(v) Node density		

3 Conclusion

Recent implementations of CI techniques in varied dynamical UWSNs are the subject of this detailed study. Fuzzy logic for CH selection in sensor networks is briefly discussed. Depending upon the input parameters like residual energy, hop distance to sink, link quality, trust value, relative mobility, sensitivity, the density of nodes, etc. CH selection and routing are performed. FL is leading to an optimized solution with lower energy consumption, longer network lifetime, greater security, fault management, and network stability. The future work aims to design the cluster head selection protocols using a Type-II fuzzy logic system that deals with input uncertainties with accuracy. Cluster head selection can be done using CI approaches such as ant colony optimization and particle swarm optimization, which reduce energy consumption by providing optimal solutions.

References

1. Felemban E, Shaikh FK, Qureshi UM, Sheikh AA, Qaisar SB (2015) Underwater sensor network applications: a comprehensive survey. *Int J Distrib Sensor Netw* 11(11):1–14
2. Sozer EM, Stojanovic M, Proakis JG (2000) Underwater acoustic networks. *IEEE J Oceanic Eng* 25(1):72–83
3. Akyildiz IF, Pompili D, Melodia T (2005) Underwater acoustic sensor networks: research challenges. *Ad Hoc Netw* 3(3):257–279
4. Heinzelman WB, Chandrakasan AP, Balakrishnan H (2002) An application-specific protocol architecture for wireless microsensor networks. *IEEE Trans Wirel Commun* 1(4):660–670
5. Yan H, Shi ZJ, Cui JH (2008) DBR: depth-based routing for underwater sensor networks. In: Das A, Pung HK, Lee FBS, Wong LWC (eds) NETWORKING 2008 ad hoc and sensor networks, wireless networks, next generation internet. NETWORKING 2008. LNCS, vol 4982. Springer, Berlin, Heidelberg, pp 72–86
6. Xie P, Cui JH, Lao L (2006) VBF: vector-based forwarding protocol for underwater sensor networks. In: Boavida F, Plagemann T, Stiller B, Westphal C, Monteiro E (eds) Networking technologies, services, and protocols; performance of computer and communication networks; mobile and wireless communications systems. NETWORKING 2006. LNCS, vol 3976. Springer, Berlin, Heidelberg, pp 1216–1221
7. Amit K (2005) Computational intelligence: principles, techniques, and applications. Springer, Heidelberg
8. Goyal N, Dave M, Verma AK (2014) Fuzzy based clustering and aggregation technique for under water wireless sensor networks. *Int Conf Electron Commun Syst*:12–16
9. Mao S, Zhao CL (2011) Unequal clustering algorithm for WSN based on fuzzy logic and improved ACO. *J China Univ Posts Telecommun* 18(6):89–97
10. Goyal N, Dave M, Verma AK (2016) Energy efficient architecture for intra and inter-cluster communication for underwater wireless sensor networks. *Wirel Pers Commun* 89(2):687–707
11. Kim D, Wang W, Ding L, Lim J, Oh J, Wu W (2010) Minimum average routing path clustering problem in multi-hop 2-D underwater sensor networks. *Optim Lett* 4(3):383–392
12. Li Z, Guo Z, Hong F, Hong L (2013) E²DTS: an energy efficiency distributed time synchronization algorithm for underwater acoustic mobile sensor networks. *Ad Hoc Netw* 11(4):1372–1380
13. Goyal N, Dave M, Verma AK (2018) A novel fault detection and recovery technique for cluster-based underwater wireless sensor networks. *Int J Commun Syst* 31(4):1–15

14. Climent S, Capella JV, Meratnia N, Serrano JJ (2012) Underwater sensor networks: a new energy-efficient and robust architecture. *Sensors* 12(1):704–731
15. Ovaliadis K, Savage N, Tsiantos V (2014) A new approach for a better recovery of cluster head nodes in underwater sensor networks. In: 2014 international conference on telecommunication and multimedia, pp 167–172
16. Senel F, Akkaya K, Erol-Kantarci M, Yilmaz T (2015) Self-deployment of mobile underwater acoustic sensor networks for maximized coverage and guaranteed connectivity. *Ad Hoc Netw* 34:170–183
17. Song X, Sun W, Zhang Q (2020) A dynamic hierarchical clustering data gathering algorithm based on multiple criteria decision making for 3D underwater sensor networks. *Complexity* 2020:1–14
18. Tsai YR (2006) Coverage-preserving routing protocols for randomly distributed wireless sensor networks. *GLOBECOM IEEE Glob Telecommun Conf* 6(4):1240–1245
19. Jiang P, Wang X-M (2016) Network layered coverage preserving routing algorithm for underwater sensor networks. *Acta Elect Ronica Sinica* 44(5):1240–1246
20. Souiki S, Hadjila M, Feham M (2015) Fuzzy based clustering and energy efficient routing for underwater wireless sensor. *Int J Comput Netw Commun* 7(2):33–44
21. Krishnaswamy V, Manvi SS (2021) Trusted node selection in clusters for underwater wireless acoustic sensor networks using fuzzy logic. *Phys Commun* 47:101388
22. Han G, Liu L, Jiang J, Shu L, Rodrigues JJPC (2016) A collaborative secure localization algorithm based on trust model in underwater wireless sensor networks. *Sensors (Switzerland)* 16(2):229
23. Goyal N, Dave M, Verma AK (2017) Trust model for cluster head validation in underwater wireless sensor networks. *Underw Technol* 34(3):107–114
24. Tavakoli J, Moghim N, Amini L (2021) Pasandideh: a fuzzy based energy efficient clustering routing protocol in underwater sensor networks. *J Commun Eng* 9(1):154–167
25. Goyal N, Dave M, Verma AK (2017) Improved data aggregation for cluster-based underwater wireless sensor networks. *Proc Natl Acad Sci India Sect A* 87:235–245
26. Wang F, Wang L, Han Y, Liu B, Wang J, Su X (2014) A study on the clustering technology of underwater isomorphic sensor networks based on energy balance. *Sensors (Switzerland)* 14(7):12523–12532
27. Krishnaswamy V, Manvi SKS (2019) Clustering and data aggregation scheme in underwater wireless acoustic sensor network *Telkomnika*. *Telecommun Comput Electron Control* 17(4):1604–1614
28. Nasser S, Alkhalidi R, Vert G (2006) A modified fuzzy K-means clustering using expectation maximization. In: 2006 IEEE international conference on fuzzy systems, pp 231–235
29. Tran KTM, Oh SH, Byun JY (2013) Well-suited similarity functions for data aggregation in cluster-based underwater wireless sensor networks. *Int J Distrib Sens Netw* 2013:1–7
30. Sahana S, Singh K (2019) Fuzzy based energy efficient underwater routing protocol. *J Discr Mathematical Sci Cryptogr* 22(8):1501–1515
31. Yadav AK, Kush A (2019) E²-SCRIP: an energy-efficient secure cluster-based routing protocol for 3d underwater acoustic MANET (UWMANET). *Int J Innov Technol Explor Eng* 8(11):2430–2442
32. Han G, Long X, Zhu C, Guizani M, Zhang W (2020) A high-availability data collection scheme based on multi-AUVs for underwater sensor networks. *IEEE Trans Mobile Comput* 19(5):1010–1022
33. Manvi SS, Bharmagoudra MR (2017) Agent-based secure routing for underwater acoustic sensor networks: agent-based secure routing for UWASNs. *Int J Commun Syst* 30(13):1–19
34. Natesan S, Krishnan R (2020) FLCEER: fuzzy logic cluster-based energy-efficient routing protocol for underwater acoustic sensor network. *Int J Inf Technol Web Eng* 15:76–101

35. Wahid A, Lee S, Jeong HJ, Kim D (2011) EEDBR: energy-efficient depth-based routing protocol for underwater wireless sensor networks. In: Kim T, Adeli H, Robles RJ, Balitanas M (eds) *Advanced computer science and information technology. AST 2011. Communications in computer and information science*, vol 195. Springer, Berlin, Heidelberg
36. Khan W, Wang H, Anwar MS, Ayaz M, Ahmad S, Ullah I (2019) A multi-layer cluster based energy efficient routing scheme for UWSNs. *IEEE Access* 7:77398–77410

Improving the Efficiency of Forecasting Sports Events Using a Cascade of Neural Networks



Vasily Meltsov , Alexander Krutikov , and Dmitry Strabykin 

Abstract In addition to classical methods of statistical analysis, software systems based on the neural network approach have been widely used recently to solve forecasting and planning problems. In this paper, the application of various neural network models for predicting events in individual and team sports is considered. To solve the existing problems, the authors propose an approach based on cascading several types of neural networks. The results of processing input information by modules of the first level serve as data sets for modules of the second tier. To assess the correctness and effectiveness of the principle of cascading modules, experiments were conducted to predict a well-known sporting event—the fight between D. Wilder and T. Fury for the WBC world title. The experimental results showed that when using a cascade of neural networks, the prediction accuracy increases from 66% (GRNN network) to almost 93%. In addition, observation of the data vectors of the second and subsequent levels allows us to determine predictive factors—parameters that most significantly affect the final forecast. With the increase in the dimension of the source data arrays, the use of other neural network models, as well as the use of more complex configurations of the proposed system, it is possible to significantly increase the accuracy of forecasts with recommendations on the specifics of setting up such a complex.

Keywords Artificial neural network · Sports forecasting · Training sampling · Vector quantization · Cascading · GRNN network · LVQ network · Software system

V. Meltsov (✉) · A. Krutikov · D. Strabykin
Vyatka State University, Moskovskaya str. 36, Kirov 610000, Russia
e-mail: meltsov69@mail.ru

A. Krutikov
e-mail: usr09603@vyatsu.ru

D. Strabykin
e-mail: strabykin@vyatsu.ru

1 Introduction

Today, information technology (IT) and artificial intelligence (AI) technologies are perhaps the most dynamically developing areas of science and technology. The versatility and efficiency of IT tools make it possible to use them to solve a wide range of applied tasks in almost all spheres of human activity [1, 2]. One of the most important issues in solving many applied tasks is the issues of planning and forecasting.

In this paper, forecasting in the field of individual and team sports is considered. Forecasting in this case is aimed at obtaining knowledge about sports events that should occur in the future, that is, to obtain probabilistic information. This event can be not only the quantitative result of the athlete (team) shown at specific starts or his place in the overall standings of the tournament, but also the indicators of the “medal standings” of the national team of the country at major sporting events, as well as trends in the development of a particular sport.

According to time characteristics, short-term, medium-term, long-term and ultra-long-term forecasting can be distinguished. The main difficulty of forecasting in sports is that the result of the forecast can be either some numerical value that determines the result in a particular sports discipline (of various formats, accuracy and data types), where it is provided for by the regulations, or different types of outcomes of sports matches (victory, defeat, draw, knockout, athlete’s refusal, etc.). And sometimes, as a result of the forecast, it is necessary to form a trend or trend of changes in indicators (parameters) for a certain period of time, including in graphical form. To carry out such forecasting, both classical methods of statistical analysis are used, for example, time series analysis, and modern information technologies—systems based on fuzzy logic, various models of artificial neural networks, deep learning methods, etc. [2].

In one of the authors’ previous publications on this topic [3], a module designed to predict results in individual and team sports based on a neural network of vector quantization of signals is described in detail. In this paper, we will talk about the possibility of modifying such a module in order to improve the efficiency of the system and the accuracy of the predicted results.

2 Cascading Modules of a Neural Network System

The scientific staff of the laboratory “Intelligent Systems” named after V. A. Baykov of Vyatka State University has developed a software prototype of a system based on artificial neural networks designed to predict sports results [3]. To check the correctness and quality of the system, its modules were tested on various types and types of sports forecasts.

As the mathematical basis of the specified forecasting system, the principle known as “classification assignment” is taken [4]. In the classification problem, the training sample is a set of individual objects $X = \{x[i]\}$, where i varies from 1 to n . The

sample is characterized by a vector of real-valued features $x_i = \{x[i, 1], \dots, x[i, d]\}$. The initial object x is the variable t , which takes a finite value, usually from the set $T = \{1, \dots, 1\}$. It is required to build an algorithm (classifier) that, according to the vector of features x , would return a class label or a vector of membership estimates (a posteriori probabilities) to each of the classes $\{p(s|x)\}$, $s = 1, 1$.

In relation to sporting events, the classification task is the task of determining the outcome of a sporting event, that is, assigning it to a certain class (for example, class 1—victory, class 2—defeat, class 3—draw). At the same time, it is necessary to solve the problems of regression analysis—determining the potential numerical result of an athlete (time of a race or swim in seconds, the maximum weight of a raised barbell in kilograms, the result of a jump in centimeters, etc.).

Taking into account the above, the tasks of sports forecasting can be formulated as follows. The training sample is a set of measurements $X = \{x[i]\}$, $i = 1, n$, representing a vector of real-valued quantities $x[i] = (x_1[i], \dots, x_d[i])$ made at certain points in time. It is required to construct an algorithm (predictor) that would return a point estimate $\{x'[i]\}$ $i = n + 1, n + q$, confidence interval $\{(x_{-}[i], x + [i])\}$, where i is from $n + 1$ to $n + q$, or a posteriori distribution $p(x[n + 1], \dots, x[n + q]|x[1], \dots, x[n])$ of the forecast for a given depth q . Unlike the regression recovery task, here the forecast is made by time, and not by signs.

In numerous experiments on the previously presented software system [3], various models of neural networks were used. The analysis of the results showed that the modules based on the generalized regression neural network (GRNN) [5] and the neural network with radial basis functions (RBF) cope most successfully with the tasks of predicting numerical sports results [6]. The structure of the generalized regression neural network is shown in Fig. 1.

Modules with a cascade neural network (CNN) [7] and a direct propagation neural network (FNN) [8] also showed good accuracy. But on the tasks of determining the “non-numerical” outcomes of sports events (victory, draw, defeat, etc.), the module based on the neural network of vector quantization of signals (LVQ) demonstrates the best indicators [3, 9, 10].

The main thing that experiments have shown is that different models of neural networks can be effective for different types and types of forecasts. The second conclusion is that the main feature of forecasting tasks in sports is the aggregation of classification and regression analysis tasks. In most cases, it is necessary not only to predict the outcome of a sporting event, but also to approximate some numerical values for it—the time of the winner of the race, the weight of the barbell lifted by the champion, and the number of blows inflicted by the competitors. For example, for a boxing match, the likely result may be the victory of one of the fighters on points—116:112.

As one example, let's consider an experiment on predicting the result of a boxing match between American Deontay Wilder and Briton Tyson Fury for the WBC world title. Before this sporting event, the bookmakers' favorite was Tyson Fury (odds of 1.70 versus 3.20 for Deontay Wilder), a draw was considered almost impossible (odds of 30.00).

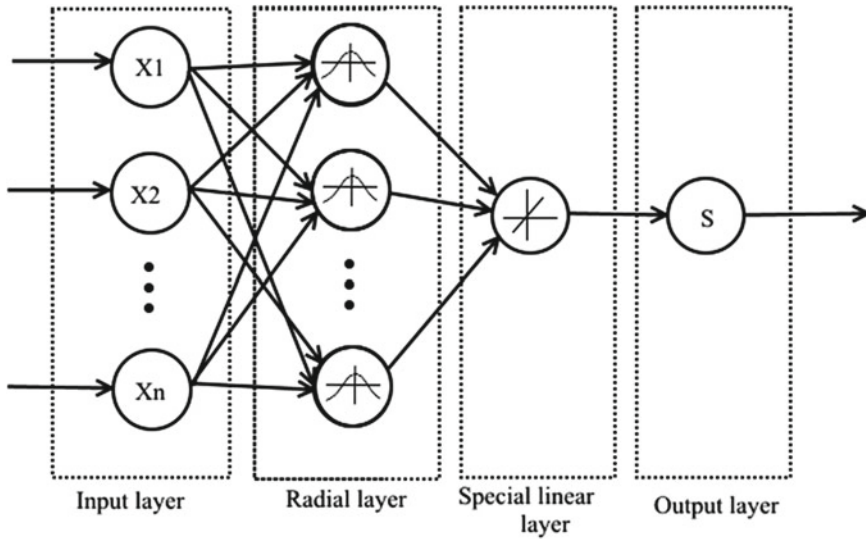


Fig. 1 GRNN network structure

A training sample consisting of a special set of vectors was developed to train neural networks and make a forecast. Each vector contained source data taken from open Internet sources [11]. As the main analyzed characteristics, both classical statistical parameters (the number of fights conducted by the athlete; the number of victories; the number of knockout victories; height; weight, etc.) and specific data were selected—the style of fighting, the main stance of the athlete, the size of the arms, etc. The result of training and forecasting is also a vector containing a set of elements corresponding to one of the possible outcomes of a given sporting event, i.e., one of the classes: (1—victory of the first boxer, 2—victory of the second boxer, 3—draw).

The MATLAB environment is used as a tool for designing a multimodule forecasting system. With the help of the Neural Networks Toolbox package, both software modules are implemented—both on the basis of the GRNN network and on the basis of the LVQ network. The models are trained using a special training sample described above.

Table 1 shows the sample results of predicting the outcome of the battle obtained during testing of the LVQ network (LVQ2.1 learning algorithm). The resulting vector is interpreted as follows: the vector (1,0,0) means a victory for Deontay Wilder, (0,1,0)—Tyson Fury, (0,0,1)—a draw.

The fight, which took place on October 9, 2021, ended with the victory of British boxer Tyson Fury [11]. That is, when analyzing all the known initial parameters before the fight, the vector quantization neural network gave an incorrect forecast for this sporting event in more than 78% of cases.

Data on testing the module based on a generalized regression neural network are shown in Table 2. The result of the forecast is interpreted as follows: the value “1”

Table 1 The result of the vector quantization neural network (LVQ2.1 learning algorithm)

Number of neurons	Number of training epochs	The result of the forecast
1	300	(0,0,1)
10	300	(0,1,0)
20	300	(1,0,0)
100	300	(0,1,0)
190	300	(0,1,0)
200	300	(1,0,0)
380	300	(0,1,0)
420	300	(1,0,0)
480	300	(1,0,0)
550	300	(1,0,0)
940	300	(1,0,0)
2200	300	(1,0,0)
3100	300	(1,0,0)

Table 2 The result of the generalized regression neural network

Target error	The result of the forecast
0.001	2
0.01	2
0.1	2
1.0	2
1.1	2
10	1
100	1
1000	1
10,000	1
100,000	1
0.00001	2
0.0001	2

means the victory of Deontay Wilder, “2”—Tyson Fury, and “3”—a draw.

In more than 64% of the experiments, the generalized regression neural network correctly identified the winner of this battle.

Thus, experiments on a prototype of a software system have shown that the most effective of the studied modules for solving such complex problems will be a module based on a generalized regression neural network. It was this option that showed the most qualitative forecasts of the complex result. However, the analysis of testing on a large set of different types of tasks revealed four serious problems.

First, even the GRNN network does not give a sufficiently accurate result when solving such non-trivial tasks as predicting the results of various martial arts and numerous metrics of team competitions in game sports. During the experiments conducted on a number of forecasts, the GRNN-based software module showed only 66% correct answers. The closest indicator to this was demonstrated by a software module based on the LVQ network—61% of the “guessed” results.

Secondly, the use of a high target error is not beneficial when training networks of this type, since with such an increase in the base parameter, the minimum root-mean-square error of training also increases.

Third, the structure of a generalized regression neural network has some features, and during training, as many neurons of the radial layer are formed as the vectors of the training sample [12]. Consequently, with a large number of vectors, the neural network will have an extremely complex structure, which, in turn, will occupy a significant amount of memory (a significant amount of crystal, when implemented on FPGA) [13].

And, fourth, it should be noted that a large-scale sample may contain large sets of different types of initial parameters [14] that affect the predicted result, and it is either impossible or extremely difficult to determine the most important and key parameters. To do this, it will be necessary to attract qualified specialists in the analyzed profile sports discipline.

One of the possible solutions to these problems is the use of the method of “cascading” modules in the forecasting system (by analogy with the cascade model of NN [15], as well as cascade filtering of data using Kohonen networks [16]), thus forming a specialized pipeline. Figure 2 shows the generalized structure of the system using GRNN networks on the first tier and LVQ networks on the second tier. All networks are trained separately, before they are included in the working cascade, each based on its own specialized training sample. The initial data (formed vectors of the analyzed parameters of the sports event) are fed to inputs 1 and 2 of the GRNN networks. The results of processing this information, also in the form of vectors, serve as data sets for the modules of the second tier. In the case under consideration—for the LVQ network module.

Training samples(s) are developed for each module separately. If this is a known or previously developed large sample, then it is decomposed in accordance with the types of models and the selected structure of the cascade of networks. The principle of sampling for each element of the cascade is similar to the one described above, but there is a “logical connection” between the analyzed parameters in the samples, determined by the subject area in which the prediction is made. An example of the relationship of elements of vectors of training samples is shown in Fig. 3. The elements of the vector of the second layer are the “forecast” of input elements for modules in subsequent layers of the cascade.

Sample vector #1 is a vector that contains general statistical, anthropometric and age indicators of boxers. Sample vector #2 is the average statistical indicators of a sporting event that are noted in statistical systems. In this example, the elements of the second layer are the percentage of accurate hits, thrown punches of the first and second boxer, etc.

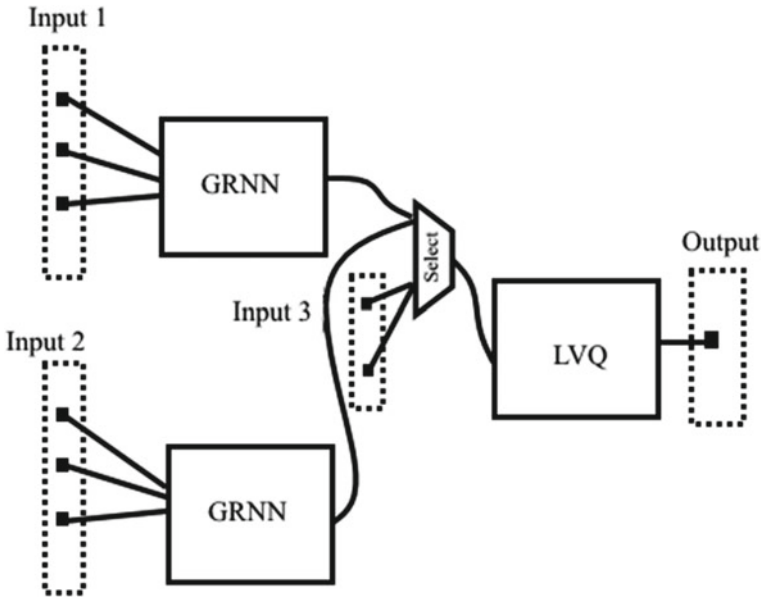


Fig. 2 Cascading neural networks

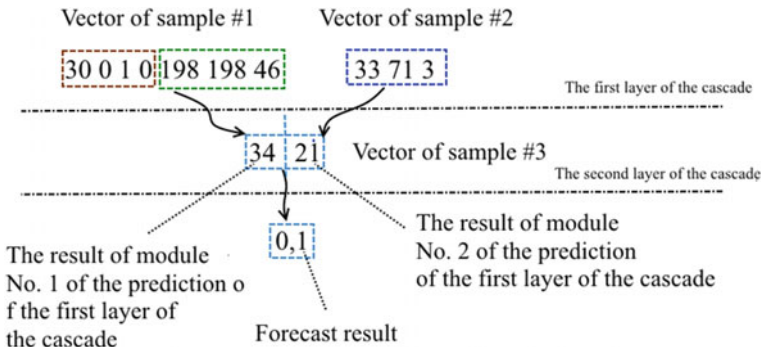


Fig. 3 The relationship of elements in the sample by cascade levels

The result of the work of the last module will be an output vector with an answer about the probability of the outcome of the declared sports event. An important point in this approach is the fact that the results of preprocessing obtained at the previous stages can serve as training samples for all modules of the system (except for the modules of the first tier). It is noted that in this case, at the intermediate stages, it is possible to determine the most important parameters of the upcoming sports event (predictive factors) and include only them in the training sample for the next tier. Thus, it is possible not only to minimize the sets of analyzed parameters, but also to reduce the training time of neural network models.

Table 3 The result of the forecast, using a cascade of neural networks

Number of neurons	Number of training epochs	The result of the forecast
100	300	(1,0,0)
130	300	(0,1,0)
150	300	(0,1,0)
380	300	(0,1,0)
420	300	(0,1,0)
480	300	(0,1,0)
550	300	(0,1,0)

The considered structure was implemented in the MATLAB environment. To assess the correctness and effectiveness of the principle of cascading modules, experiments were conducted to predict the same sporting event—the fight between Wilder and Fury for the WBC world title. The main results of testing the new structure are presented in Table 3.

The table shows that when using a cascade of neural networks, the prediction accuracy increases, almost reaching 93%. However, to achieve such high accuracy requires significant time spent on training network models. This is due to the fact that the modules are currently being trained sequentially, for each network separately. Data dependencies between modules do not allow parallelizing the learning process.

A similar example of the implementation of the forecast will be given in Table 4. In this case, the result of predicting the fight between Sergey Kovalev and Saul Alvarez is given, demonstrating similar results, which is 11% higher than when predicted by a separate LVQ network.

Since all the information for the formation of training samples is taken from open sources, the samples turned out to be quite general, we can say, enlarged. With the specified structure of the system (only two tiers) and the proposed cascading option, the experimental results obtained look quite optimistic. With the increase in the dimension of the source data arrays, the use of other neural network models, as well as the use of more complex configurations of the proposed system, it is possible to

Table 4 The result of the forecast, using a cascade of neural networks

Number of neurons	Number of training epochs	The result of the forecast
90	450	(1,0,0)
160	450	(1,0,0)
190	450	(0,1,0)
440	450	(0,1,0)
660	450	(0,1,0)
780	450	(0,1,0)
910	450	(0,1,0)

significantly increase the accuracy of forecasts with recommendations on the features of configuring such a complex and the features of forming training samples for each cascade of the neural network.

3 Conclusion

Thus, the results of the conducted research confirm that neural networks can be successfully used to predict the outcome of events not only in team, but also in individual sports. When using a single module of the generalized regression network, more or less reliable forecasts were obtained only with a target error of more than 100. The module based on the trained LVQ network, in almost half of the cases, gave the wrong result. With an increase in the number of neurons and other ways to increase the complexity of the model, the training time significantly increased.

The approach considered in the paper with cascading modules in the system leads to a significant increase in the accuracy of forecasts. However, it should be remembered that the formation of a pipeline from various models must be carried out taking into account the type and type of forecasting, the features of the formation of training samples for all models, the definition and allocation of the most important (predictor) parameters into special samples that affect the course and results of forecasting more than others. It should also be remembered that when using a cascade of neural networks, the time of training models can significantly increase if this process is carried out sequentially for each stage of the pipeline.

The use of the software system by the coaching staff or personal trainers of athletes will help to increase the effectiveness of training, to assess the level of indicators of control tests to achieve the necessary competitive result. At the moment, experiments are continuing in such disciplines as athletics and cross-country skiing. The preparation of specialized training samples has shown that even within the framework of one discipline, for example, athletics, it is necessary to take into account a huge number of different types of testing of athletes, the characteristics of their anthropometric data, psychological stability, pharmacological program, the number of fees, the remoteness of the venues of competitions, etc. All this leads to the need to select the most effective models of well-known artificial neural networks and various options for their interaction. It is obvious that the complication of each module and the structure of the system as a whole, in order to increase the accuracy of the forecast, leads not only to an increase in training time, but also to a decrease in the speed of the software system. The use of a cascade of neural networks is almost guaranteed to increase the forecast generation time.

References

1. Strabykin D, Meltsov V, Dolzhenkova M, Chistyakov G, Kuvaev A (2016) Formal verification

- and accelerated inference. In: 5th computer science on-line conference, CSOC-2016. Advances in intelligent systems and computing, vol 464. Prague, pp 203–211. https://doi.org/10.1007/978-3-319-33625-1_19
2. Szadkowski R, Drchal J, Faigl J (2019) Basic evaluation scenarios for incrementally trained classifiers. In: Tetko I, Kůrková V, Karpov P, Theis F (eds) Artificial neural networks and machine learning, ICANN 2019, LNCS, vol 11728. Springer, Cham, pp 507–517
 3. Krutikov AK, Meltsov VY, Podkovyirin VD (2020) Neural network of vector quantization as a tool for predicting sports achievements in individual and team sports. In: Proceedings of the first international Volga region conference on economics, humanities and sports, FICEHS 2019. AEBMR, vol 114. Atlantis Press, Kazan, pp 789–792
 4. Everingham M, Van Gool L, Williams C, Winn J, Zisserman A (2010) The pascal visual object classes (VOC) challenge. *Int J Comput Vision* 88(2):303–338
 5. Li-Na H, Jing-Chang N (2011) Researches on GRNN neural network in RF nonlinear systems modeling. In: International conference on IEEE: computational problem-solving (ICCP), pp 1–4
 6. Specht DF (1991) A general regression neural network. *IEEE Trans Neural Netw* 2(6):568–576
 7. Avedyan ED, Barkan GV, Levin IK (1999) Cascade neural networks. *Autom Telemech* 3:38–54
 8. Wang J, Gao N, Peng J, Mo J (2019) DCAR: deep collaborative autoencoder for recommendation with implicit feedback. In: Tetko I, Kůrková V, Karpov P, Theis F (eds) Artificial neural networks and machine learning, ICANN 2019, LNCS, vol 11730. Springer, Cham, pp 172–184
 9. Kohonen T (2008) Self-organizing maps, 3rd edn. Binom. Laboratory Knowledge, Moscow
 10. Fischer A, Do M, Stein T, Asfour T, Dillmann R, Schwameder H (2011) Recognition of individual kinematic patterns during walking and running—a comparison of artificial neural networks and support vector machines. *Int J Comput Sci Sport* 10(1):63–67
 11. Statistics. BoxRec. <https://boxrec.com/>. Last accessed 2021/01/25
 12. Pascanu R, Mikolov T, Bengio Y (2013) On the difficulty of training recurrent neural networks. *ICML* 28(3):1310–1318
 13. Krutikov AK, Meltsov VY, Lapitsky AA, Rostovtsev VS (2020) FPGA-implementation of a prediction module based on a generalized regression neural network. In: Proceedings of the 2020 IEEE conference of Russian young researchers in electrical and electronic engineering, EIcon Rus. IEEE Inc., S-Petersburg, pp 147–150
 14. Wang Y, Verba V, Bezumnov D, Tarasenko E (2019) Neural network using for prediction spinal diseases. In: 24th conference of FRUCT association. FRUCT, pp 790–796
 15. Gilfanov AF (2017) Neural network cascade filtering of multi-dimensional data. *Sci Without Borders* 6(11):74–78
 16. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein A, Berg A, Li F-F (2015) Imagenet large scale visual recognition challenge. *Int J Comput Vis* 115(3):211–252

Meta-Analysis of Research into the Issue of Brand Building on Social Media as a Subset of e-Business During the COVID-19 Pandemic



L'udovít Nastišin  and Richard Fedorko 

Abstract The paper presents a qualitative view of brand building with an emphasis on social media during the COVID-19 pandemic. The paper tackles the issues of what are the most frequently examined goals in terms of the given problem and under what conditions are these connections examined. The present study falls within the field of e-business. To achieve our goal, a meta-analytical procedure of qualitative research was applied. In doing so, 64 scientific studies from 2020 to 2021 were consulted in order to summarize the most interesting findings, draw implications for practice, and clearly point to the current direction of research in this area. The most frequently researched areas were identified, in particular the adaptation of the brand to new market conditions as well as the effect social media involvement has on brand loyalty or image. The greatest emphasis was placed on analyses carried out on Facebook and Instagram. The results indicate the onset of a change in the perception of these tools and the need to monitor the current state of knowledge. These findings can be considered significant not only in the e-business dimension.

Keywords Social media · Brand · e-business · Pandemic · Meta-analysis

1 Introduction

Due to the restrictions during the pandemic, entrepreneurs are forced to look for alternative ways to promote the brand as well as the products and services offered to remain relevant. Social media is a new communication platform that has grown in popularity over the last few years and has a high impact on consumers. With an increased number of followers, businesses can easily promote and support their ideas or business. Social media is a simple yet efficient tool facilitating interactions

L'. Nastišin · R. Fedorko (✉)

Faculty of Management and Business, University of Prešov, Konštantínova 16, 080 01 Prešov, Slovakia

e-mail: richard.fedorko@unipo.sk

L'. Nastišin

e-mail: ludovit.nastisin@unipo.sk

between parties involved allowing individuals to express themselves. Social media provides immense potential for e-business because consumers habitually log on to it daily and are exposed to companies. It also presents huge challenges for e-business, however, because it is an ever-changing space that is extremely noisy and crowded. Moreover, social media is a valuable source of market information allowing for brand involvement. Social media can enable interaction, collaboration, and networking by strengthening the relationships between parties involved.

As social media is used by brands to connect with customers, it is increasingly important to constantly monitor activities on social networks. Social media is undergoing a radical shift, and businesses nowadays use them in their business processes. The use of social media for business purposes may vary across the business world, but the ultimate goal is to create and share content on social networks. The use of social media is essential in business. Management in organizations should look for ways to get the most out of using social media in business processes. The use of social media is ubiquitous, and brands must carefully manage this channel in order to achieve their strategic goals. The ability to influence the public through social media is gradually becoming key to shaping consumer attitudes.

The latest data, according to [1, 2], show that at the beginning of 2021, there have been 4.20 billion social media users worldwide, representing more than 53% of the total global population. More than 9 out of 10 users use social media every month. The number of users has risen sharply in the last 12 months, with 490 million new users joining. A typical user has an account on more than 8 different platforms and spends an average of 2.5 h each day using social media. Japan had the lowest average number of social network accounts at 3.8 accounts, comparably the highest in India at 11.5 accounts per person [1]. Facebook remains the most widely used social media platform. Platform preferences vary from country to country. Social media platforms have almost tripled in the last decade. However, the new user registrations are slowing down. Social media now relies on the steady growth in the number of people with Internet access and smartphones, especially in developing regions. The current percentage of people using social media is 50.64% of the world's population. However, when looking at the penetration rate of platforms concerning the eligible audiences, 83.36% of the 4.57 billion global Internet users and 90.71% of the 4.20 billion mobile Internet users use social networks. Of the 7.82 billion people in the world, 50.64% of the population use social networks regardless of age or Internet access; of the 4.57 billion Internet users, 83.36% are active users; of the 4.17 billion mobile Internet users, 90.71% are active users; of 3.96 billion social media users, 99% have access to websites or applications via mobile devices. 40% of all Internet users worldwide use social media for work purposes [3]. In US, only 27% of people actively use social media at work [4], compared to 65% in Indonesia or 13% in Israel. The use of social networks in terms of location varies globally between men and women. Probably the most notable gap in gender differences was found in terms of how they are being used. Looking at the top 8 social media platforms by monthly active users, YouTube, LinkedIn, Twitter, and TikTok are used by men more frequently. Sites like Facebook and Instagram are more focused on women, especially Pinterest, which dominates among female audiences. When it comes to using social media on

devices, 99% of people in the world access networks exclusively on mobile devices (tablets or phones). About 78% access platforms exclusively from their mobile phone, compared to only 1.32% who access social networks only by computer [1, 2]. Current events in society and the state of knowledge point to the need for a synthesis of social media knowledge and their role in brand management in the context of an ongoing pandemic. This paper seeks to do this through a meta-analysis of existing research and synthesizing the resulting knowledge and information.

2 Materials and Methods

This part further describes the meta-analytical procedure used to identify relevant academic studies dealing with the phenomenon of brand building with the help of social media in the period of the pandemics caused by COVID-19. This meta-analysis aims to summarize the most interesting findings and draw implications for practice and clearly point out the current direction of research in this area. The meta-analysis refers to the process of integrating the results of many studies in order to arrive at a synthesis of evidence. The resulting overview combines the results of reviewed studies and provides a summary estimate. For this purpose, meta-analysis is considered to be a reliable tool for a qualitative summary of research results. This analytical approach does not require the formulation of hypotheses but focuses on a qualitative summary of the research instead. This analytical approach was also used in the studies of [5–10]. These studies have verified the legitimacy of the procedure and declared its significant added value. To meet our goal, we have divided the process into two main parts below as shown in Fig. 1.

The suitable scientific studies were taken from the Web of Science, Scopus, and Science Direct databases using relevant keywords. Thus, $N = 464$ scientific studies

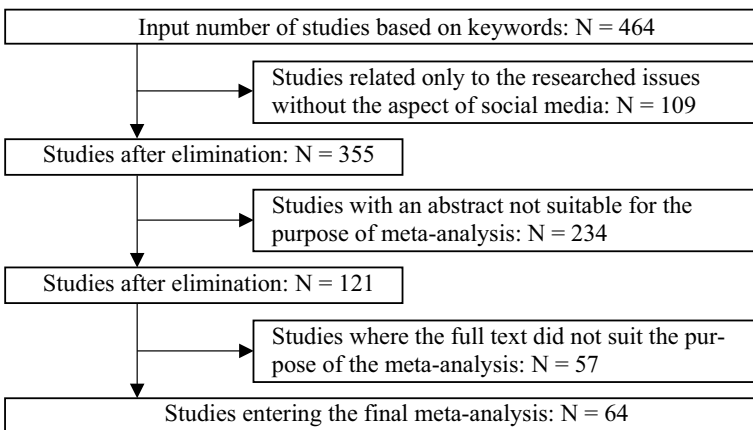


Fig. 1 The elimination process of meta-analysis

were identified. In the second step, $N = 109$ studies were eliminated, as they did not meet the particulars for being included in the meta-analysis. Another step in the elimination of unsuitable studies was the study of abstracts—we eliminated another $N = 234$ study as some abstracts did not discuss the issue we intended to examine. The last elimination step excluded further $N = 57$ studies. At this stage, the whole text of the paper was assessed. As a result, $N = 64$ scientific studies met the criteria for the purpose of the analysis. All of the studies studied were published in 2020 and 2021 and focused on the current situation— $N = 155$ studies were published in 2020 and $N = 309$ studies in 2021. The search chain chosen copied the purpose of this study, focusing on the keywords “social media,” “branding,” “business,” and “pandemic.”

3 Results and Discussion

We have created an overview table of meta-analytical output of selected works—listed below. Its full version is given in the Appendix. Table 1 outlines the areas of research covered by selected indexed studies, research subjects, research methods used, what results and findings the scientific studies bring, and which social networks the studies focused on.

3.1 *The Most Frequently Researched Areas*

Based on the meta-analysis, research areas such as market adaptation [12, 33, 61], ecosystems based on digital platforms [12, 28, 43, 56], the importance of the value of a retailer and consumer identity [14], flexibility of the sales process, attitudes, and key determinants of shoppers' behavior [22, 41, 60], growing use of social networks [1, 2] to measure performance, and consumer interaction using state-of-the-art technologies [33, 47, 53, 66, 73] came out as the most significant. Given the ongoing global pandemic, pandemic-related issues were also addressed [13, 24, 36, 47, 61, 73]. To a large extent, the studies have focused on crisis management or customer observation for interaction purposes. Areas affected by the pandemic include the ability of companies to respond to critical situations [71], measures in various sectors, whether hotel services, restaurants, hospitality, or tourism [13, 23]. The pandemic also affected consumer behavior in the first days of the pandemic with a high volume of social media activities during the pandemic, panic shopping on the Internet [47, 64], customer brand involvement, and subsequent impact of the brand on co-creation and re-intention in pandemics [54], the similarity of advertising during the pandemic [70]. On the positive side, this crisis has brought new solutions and ideas. According to research, the positives of the pandemic include increased operational efficiency and improved customer experience [23], the way businesses interact with customers, the growing use of social networks to measure performance,

Table 1 Overview of meta-analytical output

Research area	Research subject	Research method	Findings and results	Social media	Source
Continuity and responsiveness for a critical scenario	Fortune Glob. 500	1	Increased interest in decision analysis, the importance of emergency response	1	[11]
Growing conversation on ecosystems based on digital platforms		2	Current manufacturers are confronted with competition from digital subscriber platforms		[12]
The devastating impact of the pandemic on the hospitality industry	Hotel managers	3	Practical ideas aimed at creating a customer experience		[13]
The importance of matching the identity of the retailer and the consumer	Retail	3	Research develops a conceptual framework focused on the importance of creating a strong identity match between retailer and consumer		[14]
Social capacity in the EU			Social media as an aid to organizations in absorbing capacity development	1, 2, 3, 4, 5	[15]
The flexibility of the sales process	Leading sellers	4	The sales force should focus on increasing flexibility, improving the scale, and adaptability of technologies		[16]
Artificial intelligence in crisis detection	Senior managers	4	Critical events related to business activities		[17]
Technological revolutions			The aim of the site is to disseminate ideas, progress, and practices that are useful to the general public	1, 2, 3, 6	[18]
Analysis of social media using machine learning	Elsevier, IEEE, Springer, ACM	2	We can use social media to improve our relationships and thanks to popularity, we can easily promote our business	2, 3, 5, 6, 7	[19]

(continued)

Table 1 (continued)

Research area	Research subject	Research method	Findings and results	Social media	Source
The added value of products beyond traditional compromises			Digital logistics channels extend the functions of creating traditional logistics		[20]
A deeper understanding of innovative actors	Pandemic	2	Startups are able to respond to a changing environment the fastest		[21]
Attitudes and key determinants of shopper behavior	Consumers	2	Consumers' online shopping behavior depends significantly on their demographic characteristics		[22]
Artificial intelligence as a tool to increase operational efficiency	Hotel customers	5	Prioritizing human interaction with employees		[23]
Consumer behavior			Informing future research and marketing practice		[24]
Application domains in social networks	Software tool	1	Defining new metrics to evaluate different tools and social networking frameworks		[25]
Product networks and product convergence		1	The combination of the two products leads to the convergence of these products		[26]
The way sellers interact	Marketing and sales staff	3	Customer management of the interaction between the seller and the customer		[27]
Online media and pandemics	14,000 tweets		The high volume of online media activities during pandemic	6	[28]
Theoretical model of solving sensitivity in social networking sites	215 respondents		Heuristic behavior increases the susceptibility to phishing		[29]
Emergency and consumer behavior	15,000 tweets	2	Cultural differences between consumers during a pandemic	6	[30]

(continued)

Table 1 (continued)

Research area	Research subject	Research method	Findings and results	Social media	Source
Development and application of a social construction model	Exchange students	2	Insufficient standardization of items, which is observed in the visuals of the website		[31]
Coronavirus health crisis	1.6 mil. tweets	2	Pandemic coverage by traditional and social media	6	[32]
Transnational migration and information and communication technologies			Realizing business desires using digital platforms	2, 5	[33]
Advances in technology and popularity of social networking sites			Addiction caused by social media		[34]
The dark side of using social media	Gen Z	6	The harmful effect of pandemic information overload on social media		[35]
Development of corporate social responsibility and marketing			A pandemic for businesses means an opportunity to move toward social responsibility	5, 8, 9	[36]
Hospitality industry	Restaurant customers	1	The measures have negatively affected the restaurant industry and brand confidence		[37]
Interdependence of business and society			The pandemic as an alarm clock for industrial enterprises		[38]
Consumer emotions			The pandemic in consumers evokes much more negative feelings such as fear, shame, and guilt		[39]

(continued)

Table 1 (continued)

Research area	Research subject	Research method	Findings and results	Social media	Source
Communication of hotels with customers	Amazon Mechanical Turk	1, 2	A more comprehensive analysis of the interaction between the hotel brand and marketing communication		[40]
Social networks and increasing user involvement		1	Challenging economic approaches in business models		[41]
Social media skills	Employers		Improving relationships through social media		[42]
The pandemic and business activities	Consumers	1	The positive mitigating effect of pandemic fear on the relationship between the impact of perceived efficiency of e-commerce platforms, economic benefits, and sustainable consumption		[43]
Understanding trends and consumer shopping behavior	BB, Gen X, Gen Y	6	Fear is related to changes in customer behavior and affects traditional shopping		[44]
Consumer preferences in catering establishments	Consumers	2	Trust in the brand represents a feeling of security for the customer	3, 5	[45]
Shared mobility services	Service consumers	6	Consequences of trust in a shared economy		[46]
Panic shopping on the Internet	Consumers	3	Building global logistics through the use of social media		[47]
Solving pandemic-related problems		1	Lack of technological readiness		[48]
Consumer interaction with the latest technologies			The influence of cutting-edge technologies on marketing practice		[49]
Effects of the pandemic on the accommodation sector	Accommodation sector	2	Post pandemic recovery in the hospitality industry		[50]

(continued)

Table 1 (continued)

Research area	Research subject	Research method	Findings and results	Social media	Source
Development of the global economy			Pandemic as a source of concern and long-term effects on globalization	2	[51]
Design thinking in times of crisis			A pandemic as a benefit of good business methods for the future		[52]
Identification of different types of uncertainty			The importance of information uncertainty and the growing role of the impact of direct communication and social media		[53]
The effect of social media on customer brand involvement		2	Social media has a positive and significant impact on brand and customer involvement		[54]
Emerging markets			Economic growth of emerging markets		[55]
Use of social media when planning a trip	539 observations	6	Determinants predicting the actual use of social media		[56]
Aria project from Facebook		1	A project that inspires a change of focus to competition	2	[57]
Key features and functionality of LinkedIn			Social media as innovative pedagogical frameworks for teaching and learning	1	[58]
Ethical and private concerns regarding user data		3	Social media and involvement in international perspectives		[59]
Individual and collective effects of customers	Customers	5	Hospitality companies should make effective use of customer loyalty		[60]
Measuring consumer behavior	Customers	1	The effects of a pandemic and the type of product in shopping and the change in consumer behavior and priorities in the current crisis		[61]

(continued)

Table 1 (continued)

Research area	Research subject	Research method	Findings and results	Social media	Source
Customer satisfaction and loyalty	Customers	1	Dimension efficiency		[62]
Online clothing shopping		6	Verbal information as a mechanism influencing haptic display		[63]
Panic purchase		6	Excessive use of social media intensifies the relationship between reports of scarcity and perceived excitement		[64]
Areas in the hospitality and tourism sectors	Articles		Disaster crisis management dominated the main crisis management research		[65]
Consumer self-improvement			How users can best achieve their self-improvement goals with technology		[66]
The food industry's response to the pandemic crisis	Social posts	2	A new type of value offers that entrepreneurs experimented with	3	[67]
Machine learning based on image content	Public posts	2	Connecting tourist experiences via Instagram	3	[68]
Hospitality and tourism during the pandemic	Leaders in the hospitality industry	2	Future managers need to be aware of evolving practices related to employee involvement in multiple workplaces roles		[69]
The sameness of advertising in times of crisis			The similarity of advertising due to the pandemic in different areas	2	[70]
Customer behavior in co-creation			The influence of customer co-creation behavior, customer reaction, and customer experience value on customer satisfaction		[71]

(continued)

Table 1 (continued)

Research area	Research subject	Research method	Findings and results	Social media	Source
Motivation for shopping	Consumers	6	The size of the network predicted perceived pleasure, social interaction, and presence and usefulness		[72]
Key issues affecting organizations	Practice experts		The pandemic has forced many organizations to undergo significant transformation, rethink key elements of their business processes, and use technology to keep it running	2, 6	[73]
The revolution in the digital economy	Pharmaceutical org	2	It is possible to capture and analyze large amounts of user-generated content on social networks, which will help organizations gain an overview of market requirements and improve business	6	[74]

Note Research method: 1—quantitative analysis; 2—qualitative analysis, 3—interview; 4—research; 5—observation, 6—online survey/questionnaire. Social network: 1—LinkedIn; 2—Facebook; 3—Instagram; 4—Pinterest; 5—WhatsApp; 6—Twitter; 7—Snapchat; 8—Skype; 9—Zoom

and the revolution in the digital economy [12, 20, 33, 41, 42, 46, 58, 72]. Artificial intelligence has been employed to detect crises related to events in the company [23, 48]. With the growing use of social networks, phishing on online sites has begun to grow, thus giving rise to problems, ethical and private concerns about the use of social networks [29, 51].

3.2 Research Subjects and Research Methods

Subjects in each scientific study varied, with senior managers, hotel, and facility managers [11, 13, 17, 69], vendors, employers, experts in various fields, professional journals, and scientific articles [14, 42, 69] dominated. Another group included different generations of consumers, customers, students, and their tweets or social media posts [35, 44]. Due to a large number of research studies and their research methods, these were divided into six methods for easier understanding. The most

common research analysis employed was qualitative analysis ($N = 15$) and quantitative analysis ($N = 11$), followed by an interview ($N = 5$) and an online survey or questionnaire ($N = 6$). The least used research methods were research ($N = 2$) and observation ($N = 2$).

3.3 *Social Media*

Not all papers under research used social networks as a tool. Out of a total of 64 research indexed papers, only 34 studies used social networks as a data source for research. The most used social network in the studies under review was Facebook ($N = 8$) [15, 18, 19, 33, 51, 57, 70, 73]. Social networks like LinkedIn ($N = 4$) [11, 15, 18, 58], Instagram ($N = 6$) [15, 18, 19, 45, 67, 68], WhatsApp ($N = 5$) [15, 19, 33, 36, 45], and Twitter ($N = 5$) [18, 19, 28, 30, 32] have also been used frequently in various studies. The least attention in the research was paid to the social networks Pinterest ($N = 1$) [15], Snapchat ($N = 1$) [19], Skype ($N = 1$) [30], and Zoom ($N = 1$) [36].

3.4 *Discussion and Implications*

There have been countless results and findings, the most important of which is the importance of responding to crisis situations [11, 21, 24, 34, 61, 63, 71]. With the help of social media, relationships can be improved, and businesses promoted much easier. All of this can have a significant impact on e-business KPIs. The pandemic has had many negative effects, including high volumes of online media activity during the pandemic period, social media dependence [34, 66], the harmful effect of pandemic-related information overload on social media, and negative feelings, in particular fear, shame, and guilt [29, 39, 74]. The positive impact of the pandemic includes trust in the brand and a sense of security for the customer, good methods for business in the future, social media positively and significantly affect brand and customer involvement, and the similarity of advertising in different areas [37, 39, 43, 45, 54]. The pandemic has forced many organizations to undergo significant transformation, rethink key elements of their e-business processes, and use technology to maintain operations. It is possible to gather and analyze large amounts of user-generated content on social networks. This information helps organizations gain insight into market requirements and improve their business [26, 34, 74]. The abovementioned findings are considered as the most beneficial for the social networking and branding sector during the pandemic. When the world faced the lockdown, people limited themselves to the four walls of their homes, increasing the time they spent on social media platforms and behind the screens [1, 2, 4]. This yielded also some positives. Social media kept the world connected as the harsh reality of the new normal kept everyone physically apart. Brands and organizations were able to present themselves

during the pandemic as lockdowns provided them with an opportunity to connect with their audience like never before. When we look at 2021, we must understand that 2020 forever changed the way business is done [27, 28, 41, 44, 50, 57]. It is clear that there are no effective marketing strategies without social media [9]. However, since the beginning of the COVID-19 pandemic, we have seen an increase in the number of users. More users mean more space for advertising and organic communication. The proportion of users in the right age group makes it easier to choose the right communication channel [8, 22]. Small businesses have learned a lot since the first lockdowns, and many have made an important and successful transition to digital technology. According to a report on the state of small businesses on Facebook, the share of companies selling via digital channels has increased by 22% since the first survey among the same audience in April 2020 [75]. Despite this shift, 2021 is proving to be another difficult year for small business owners as the end is not in sight [75]. Brand identity and brand identification by customers are essential for the development of brand loyalty. Consumers perceive that products contain the essence of the brand [23, 46, 60, 62, 74]. These products are then considered to be more valuable and authentic. Consumers now know materialism, social class, and self-expression are reflected through both global and local products. Consumers also report higher purchasing intentions when employees are close to them [34, 37, 45, 46, 54, 63, 76].

4 Conclusion

The paper made use of a meta-analytical procedure of qualitative analysis in the field of brand building in the online space of social media as a subset of e-business during the pandemic. The paper aimed to summarize the most interesting findings and draw implications for practice, as well as to point out the future direction of research in the field. These objectives have been met, and many opportunities have opened up for future research. The biggest limitation of this analysis is the short time horizon as the pandemic is an ongoing issue (two years now), and many facts will only become apparent over a longer period of time. That is why there is a need to expand the knowledge base in this field.

Funding This paper is partial output under the scientific research grant VEGA 1/0694/20—Relational marketing research—perception of e-commerce aspects and its impact on purchasing behavior and consumer preferences and VEGA 1/0609/19—Research on the development of electronic and mobile commerce in the aspect of the impact of modern technologies and mobile communication platforms on consumer behavior and consumer preferences.

References

1. Datareportal (2021) Digital 2021 July global statshot report. <https://datareportal.com/reports/digital-2021-july-global-statshot>. Last accessed 2021/10/13
2. Dean B (2021) Backlinko. Social network usage and growth statistics: how many people use social media in 2021? <https://backlinko.com/social-media-users>. Last accessed 2021/10/15
3. Fedorko I, Bačík R, Gavurová B (2018) Technology acceptance model in e-commerce segment. *Managem Market Challenges Knowl Soc* 13(4):1242–1256. <https://doi.org/10.2478/mmcks-2018-0034>
4. Kang K, Lu J, Guo L, Zhao J (2020) How to improve customer engagement: a comparison of playing games on personal computers and on mobile phones. *J Theor Appl Electron Commer Res* 15(2). <https://doi.org/10.4067/s0718-18762020000200106>
5. Basu A (2017) How to conduct meta-analysis: a basic tutorial. *PeerJ Preprints*. <https://doi.org/10.7287/peerj.preprints.2978v1>
6. Crosno J, Dahlstrom R, Liu Y, Tong PY (2021) Effectiveness of contracts in marketing exchange relationships: a meta-analytic review. *Ind Mark Manage* 92:122–139. <https://doi.org/10.1016/j.indmarman.2020.11.007>
7. Folkvord F, van 't Riet J (2018) The persuasive effect of advergames promoting unhealthy foods among children: a meta-analysis. *Appetite* 129:245–251. <https://doi.org/10.1016/j.appet.2018.07.020>
8. Ismagilova E, Slade E, Rana NP, Dwivedi YK (2020) The effect of characteristics of source credibility on consumer behaviour: a meta-analysis. *J Retail Consum Serv* 53:101736. <https://doi.org/10.1016/j.jretconser.2019.01.005>
9. Leonidou LC, Katsikeas CS, Samiee S (2002) Marketing strategy determinants of export performance: a meta-analysis. *J Bus Res* 55(1):51–67. [https://doi.org/10.1016/s0148-2963\(00\)00133-8](https://doi.org/10.1016/s0148-2963(00)00133-8)
10. Sarkar S, Chauhan S, Khare A (2020) A meta-analysis of antecedents and consequences of trust in mobile commerce. *Int J Inf Manage* 50:286–301. <https://doi.org/10.1016/j.ijinfomgt.2019.08.008>
11. Margherita A, Heikkilä M (2021) Business Continuity in the COVID-19 emergency: a framework of actions undertaken by world-leading companies. *Business Horizons*. <https://doi.org/10.1016/j.bushor.2021.02.020>
12. Cozzolino A, Corbo L, Aversa P (2021) Digital platform-based ecosystems: the evolution of collaboration and competition between incumbent producers and entrant platforms. *J Bus Res* 126:385–400. <https://doi.org/10.1016/j.jbusres.2020.12.058>
13. Bonfanti A, Vigolo V, Yfantidou G (2021) The impact of the Covid-19 pandemic on customer experience design: the hotel managers' perspective. *Int J Hosp Manage* 94:102871. <https://doi.org/10.1016/j.ijhm.2021.102871>
14. Roggeveen AL, Grewal D, Karsberg J, Noble SM, Nordfält J, Patrick VM, Schweiger E, Soysal G, Dillard A, Cooper N, Olson R (2021) Forging meaningful consumer-brand relationships through creative merchandising offerings and innovative merchandising strategies. *J Retail* 97(1):81–98. <https://doi.org/10.1016/j.jretai.2020.11.006>
15. Arora AS, Sivakumar K, Pavlou PA (2021) Social capacitance: leveraging absorptive capacity in the age of social media. *J Bus Res* 124:342–356. <https://doi.org/10.1016/j.jbusres.2020.11.035>
16. Sharma A, Rangarajan D, Paesbrugge B (2020) Increasing resilience by creating an adaptive salesforce. *Ind Mark Manage* 88:238–246. <https://doi.org/10.1016/j.indmarman.2020.05.023>
17. Farrokhi A, Shirazi F, Hajli N, Tajvidi M (2020) Using artificial intelligence to detect crisis related to events: decision making in B2B by artificial intelligence. *Ind Mark Manage* 91:257–273. <https://doi.org/10.1016/j.indmarman.2020.09.015>
18. Raja Koti B, Raj Kumar G, Kumar KN, Srinivas Y (2021) Influence of social information networks and their propagation. *Secur IoT Soc Netw*:83–108. <https://doi.org/10.1016/b978-0-12-821599-9.00004-2>

19. Balaji TK, Annavarapu CSR, Bablani A (2021) Machine learning algorithms for social media analysis: a survey. *Comput Sci Rev* 40:100395. <https://doi.org/10.1016/j.cosrev.2021.100395>
20. Burroughs B, Burroughs WJ (2020) Digital logistics: enchantment in distribution channels. *Technol Soc* 62:101277. <https://doi.org/10.1016/j.techsoc.2020.101277>
21. Ebersberger B, Kuckertz A (2021) Hop to it! The impact of organization type on innovation response time to the COVID-19 crisis. *J Bus Res* 124:126–135. <https://doi.org/10.1016/j.jbusres.2020.11.051>
22. Melović B, Šehović D, Karadžić V, Dabić M, Ćirović D (2021) Determinants of millennials' behavior in online shopping—implications on consumers' satisfaction and e-business development. *Technol Soc* 65:101561. <https://doi.org/10.1016/j.techsoc.2021.101561>
23. Prentice C, Nguyen M (2020) Engaging and retaining customers with AI and employee service. *J Retail Consum Serv* 56:102186. <https://doi.org/10.1016/j.jretconser.2020.102186>
24. Kirk CP, Rifkin LS (2020) I'll trade you diamonds for toilet paper: consumer reacting, coping and adapting behaviors in the COVID-19 pandemic. *J Bus Res* 117:124–131. <https://doi.org/10.1016/j.jbusres.2020.05.028>
25. Camacho D, Panizo-Lledot N, Bello-Orgaz G, Gonzalez-Pardo A, Cambria E (2020) The four dimensions of social network analysis: an overview of research methods, applications, and software tools. *Inf Fusion* 63:88–120. <https://doi.org/10.1016/j.inffus.2020.05.009>
26. Stöckli DR, Khobzi H (2021) Recommendation systems and convergence of online reviews: the type of product network matters! *Decis Support Syst* 142:113475. <https://doi.org/10.1016/j.dss.2020.113475>
27. Rangarajan D, Sharma A, Lyngdoh T, Paesbrugge B (2021) Business-to-business selling in the post-COVID-19 era: developing an adaptive sales force. *Business Horizons*. <https://doi.org/10.1016/j.bushor.2021.02.030>
28. Mutlu EC, Oghaz T, Jasser J, Tutunculer E, Rajabi A, Tayebi A, Ozmen O, Garibay I (2020) A stance data set on polarized conversations on Twitter about the efficacy of hydroxychloroquine as a treatment for COVID-19. *Data Brief* 33:106401. <https://doi.org/10.1016/j.dib.2020.106401>
29. Frauenstein ED, Flowerday S (2020) Susceptibility to phishing on social network sites: a personality information processing model. *Comput Secur* 94:101862. <https://doi.org/10.1016/j.cose.2020.101862>
30. Pantano E, Priporas CV, Devereux L, Pizzi G (2021) Tweets to escape: intercultural differences in consumer expectations and risk behavior during the COVID-19 lockdown in three European countries. *J Bus Res* 130:59–69. <https://doi.org/10.1016/j.jbusres.2021.03.015>
31. Kanazawa FN, Lourenção M, de Oliveira JHC, de Moura Engracia Giraldo J (2021) Destination website management: a social constructionist approach. *J Destination Market Manag* 19:100545. <https://doi.org/10.1016/j.jdmm.2020.100545>
32. Rodrigues De Andrade FM, Barreto TB, Herrera-Feligueras A, Ugolini A, Lu YT (2021) Twitter in Brazil: discourses on China in times of coronavirus. *Soc Sci Human Open* 3(1):100118. <https://doi.org/10.1016/j.ssaho.2021.100118>
33. Steel G (2021) Going global—going digital. Diaspora networks and female online entrepreneurship in Khartoum Sudan. *Geoforum* 120:22–29. <https://doi.org/10.1016/j.geoforum.2021.01.003>
34. Wenninger H, Cheung CM, Chmielinski M (2021) Understanding envy and users' responses to envy in the context of social networking sites: a literature review. *Int J Inf Manag* 58:102303. <https://doi.org/10.1016/j.ijinfomgt.2020.102303>
35. Liu H, Liu W, Yoganathan V, Osburg VS (2021) COVID-19 information overload and generation Z's social media discontinuance intention during the pandemic lockdown. *Technol Forecast Soc Chang* 166:120600. <https://doi.org/10.1016/j.techfore.2021.120600>
36. He H, Harris L (2020) The impact of Covid-19 pandemic on corporate social responsibility and marketing philosophy. *J Bus Res* 116:176–182. <https://doi.org/10.1016/j.jbusres.2020.05.030>
37. Wei CV, Chen H, Lee YM (2021) Factors influencing customers' dine out intention during COVID-19 reopening period: the moderating role of country-of-origin effect. *Int J Hosp Manag* 95:102894. <https://doi.org/10.1016/j.ijhm.2021.102894>

38. Sheth J (2020) Business of business is more than business: managing during the Covid crisis. *Ind Mark Manag* 88:261–264. <https://doi.org/10.1016/j.indmarman.2020.05.028>
39. de Medeiros JF, Marcon A, Ribeiro JLD, Quist J, D'Agostin A (2021) Consumer emotions and collaborative consumption: the effect of COVID-19 on the adoption of use-oriented product-service systems. *Sustain Prod Consump* 27:1569–1588. <https://doi.org/10.1016/j.spc.2021.03.010>
40. Jiménez-Barreto J, Loureiro S, Braun E, Sthapit E, Zenker S (2021) Use numbers not words! Communicating hotels' cleaning programs for COVID-19 from the brand perspective. *Int J Hosp Manag* 94:102872. <https://doi.org/10.1016/j.ijhm.2021.102872>
41. Saura JR, Palacios-Marqués D, Iturricha-Fernández A (2021) Ethical design in social media: assessing the main performance measurements of user online behavior modification. *J Bus Res* 129:271–281. <https://doi.org/10.1016/j.jbusres.2021.03.001>
42. Sutherland K, Freberg K, Driver C, Khattab U (2020) Public relations and customer service: employer perspectives of social media proficiency. *Publ Relations Rev* 46(4):101954. <https://doi.org/10.1016/j.pubrev.2020.101954>
43. Tran LTT (2021) Managing the effectiveness of e-commerce platforms in a pandemic. *J Retail Consum Serv* 58:102287. <https://doi.org/10.1016/j.jretconser.2020.102287>
44. Eger L, Komárková L, Egerová D, Mičková M (2021) The effect of COVID-19 on consumer shopping behaviour: generational cohort perspective. *J Retail Consum Serv* 61:102542. <https://doi.org/10.1016/j.jretconser.2021.102542>
45. Hakim MP, Zanetta LD, da Cunha DT (2021) Should I stay, or should I go? Consumers' perceived risk and intention to visit restaurants during the COVID-19 pandemic in Brazil. *Food Res Int* 141:110152. <https://doi.org/10.1016/j.foodres.2021.110152>
46. Mas-Machuca M, Marimon F, Jaca C (2021) The unexplored potential of trust to boost customer loyalty for transport platforms. *Res Transp Bus Manag*:100618. <https://doi.org/10.1016/j.rtbm.2021.100618>
47. Naeem M (2021) Do social media platforms develop consumer panic buying during the fear of Covid-19 pandemic. *J Retail Consum Serv* 58:102226. <https://doi.org/10.1016/j.jretconser.2020.102226>
48. Donthu N, Gustafsson A (2020) Effects of COVID-19 on business and research. *J Bus Res* 117:284–289. <https://doi.org/10.1016/j.jbusres.2020.06.008>
49. Ameen N, Hosany S, Tarhini A (2021) Consumer interaction with cutting-edge technologies: implications for future research. *Comput Hum Behav* 120:106761. <https://doi.org/10.1016/j.chb.2021.106761>
50. Gerwe O (2021) The Covid-19 pandemic and the accommodation sharing sector: effects and prospects for recovery. *Technol Forecast Soc Chang* 167:120733. <https://doi.org/10.1016/j.techfore.2021.120733>
51. Ghauri P, Strange R, Cooke FL (2021) Research on international business: the new realities. *Int Bus Rev* 30(2):101794. <https://doi.org/10.1016/j.ibusrev.2021.101794>
52. Cankurtaran P, Beverland MB (2020) Using design thinking to respond to crises: B2B lessons from the 2020 COVID-19 pandemic. *Ind Mark Manag* 88:255–260. <https://doi.org/10.1016/j.indmarman.2020.05.030>
53. Sharma P, Leung T, Kingshott RP, Davcik NS, Cardinali S (2020) Managing uncertainty during a global pandemic: an international business perspective. *J Bus Res* 116:188–192. <https://doi.org/10.1016/j.jbusres.2020.05.026>
54. Rather RA (2021) Demystifying the effects of perceived risk and fear on customer engagement, co-creation and revisit intention during COVID-19: a protection motivation theory approach. *J Destin Mark Manag* 20:100564. <https://doi.org/10.1016/j.jdmm.2021.100564>
55. Cavusgil ST (2021) Advancing knowledge on emerging markets: past and future research in perspective. *Int Bus Rev* 30(2):101796. <https://doi.org/10.1016/j.ibusrev.2021.101796>
56. . Sakshi TU, Ertz M, Bansal H (2020) Social vacation: proposition of a model to understand tourists' usage of social media for travel planning. *Technol Soc* 63:101438. <https://doi.org/10.1016/j.techsoc.2020.101438>

57. Applin SA, Flick C (2021) Facebook's project aria indicates problems for responsible innovation when broadly deploying AR and other pervasive technology in the commons. *J Respons Technol* 5:100010. <https://doi.org/10.1016/j.jrt.2021.100010>
58. López-Carril S, Anagnostopoulos C, Parganas P (2020) Social media in sport management education: Introducing LinkedIn. *J Hosp Leis Sport Tour Educ* 27:100262. <https://doi.org/10.1016/j.jhlste.2020.100262>
59. Fraccastoro S, Gabriellson M, Pullins EB (2021) The integrated use of social media, digital, and traditional communication tools in the B2B sales process of international SMEs. *Int Bus Rev* 30(4):101776. <https://doi.org/10.1016/j.ibusrev.2020.101776>
60. Jang S, Chong K, Yoo C (2021) The effect of mobile application-driven customer participation on bakery purchase behavior: evidence from a field experiment. *Int J Hosp Manag* 94:102865. <https://doi.org/10.1016/j.ijhm.2021.102865>
61. Goswami S, Chouhan V (2021) Impact of change in consumer behaviour and need prioritisation on retail industry in Rajasthan during COVID-19 pandemic. *Mater Today Proc.* <https://doi.org/10.1016/j.matpr.2020.12.073>
62. Omar S, Mohsen K, Tsimonis G, Oozeerally A, Hsu JH (2021) M-commerce: the nexus between mobile shopping service quality and loyalty. *J Retail Consum Serv* 60:102468. <https://doi.org/10.1016/j.jretconser.2021.102468>
63. Silva SC, Rocha TV, de Cicco R, Galhanone RF, Manzini Ferreira Mattos LT (2021) Need for touch and haptic imagery: an investigation in online fashion shopping. *J Retail Consumer Serv* 59:102378. <https://doi.org/10.1016/j.jretconser.2020.102378>
64. Islam T, Pitafi AH, Arya V, Wang Y, Akhtar N, Mubarak S, Xiaobei L (2021) Panic buying in the COVID-19 pandemic: a multi-country examination. *J Retail Consum Serv* 59:102357. <https://doi.org/10.1016/j.jretconser.2020.102357>
65. Wut TM, Xu JB, Wong SM (2021) Crisis management research (1985–2020) in the hospitality and tourism industry: a review and research agenda. *Tour Manag* 85:104307. <https://doi.org/10.1016/j.tourman.2021.104307>
66. Wolf T, Jahn S, Hammerschmidt M, Weiger WH (2021) Competition versus cooperation: how technology-facilitated social interdependence initiates the self-improvement chain. *Int J Res Mark* 38(2):472–491. <https://doi.org/10.1016/j.ijresmar.2020.06.001>
67. Björklund TA, Mikkonen M, Mattila P, van der Marel F (2020) Expanding entrepreneurial solution spaces in times of crisis: business model experimentation amongst packaged food and beverage ventures. *J Bus Ventur Insights* 14:e00197. <https://doi.org/10.1016/j.jbvi.2020.e00197>
68. Arefieva V, Egger R, Yu J (2021) A machine learning approach to cluster destination image on Instagram. *Tour Manag* 85:104318. <https://doi.org/10.1016/j.tourman.2021.104318>
69. Kaushal V, Srivastava S (2021) Hospitality and tourism industry amid COVID-19 pandemic: perspectives on challenges and learnings from India. *Int J Hosp Manag* 92:102707. <https://doi.org/10.1016/j.ijhm.2020.102707>
70. Demsar V, Sands S, Campbell C, Pitt L (2021) “Unprecedented”, “extraordinary”, and “we're all in this together”: does advertising really need to be so tedious in challenging times? *Bus Horiz* 64(4):415–424. <https://doi.org/10.1016/j.bushor.2021.02.007>
71. Tueanrat Y, Papagiannidis S, Alamanos E (2021) A conceptual framework of the antecedents of customer journey satisfaction in omnichannel retailing. *J Retail Consum Serv* 61:102550. <https://doi.org/10.1016/j.jretconser.2021.102550>
72. Ma Y (2021) To shop or not: Understanding Chinese consumers' live-stream shopping intentions from the perspectives of uses and gratifications, perceived network size, perceptions of digital celebrities, and shopping orientations. *Telematics Inform* 59:101562. <https://doi.org/10.1016/j.tele.2021.101562>
73. Dwivedi YK, Hughes DL, Coombs C, Constantiou I, Duan Y, Edwards JS, Gupta B, Lal B, Misra S, Prashant P, Raman R, Rana NP, Sharma SK, Upadhyay N (2020) Impact of COVID-19 pandemic on information management research and practice: Transforming education, work and life. *Int J Inf Manag* 55:102211. <https://doi.org/10.1016/j.ijinfomgt.2020.102211>

74. Zhan Y, Han R, Tse M, Ali MH, Hu J (2021) A social media analytic framework for improving operations and service management: a study of the retail pharmacy industry. *Technol Forecast Soc Chang* 163:120504. <https://doi.org/10.1016/j.techfore.2020.120504>
75. Facebook—Meld je aan of registreer je (2021) Facebook. https://www.facebook.com/unsupportedbrowser?id=1077620002609475&+locale=cs_CZ
76. Štefko R, Bačík R, Fedorko I (2014) Facebook content analysis of banks operating on Slovak market. *Polish J Manag Stud* 10(1):145–152. <https://pjms.zim.pcz.pl/resources/html/article/details?id=187588&language=en>

Nonlinear Direct Adaptive Inverse Control Methodology Based on Volterra Model



Rodrigo Possidão Noronha

Abstract The main purpose of this work is to evaluate the performance of NLMS algorithm in the design of direct adaptive inverse control (DAIC). In order to evaluate this proposal in nonlinear dynamics, the controller is based on Volterra model. Since the Volterra model performs well in modeling nonlinear dynamic systems of polynomial type, the plant model, referring to a variable dissipation electric heater, is of the Nonlinear AutoRegressive with eXogenous inputs (NARX)-type subject to a disturbance signal of sinusoidal type. Motivated by the goal of DAIC of tracking the model inverse dynamics, thus it is pertinent to analyze the performance of this algorithm.

Keywords Control based on Volterra model · MSE · NLMS · Nonlinear adaptive inverse control · Speed convergence

1 Introduction

A complexity of great interest to scientific community of dynamical systems is non-linearity. Among the various types of nonlinear representations, polynomial models stand out. There are several types of nonlinear mathematical representations of polynomial type, such as Hammerstein model [1], Wiener model [2], Nonlinear AutoRegressive with eXogenous inputs (NARX) model [3], among others.

In [4], the direct adaptive inverse control (DAIC) control technique was proposed. The goal of DAIC is to obtain a controller whose dynamics approximate of the plant inverse dynamics. To satisfy this goal, the controller mathematical representation should track the plant inverse dynamics. The tracking the inverse plant dynamics is possible through plant inverse model identification. Some contributions to DAIC can be seen in [4–7]. Although these contributions are important to DAIC theory, they are focused only on the case where the plant model is linear, and thus, the mathematical

R. P. Noronha (✉)

Department of Electrical Engineering, Federal Institute of Maranhão, Imperatriz, MA 65906-335, Brazil

e-mail: rodrigo.noronha@ifma.edu.br

representation of the controller is linear. It is important to note that in the presence of uncertainties, such as an unmodeled dynamics of nonlinear type, the performance of the controller will be deteriorated [8].

Besides the nonlinear models of polynomial type mentioned above, another nonlinear representation of polynomial type that has received a great attention is the Volterra model. The Volterra model has been used in control systems [9], prediction systems [10] and fault diagnostics systems [11]. By definition, a Volterra model is represented through a general form of the impulse response, such that its mathematical representation consists of linear terms and nonlinear terms of polynomial type [12]. Due to the characteristics mentioned in this paragraph, in this work, the DAIC is based on Volterra model. On the other hand, an additional complexity to the Volterra model is the determination of the number of Volterra kernels. The determination of the number of Volterra kernels is fundamental for a satisfactory performance of Volterra model [12, 13]. This problem is usually addressed in the literature truncating the Volterra model until the second kernel [14–17].

It is important to note that the performance of the estimation algorithm in the update of the controller weight vector is of fundamental importance for tracking the plant inverse dynamics. Since the weights of Volterra model are linear with respect to the plant output signal, then, for example, it is possible to estimate them using an algorithm based on stochastic gradient. This work has as main proposal to evaluate the performance of the normalized least mean square (NLMS) algorithm in the DAIC design. The criterion for performance evaluation is referring to convergence speed and steady-state mean square error (MSE). The controller was applied on a plant with nonlinearity of polynomial type, referring to a variable dissipation electric heater, whose plant model is of the NARX-type subject to a disturbance signal of sinusoidal type. This work is organized as follows: In Sect. 2, the mathematical representation of Volterra model is shown; in Sect. 3, the proposed DAIC methodology is presented; in Sect. 4, computational results obtained are shown.

2 Volterra Model

Through a Volterra model, where $q(k)$ is the input signal and $\bar{v}(k)$ is the output signal, a nonlinear dynamic system is modeled by the following general form [14]:

$$\bar{v}(k) = \sum_{s=1}^{+\infty} \sum_{s_1=-\infty}^{+\infty} \dots \sum_{s_g=-\infty}^{+\infty} a_s(s_1, s_2, \dots, s_g) \prod_{i=1}^g q(k - s_i), \quad (1)$$

$a_s(s_1, s_2, \dots, s_g)$ is the s -th kernel. For simplicity in the design and analysis of DAIC, the Volterra model (1) is developed and truncated as follows:

$$\bar{v}(k) = \sum_{s_1=0}^{H_1} a_1(s_1) q(k - s_1) + \sum_{s_1=0}^{H_2} \sum_{s_2=0}^{H_2} a_2(s_1, s_2) q(k - s_1) q(k - s_2), \quad (2)$$

$a_1(s_1)$ is the first kernel and $a_2(s_1, s_2)$ is the second kernel. In (2), it is possible to note that the Volterra model is composed of the first-order convolution summation (corresponds to the linear terms) and the second-order convolution summation (corresponds to the nonlinear terms of polynomial type) [18]. Since $H_1 = H_2 = 2$, the nonlinear terms of polynomial type have maximum degree equal to 2.

2.1 Estimation of the Weight Vector of Volterra Model Via NLMS Algorithm

An important feature present in the Volterra model is that its kernels possess linear weights [19]. Through NLMS algorithm, that belong to class of algorithms based on stochastic gradient [20], the weight vector of Volterra model is estimated through the minimization of the following cost functional:

$$J = \nabla_{\mathbf{A}(k)}(l^2(k)), \quad (3)$$

such that the weight vector $\mathbf{A}(k)$ is updated as follows [19]:

$$\mathbf{A}(k + 1) = \mathbf{A}(k) - \frac{1}{2} \vartheta \nabla_{\mathbf{A}(k)}(l^2(k)), \quad (4)$$

where ϑ is the step size, $l(k) = v(k) - \bar{v}(k)$ is the error obtained between the dynamic system output signal $v(k)$, such that $\bar{v}(k) = (\mathbf{A}(k))^T \mathbf{Q}(k) = (\mathbf{Q}(k))^T \mathbf{A}(k)$, whose weight vector $\mathbf{A}(k) \in \mathbb{R}^{12 \times 1}$ and the input signal vector $\mathbf{Q}(k) \in \mathbb{R}^{12 \times 1}$ are defined as:

$$\begin{aligned} \mathbf{Q}(k) &= [q(k) \dots q(k - 2) \ q(k)^2 \ q(k)q(k - 1) \dots q(k - 2)^2] \\ \mathbf{A}(k) &= [a_1(0) \dots a_1(2) \ a_2(0, 0) \ a_2(0, 1) \dots a_2(2, 2)]. \end{aligned} \quad (5)$$

Since $\nabla_{\mathbf{A}(k)}(l^2(k)) = -2l(k)\mathbf{Q}(k)$, substituting it in (4), it is obtained that [19]:

$$\mathbf{A}(k + 1) = \mathbf{A}(k) - \frac{1}{2} \vartheta \nabla_{\mathbf{A}(k)}(l^2(k)) = \mathbf{A}(k) + \vartheta l(k)\mathbf{Q}(k), \quad (6)$$

dividing (6) by $(\mathbf{Q}(k))^T \mathbf{Q}(k)$, it is obtained the NLMS algorithm:

$$\mathbf{A}(k + 1) = \begin{cases} \mathbf{A}(k) + \vartheta \frac{l(k)\mathbf{Q}(k)}{(\mathbf{Q}(k))^T \mathbf{Q}(k)}, & \text{if } (\mathbf{Q}(k))^T \mathbf{Q}(k) \neq 0 \\ \mathbf{A}(k), & \text{if } (\mathbf{Q}(k))^T \mathbf{Q}(k) = 0, \end{cases} \quad (7)$$

3 DAIC

In this work, the output signal $v(k)$ of a plant is defined as follows:

$$v(k) = T[v(k - 1), \dots, v(k - s_v), q(k), q(k - 1), \dots, q(k - s_q)], \quad (8)$$

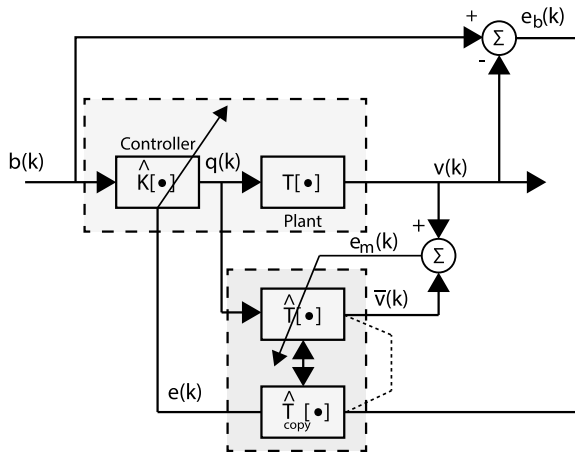
such that $q(k)$ is the control signal and $T[\cdot]$ is a functional of polynomial type that represents the stable nonlinear dynamics. The plant model $T[\cdot]$ is described through a NARX model where $s_q = 1$ and $s_v = 2$, given by:

$$\begin{aligned} v(k) = & d_{0,0} + \sum_{s_1=1}^2 d_{1,0}(s_1)v(k - s_1) + \sum_{s_1=1}^2 d_{1,0}(s_1)q(k - s_1) \\ & + \sum_{s_1=1}^2 \sum_{s_2=1}^2 d_{2,0}(s_1, s_2)v(k - s_1)v(k - s_2) \\ & + \sum_{s_1=1}^2 \sum_{s_2=1}^2 d_{1,1}(s_1, s_2)v(k - s_1)q(k - s_2) \\ & + \sum_{s_1=1}^2 \sum_{s_2=1}^2 d_{0,2}(s_1, s_2)q(k - s_1)q(k - s_2), \end{aligned} \quad (9)$$

such that $d_{0,0}, d_{1,0}(s_1), d_{2,0}(s_1, s_2), d_{1,1}(s_1, s_2)$ e $d_{0,2}(s_1, s_2)$ are the kernels.

The controller is described by the functional of polynomial type $\hat{K}[\cdot]$, as can be noted in Fig. 1. The controller is based on Volterra model, such that its goal is to track the inverse dynamics, where the controller weight vector $\mathbf{A}(k) \in \mathbb{R}^{12 \times 1}$ and the signal output are given by:

Fig. 1 DAIC



$$\begin{aligned}
 q(k) &= \sum_{s_1=0}^2 a_1(s_1) b(k-s_1) \\
 &+ \sum_{s_1=0}^2 \sum_{s_2=0}^2 a_2(s_1, s_2) b(k-s_1) b(k-s_2),
 \end{aligned}
 \tag{10}$$

$\mathbf{A}(k) = [a_1(0) \ a_1(1) \ a_1(2) \ a_1(2) \ a_2(0, 0) \ a_2(0, 1) \ a_2(2, 1) \ a_2(2, 2)]^T$. Developing (10), the following difference equation is obtained:

$$\begin{aligned}
 q(k) &= a_1(0)b(k) + a_1(1)b(k-1) + a_1(2)b(k-2) + a_2(0, 0)b(k)^2 \\
 &+ a_2(0, 1)b(k)b(k-1) + \dots + a_2(2, 1)b(k-2)b(k-1) \\
 &+ a_2(2, 2)b(k-2)^2,
 \end{aligned}
 \tag{11}$$

$b(k)$ is the reference signal. Rewriting (11), it is obtained that:

$$q(k) = (\mathbf{B}(k))^T \mathbf{A}(k) = (\mathbf{A}(k))^T \mathbf{B}(k),
 \tag{12}$$

$\mathbf{B}(k) = [b(k) \ \dots \ b(k-2) \ b(k)^2 \ b(k)b(k-1) \ \dots \ b(k-2)b(k-1) \ b(k-2)^2]^T \in \mathbb{R}^{12 \times 1}$.

In this work, the update of $\mathbf{A}(k)$ through NLMS algorithm is performed only after the error $e(k)$ being obtained [4]. The error $e(k)$ is obtained after the update of the weight vector $\mathbf{F}(k) \in \mathbb{R}^{12 \times 1}$ of the estimated plant model described by the functional of polynomial type $\hat{T}[\cdot]$, whose signal output is given by:

$$\bar{v}(k) = \sum_{s_1=0}^2 f_1(s_1) q(k-s_1) + \sum_{s_1=0}^2 \sum_{s_2=0}^2 f_2(s_1, s_2) q(k-s_1) q(k-s_2),
 \tag{13}$$

$\mathbf{F}(k) = [f_1(0) \ f_1(1) \ f_1(2) \ f_2(0, 0) \ f_2(0, 1) \ \dots \ f_2(2, 1) \ f_2(2, 2)]^T$. Developing (13) it is obtained that:

$$\begin{aligned}
 \bar{v}(k) &= f_1(0)q(k) + f_1(1)q(k-1) + f_1(2)q(k-2) + f_2(0, 0)q(k)^2 \\
 &+ f_2(0, 1)q(k)q(k-1) + \dots + f_2(2, 1)q(k-2)q(k-1) \\
 &+ f_2(2, 2)q(k-2)^2,
 \end{aligned}
 \tag{14}$$

rewriting (14), it is obtained that:

$$\bar{v}(k) = (\mathbf{Q}(k))^T \mathbf{F}(k) = (\mathbf{F}(k))^T \mathbf{Q}(k),
 \tag{15}$$

$\mathbf{Q}(k) = [q(k) \ q(k-1) \ q(k-2) \ q(k)^2 \ q(k)q(k-1) \ \dots \ q(k-2)q(k-1) \ q(k-2)^2]^T \in \mathbb{R}^{12 \times 1}$.

In this work, to perform the update of $\mathbf{F}(k)$, previously, it is obtained the residual error $e_m(k)$ [4]. According to Fig. 1, since $e_m(k) = v(k) - \bar{v}(k)$, it is obtained that:

$$\begin{aligned}
 e_m(k) &= d_{0,0} + \sum_{s_1=1}^2 d_{1,0}(s_1)v(k - s_1) + \sum_{s_1=1}^2 d_{1,0}(s_1)q(k - s_1) \\
 &+ \sum_{s_1=1}^2 \sum_{s_2=1}^2 d_{2,0}(s_1, s_2)v(k - s_1)v(k - s_2) \\
 &+ \sum_{s_1=1}^2 \sum_{s_2=1}^2 d_{1,1}(s_1, s_2)v(k - s_1)q(k - s_2) \\
 &+ \sum_{s_1=1}^2 \sum_{s_2=1}^2 d_{0,2}(s_1, s_2)q(k - s_1)q(k - s_2) \\
 &- \sum_{s_1=0}^2 a_1(s_1)q(k - s_1) \\
 &- \sum_{s_1=0}^2 \sum_{s_2=0}^2 a_2(s_1, s_2)q(k - s_1)q(k - s_2).
 \end{aligned} \tag{16}$$

With the update of $\mathbf{F}(k)$ obtained through NLMS algorithm, the error $e(k)$ is obtained as:

$$e(k) = \sum_{s_1=0}^2 f_1(s_1)e_b(k - s_1) + \sum_{s_1=0}^2 \sum_{s_2=0}^2 f_2(s_1, s_2)e_b(k - s_1)e_b(k - s_2), \tag{17}$$

developing (17), it is obtained that:

$$\begin{aligned}
 e(k) &= f_1(0)e_b(k) + f_1(1)e_b(k - 1) \\
 &+ f_1(2)e_b(k - 2) + f_2(0, 0)e_b(k)^2 \\
 &+ f_2(0, 1)e_b(k)e_b(k - 1) + \dots + f_2(2, 1)e_b(k - 2)e_b(k - 1) \\
 &+ f_2(2, 2)e_b(k - 2)^2,
 \end{aligned} \tag{18}$$

rewriting (18), it is obtained that:

$$e(k) = (\mathbf{E}_b(k))^T \mathbf{F}(k) = (\mathbf{F}(k))^T \mathbf{E}_b(k), \tag{19}$$

$\mathbf{E}_b(k) = [e_b(k) \ e_b(k - 1) \ e_b(k - 2) \ (e_b(k))^2 \ e_b(k)e_b(k - 1) \ \dots \ e_b(k - 2)e_b(k - 1) \ (e_b(k - 2))^2]^T \in \mathbb{R}^{12 \times 1}$ and the reference error $e_b(k) = b(k) - v(k)$.

4 Computational Results

Through the computational results that will be presented, the objective is to analyze the performance of NLMS algorithm, on a plant referring to a variable dissipation electric heater, with respect to the DAIC design. The NARX model is described below [21]:

$$\begin{aligned}
 v(k) = & 0.0013 - 0.5698v(k-1) \\
 & - 0.1253v(k-2) - 0.2092q(k-1) \\
 & + 0.2063q(k-2) - 1.407q(k-1)v(k-1) \\
 & + 0.8841q(k-2)v(k-1) + 1.1571q(k-1)v(k-2) \\
 & - 1.3491q(k-2)v(k-2) + 0.6784q(k-1)^2 \\
 & + 0.1119q(k-1)q(k-2) + 0.4146q(k-2)^2.
 \end{aligned} \tag{20}$$

In this simulation, $T_a = 0.025$ s was the sampling period used. In addition, 81,000 samples were used to perform the performance analysis of NLMS algorithm. For a satisfactory plant model, inverse identification, $\mathbf{A}(k)$ and $\mathbf{F}(k)$ were set equal to:

$$\begin{aligned}
 \mathbf{A}(k) &= [a_1(0) \ a_1(1) \ a_1(2) \ a_2(0,0) \ a_2(0,1) \ a_2(0,2) \ a_2(1,1) \ a_2(1,2) \ a_2(2,2)]^T \\
 \mathbf{F}(k) &= [f_1(0) \ f_1(1) \ f_1(2) \ f_2(0,0) \ f_2(0,1) \ f_2(0,2) \ f_2(1,1) \ f_2(1,2) \ f_2(2,2)]^T.
 \end{aligned} \tag{21}$$

The results that were obtained using the NLMS in the DAIC design based on Volterra model were compared with the results obtained using the LMS. For a fair comparison, with respect to the two algorithms, $\vartheta = 0.001$. It is important to note that the analysis, for standardization of results, was performed with respect to update of $\mathbf{A}(k)$. Thus, since the update of the controller weight vector was performed as a function of the plant model inverse identification, the convergence speed of $\mathbf{A}(k)$ was analyzed as a function of the convergence speed of $v(k)$ to $b(k)$. With respect to the steady-state MSE, the performance analysis was performed by means of MSE of $e(k)$.

In Fig. 2, the disturbance signal of sinusoidal type $n(k)$ is shown. The signal $v(k)$, which is the plant output, obtained through NLMS and LMS algorithm is shown in Fig. 3. It is possible to note that $v(k)$ obtained by means of NLMS developed a satisfactory convergence to $b(k)$, considering $n(k)$ added to $q(k)$, with respect to convergence speed. The obtained results demonstrate the satisfactory performance of NLMS algorithm in the update of $\mathbf{A}(k)$ through the plant model inverse identification.

In addition, the superior performance obtained by the NLMS algorithm with respect to convergence speed is maintained also for the steady-state MSE, as can be noted in Fig. 4. Through Fig. 5, it is possible to note the control signal developed by $\hat{\mathbf{K}}$. Thus, it is possible to affirm that the NLMS algorithm develops a better performance in the DAIC design based on Volterra model.

Fig. 2 Signal $n(k)$

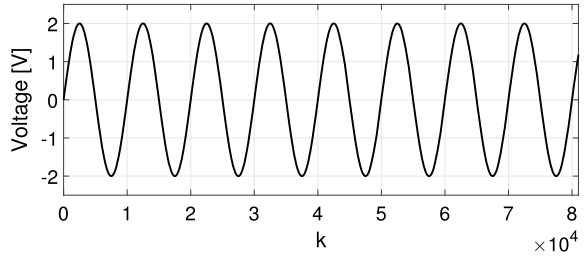


Fig. 3 Signal $\bar{v}(k)$

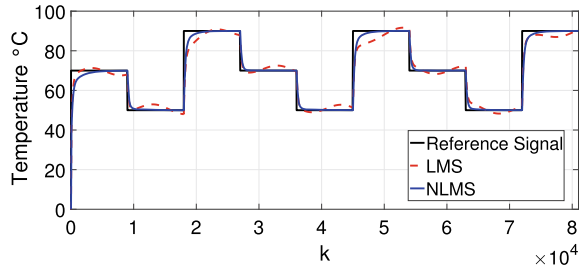


Fig. 4 MSE

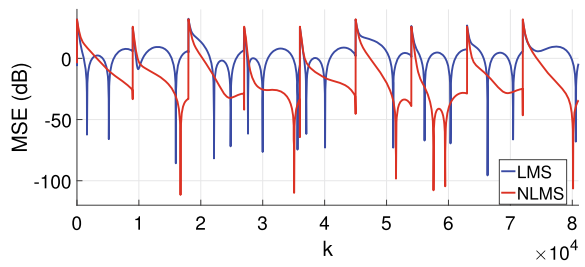
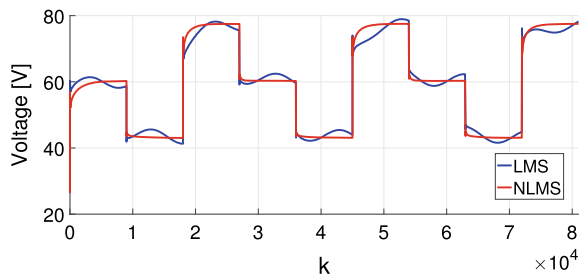


Fig. 5 Signal $q(k)$



5 Conclusion

The purpose of this work was to perform the performance analysis of NLMS algorithm with respect to the DAIC design based on Volterra model applied to a variable dissipation electric heater. Through the comparison of results obtained, considering the disturbance signal of sinusoidal type, it was possible to note the superior performance of NLMS algorithm. In addition, through the proposed methodology, it is possible to conclude that the controller performs satisfactorily in the control of nonlinear dynamics of polynomial type.

References

1. Moghaddam MJ, Mojallali H, Teshnehlab M (2018) Recursive identification of multiple-input single-output fractional-order Hammerstein model with time delay. *Appl Soft Comput* 70:486–500
2. Ranginga D, TK R, Velswamy K, (2021) Explicit nonlinear predictive control algorithms for Laguerre filter and sparse least square support vector machine-based Wiener model. *Trans Inst Measur Control* 43(4):812–883
3. Ma Y, Liu H, Zhu Y, Wang F, Luo Z (2017) The NARX model-based system identification on nonlinear, rotor-bearing systems. *Appl Sci* 7(9):911
4. Shafiq M, Shafiq AM, Yousef AH (2017) Stability and convergence. *Complexity* 2017
5. Shafiq M, Al Lawati M, Yousef H (2016) A simple direct adaptive inverse control structure. In: *Canadian conference on electrical and computer engineering (CCECE)*. IEEE, pp 1–4
6. Shafiq MA, Shafiq M, Ahmed N (2014) Closed loop direct adaptive inverse control for linear plants. *Sci World J* 2014
7. Ahmed OHA (2012) High performance speed control of direct current motors using adaptive inverse control. *WSEAS Trans Syst Control* 2:54–63
8. Ribeiro HA, Tiels K, Umenberger J, Schon BT, Aguirre AL (2020) On the smoothness of nonlinear system identification. *Automatica* 121:109158
9. Lin Y, Din Q, Rafagat M, Elsadany AA, Zeng Y (2020) Dynamics and chaos control for a discrete-time lotka-volterra model. *IEEE Access* 8:126760–126775
10. Son HJ, Kim Y (2021) Probabilistic time series prediction of ship structural response using volterra series. *Marine Struct* 76:102928
11. Wang K, Guan Y, Li D, Li X, Guan J (2020) Research on fault diagnosis of analog circuit based on volterra theory and higher-order spectrum analysis. *IOP Conf Ser Mater Sci Eng* 782(3)
12. Doyler JF, Pearson KR, Ogunnaiké AB (2002) *Identification and control using Volterra models*. Springer
13. Schoukens J, Ljung L (2019) Nonlinear system identification: a user-oriented road map. *IEEE Control Syst Mag* 39(6):28–99
14. Mortensen R (1983) *Nonlinear system theory: the volterra/wiener approach*. JSTOR
15. Shiki BS, Lopes V, da Silva S (2014) Identification of nonlinear structures using discrete-time volterra series. *J Br Soc Mech Sci Eng* 36(3):523–532
16. Da Silva S (2011) Non-linear model updating of a three-dimensional portal frame based on wiener series. *Int J Non-Lin Mech* 46(1):312–320
17. da Silva S, Cogan S, Foltête E (2010) Nonlinear identification in structural dynamics based on wiener series and kautz filters. *Mech Syst Signal Process* 24(1):52–58
18. Chatterjee A (2010) Identification and parameter estimation of a bilinear oscillator using volterra series with harmonic probing. *Int J Non-Lin Mech* 45(1):12–20

19. Paulo S (2008) Adaptive filtering, algorithms and practical implementations. Springer Publisher, New York
20. Tarrab M, Feuer A (1988) Convergence and performance analysis of the normalized LMS algorithm with uncorrelated Gaussian data. *IEEE Trans Inf Theor* 34(4):680–691
21. Verly A (2012) Caracterização de agrupamentos de termos na seleção de estrutura de modelos polinomiais narx. Universidade Federal de Minas Gerais

Alerting the Impact of Adversarial Attacks and How to Detect it Effectively via Machine Learning Approach: With Financial and ESG Data



Ook Lee, Hyodong Ha, Hayoung Choi, Hanseon Joo, and Minjong Cheon

Abstract ESG is short for “environmental, social, and governance” and it has been widely used as an indicator for investment. Investment firms asserted that they would incorporate the ESG indicator into their portfolio. Therefore, various AI approaches were applied to analyze the relationship between ESG and the benefit of the company. However, the adversarial attacks on AI models were prominent these days, and they could badly affect financial performance. This research aims to alert the danger of the attack and how to detect the anomaly data. The experiment involves two stages and focuses on classification performance and detecting noise data, respectively. In the first stage, it revealed that the accuracy of the classification on the noise dataset could drop almost 15% compared to the ordinary dataset. In the second stage, local outlier factors and isolation forest algorithms were applied to detect the noise data and they revealed 95.156% and 84.1% for detecting, respectively. These experiments yield that even tiny values of noise could influence the result significantly, and suggest a way to detect them. The limitation of this research is that it only conducts uncomplicated binary classification, and could not propose a way to defend the attack or filter the noise. Further research should be conducted to apply these algorithms to more sophisticated classification or regression. However, it is still worthwhile that this research could alert the risk of the adversarial attack on AI models, and suggest a probability of applying the proposed model to other fields of research.

Keywords ESG · Gaussian noise · Light gradient boosting machine · Local outlier factors · Machine learning

O. Lee · H. Ha · H. Choi · H. Joo · M. Cheon (✉)

Department of Information Systems, Hanyang University, 222 Wangshimni-ro, Seongdong-gu, Seoul 04673, South Korea

e-mail: jmj2316@hanyang.ac.kr

1 Introduction

1.1 Background

ESG is short for “environmental, social and governance” and it has been used as an indicator for the investor to evaluate the corporate behavior of the companies and determine whether to invest or not. The aim of the ESG term is to prevent various risks which could arise from each value. The representative examples of the environmental risks are climate impact, pollution, and environmental resources. Social risk includes social activities of the company, such as protecting human rights and focusing on product integrity. Lastly, governance risks refer to how companies run, including corporate risk management, protecting stakeholders, and excessive executive compensation [1].

Furthermore, not only preventing those risks, some people suggest that strong ESG performance could bring benefit to companies and investors. For example, some investment consulting companies claimed that they would measure the companies based on the ESG criteria and a plentiful amount of investment firms also integrated the ESG indicator into their portfolio [2].

In recent years, adversarial attacks in machine learning have been a growing threat to AI and machine learning researchers. The most prevalent type of attack is the evasion attack, which modulates the machine learning classification model in a way that is difficult to identify with the human eye, and eventually causes errors [3]. As various artificial intelligence algorithms including machine learning and deep learning algorithms were applied in various fields, these attacks could trigger financial fraud, drone strikes, and errors in autonomous driving [4]. Therefore, AI researchers should always keep in mind the danger of these attacks. The graph below exhibits that ESG is quite an important factor when choosing investments, based on gender, age, political background, and ethnicity. Figure 1 shows how important is ESG ranking when people choose investments, and most people answer that ESG is significant.

1.2 Objective

As the machine learning adversarial attacks have increased these days, the objective of this paper is to inform how the attack could influence the performance of the model with the financial dataset, and how to detect them. The experiments involve two stages. In the first stage, various machine learning algorithms were applied to classify the given dataset, whether those investments were good or bad. Then, Gaussian noise was added to the dataset, with the aim of figuring out how the noise could affect the performance of the same machine learning algorithms. In the second stage, anomaly detection algorithms are applied to detect the noisy data. The dataset is modified to contain noisy or normal values for the detection and it contains only 1%

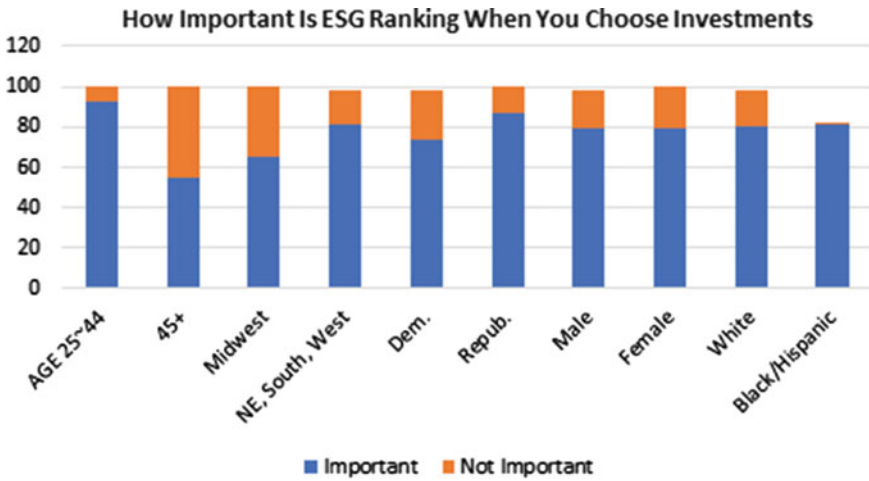


Fig. 1 American investors' responses in a survey [5]

of the noise data. Through this research, this paper could alert people how dangerous adversarial attack is and how to detect them efficiently.

2 Related Works

The aim of this paper is to solve the noise problem in credit risk classification. The research contains three different stages. The first stage includes introducing four indicators for the evaluation of the noise. Then, in the second stage, different noise levels are split based on the dual-voting results of the noise level. In the last stage, four learning strategies were applied to the dataset with different strategies. Then, principal component analysis (PCA), linear discriminant analysis (LDA), multidimensional scaling (MDS), kernel PCA (KPCA), and restricted Boltzmann machine (RBM) were applied, respectively. The dataset contains the Japanese credit, which is from the UCI Machine Learning Repository, and German credit, which is also from the same website. The result of the experiment yields that dual-voting-based learning paradigm one performed better than the other strategies [6].

Margot et al., conducted research to figure out the relationship between ESG profiles and financial performances for companies in a large investment universe. The dataset was obtained from the capitalization-weighted MSCI World Index USD and the portfolios were calculated in USD. Stock prices were derived from Thomson Reuters/Datastream. A machine learning algorithm was applied in order to find out the pattern between ESG ratings and financial performances, and the result exhibited that the Positive ML Screening algorithm, which was suggested by Margot et al.,

outperformed the other portfolios. Through this research, a novel approach to ESG behavior and the economy was figured out [7].

Lucia et al., conducted research to figure out whether ESG company's practices will bring better financial performance. The dataset involved 1038 public companies in Europe, which was obtained from Thomson Reuters Eikon. Various machine learning algorithms, such as random forest, support vector regression, ridge regression, and inferential model, were applied for the research, and the result revealed an accurate prediction of ROE and ROA values, which can be concluded that there exists a relationship between ESG practices and the financial indicators [8].

3 Prior Work

The prior research was conducted to detect fraudulent transactions from the credit card, based on deep learning methods. As the dataset contains 9.83% of normal data and 0.17% of fraud data, anomaly detection algorithms should be applied. Therefore, the proposed model consists of Bi_LSTM_autoencoder and isolation forest algorithm. The hybrid autoencoder acted as a filter for the unbalanced dataset, and the isolation forest algorithm finally found the outliers. The proposed algorithms achieved 87% detection accuracy and this score was higher than the other anomaly detection algorithms. Furthermore, when combined with bidirectional LSTM-based autoencoders, it revealed a higher probability of detecting a fraudulent transaction compared to solely used algorithms [9].

4 Materials and Methodology

4.1 Data Description

The dataset was obtained from the Kaggle website, which is available at <https://www.kaggle.com/imanolrecioerquicia/400k-nyse-random-investments-financial-ratios>. The dataset includes various financial features, such as "EPS_ratio," "inflation," "roe_ratio," and "expected_ratio," which are collected from more than 400,000 random investments about the NYSE market during the last 10 years. Furthermore, this dataset also involves ESG ranking, which has been a significant indicator these days. The total number of data is 405258, and it does not contain any missing values [10].

4.2 Exploratory Data Analysis

Through exploratory data analysis (EDA), we could figure out the relationship between certain variables. For instance, Fig. 2 shows the distribution of the “investment” column, compared to the “ESG_ranking” values, and it could be determined that there does not exist any obvious relationship between the two variables. Furthermore, Fig. 3 exhibits the density of the “ESG_ranking,” and it could be explained that the variable mostly contains high and low-ranking ones.

As the given dataset contains plentiful variables, a correlation function was applied for the feature selection. Through the correlation function, “Volatility_Buy,” “expected_return,” “nomina_return,” “NetProfitMargin_ratio,” and “current_ratio” were selected. Furthermore, the “ESG_ranking” variable was also added because it has been a hot issue around the world recently. Figure 4 exhibits the correlation

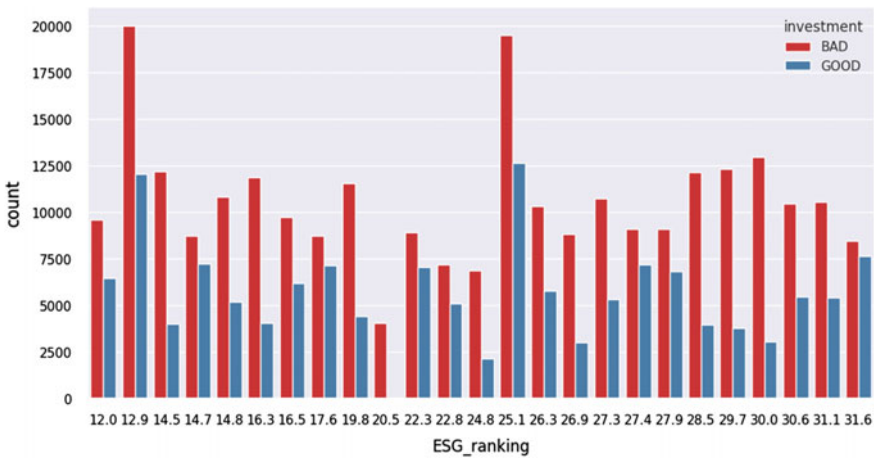


Fig. 2 Distribution of the “investment” variables compared to “ESG_ranking”

Fig. 3 The density of the ESG_ranking variable

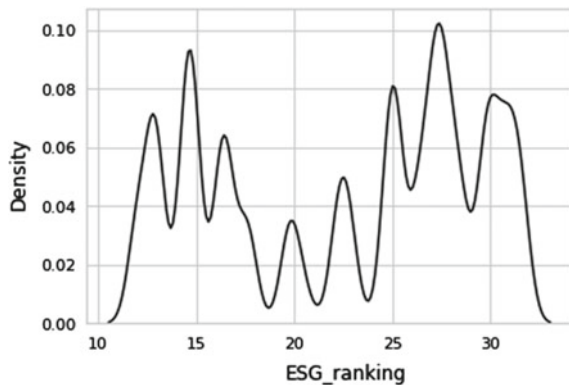
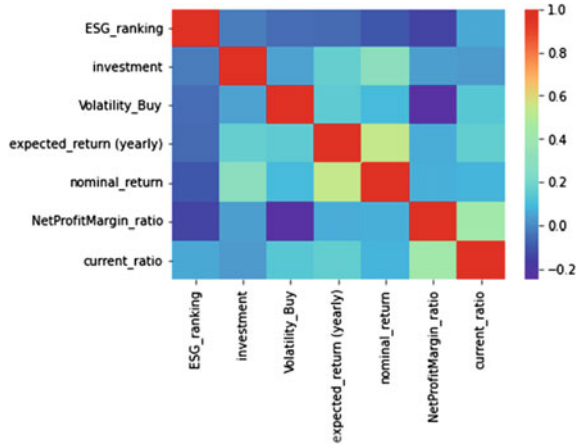


Fig. 4 Correlation heatmap among selected variables



among the selected variables. However, no significant correlation was found when they were paired with the “investment” value.

Jointplot from the “seaborn” library was also applied for the EDA. The joint plot exhibits the bivariate and univariate graphs from the two variables. However, from the figures below including Figs. 5, 6, and 7, they did not yield any relationship between the part of selected variables and “investment.”

Fig. 5 Jointplot between “ESG_ranking” and “investment”

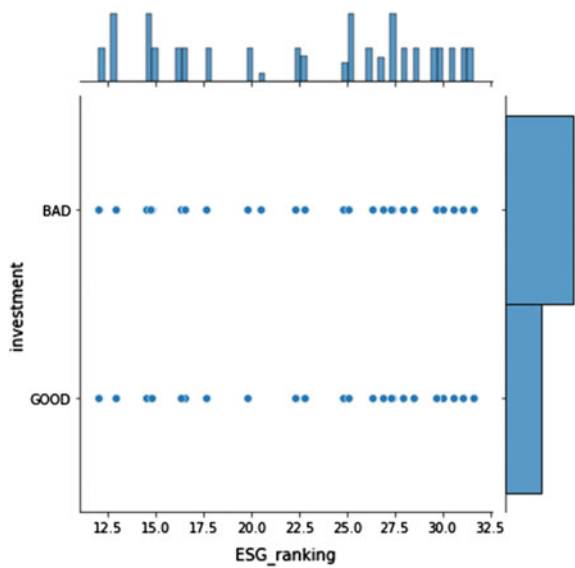


Fig. 6 Jointplot between “current_ratio” and “investment”

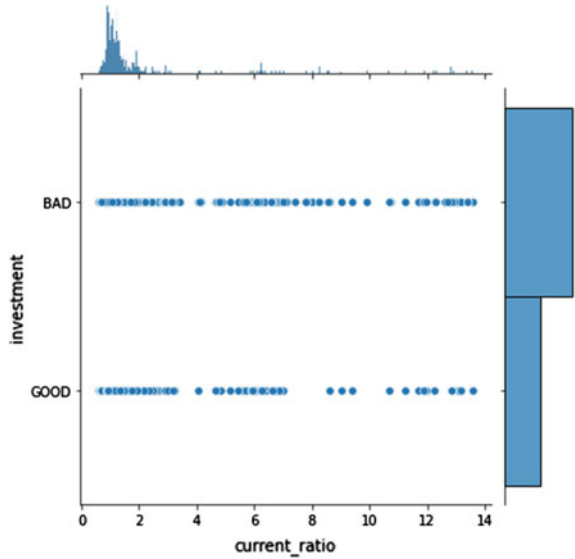
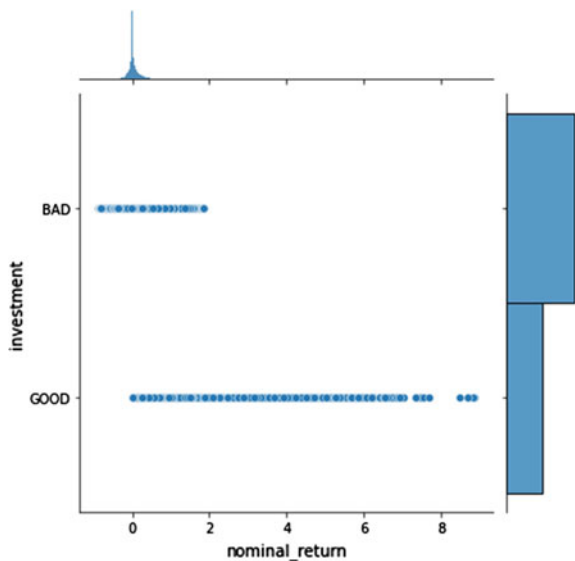


Fig. 7 Jointplot between “nomina_return” and “investment”



4.3 Light Gradient Boosting Machine

Light gradient boosting machine (LGBM) was a novel boosting algorithm, which was developed to solve the downsides of other existing boosting algorithms including gradient boosting and extreme gradient boosting (XG boosting). Those algorithms

could not yield high performance when dealing the dataset with the high dimension, as they consumed pretty much time for the computation. Therefore, two novel approaches were applied to the LGBM, which are gradient-based one-side sampling (GOSS) and exclusive feature bundling (EFB). GOSS is a technique based on the idea that data objects with large gradients play a greater role in information gain. Therefore, variables with large gradients are maintained, while variables with small gradients are randomly removed by a certain probability. EFB is a method that bundles the mutually exclusive variables. This method is applicable when the given dataset has a sparse variable space, including one-hot encoding. Through EFB, the mutually exclusive variables could be merged into one variable, and it could enhance the computation speed, and reduce memory usage [11].

4.4 Local Outlier Factors

Local outlier factors (LOF) represent the degree to which each observation deviates from the data (outlier). The most important feature of LOF is not to consider all data as a whole, but to use the neighbor of the observation to determine the degree of an outlier from a local perspective. For computing the LOF, it needs three values, which are k-distance, reach-distance, and local reachability density (LRD), per each. K-distance denotes the distance between a data point and its k-neighbors. Reach-distance indicates that when p is the maximum value of two variables, distance is calculated from the point q . Lastly, LRD is the reciprocal of the k-neighbor's reach-distance mean for point p , which means that it can represent how dense the neighborhood is around point p . LOF has the advantage of having to determine only a hyper-parameter called k , which indicates how much surrounding data to consider [12].

5 Results

5.1 Experiment #1

For the first stage of the experiment, various machine learning algorithms, including decision tree, logistic regression, gradient boosting, adaptive boosting, random forest, XG boosting, LGBM, and extra tree were applied. Before using those models, all datasets were normalized through a robust scaler and string values were labeled through a label encoder. The result shows that most of the classifiers performed well through obtaining higher than 90%, and only logistic regression achieved 67.51% of accuracy (refer Fig. 8). Particularly, LGBM, extra tree, gradient boosting, and decision tree yielded higher than 99% of accuracy. However, after applying Gaussian noise to the training dataset, the overall accuracy of the classifiers was decreased.

Accuracy score based on ordinal dataset

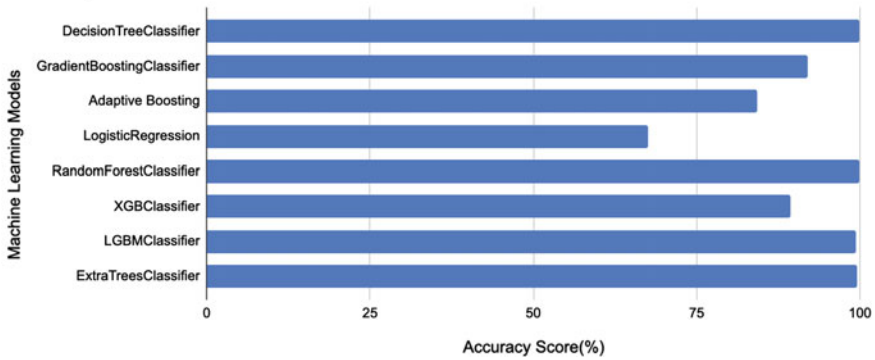


Fig. 8 Accuracy comparison with the ordinal dataset

Accuracy score based on noise-added dataset

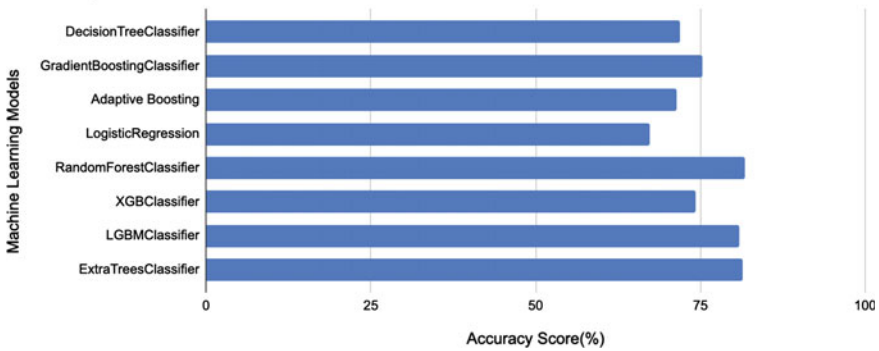


Fig. 9 Accuracy comparison with noise added dataset

The highest accuracy was 88.5% from LGBM, which dropped from 99.5%, and those other algorithms which performed higher than 99% fell by nearly 15% of accuracy (refer Fig. 9). These results exhibit the impact of the noise data was critical, which decreased the accuracy of the proposed machine learning models sharply. Furthermore, it also yielded a potential danger of noise data, which can be triggered from the adversarial attack.

5.2 Experiment #2

After the classification, the dataset was modified slightly in order to conduct an experiment for detecting the noise data. Therefore, the “investment” column, which was the previous target column, was deleted and then the new column “noise” was

added. The “noise” column only contained values of 1 and 0, and they indicated noise and normal data, per each. It contains only 1% of the noise data and the rest of it is normal data, as shown in Fig. 10. Two representative anomaly detection algorithms, which are local outlier factors (LOF) and isolation forest (IF), are utilized to detect the noise data. IF showed that the accuracy of the unsupervised anomaly detection model on the test set was 84.1%, while LOF yielded 95.156%. Furthermore, Figs. 11 and 12 show a confusion matrix from each outcome. This result indicated that even if the dataset contains noise data, from the adversarial attack, it can be detected efficiently throughout these algorithms.

Fig. 10 Number of each value, which belongs to the “noise” column

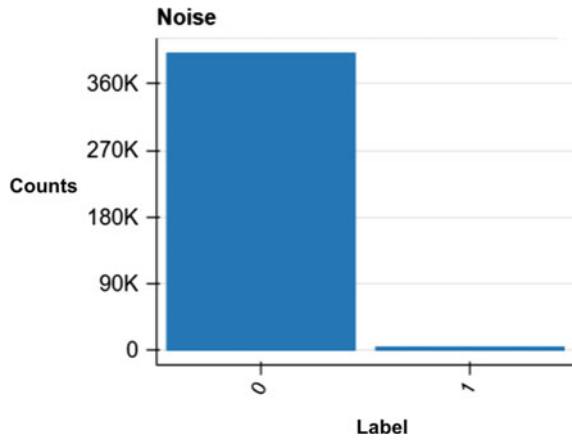


Fig. 11 Confusion matrix of the isolation forest

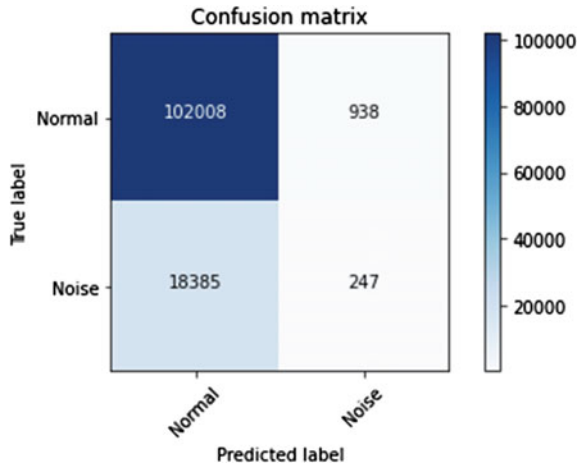
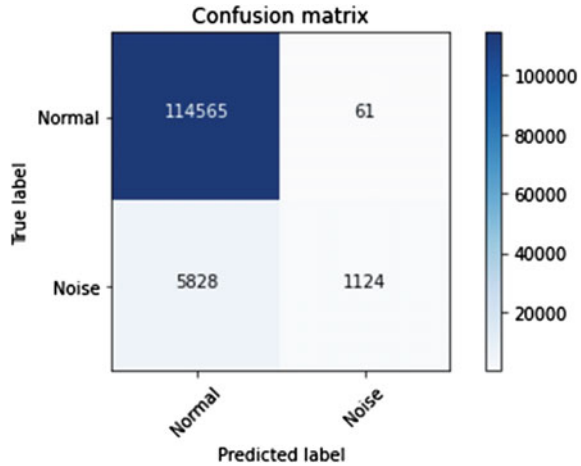


Fig. 12 Confusion matrix of the local outlier factors



6 Discussion

6.1 Principal Finding

The aim of this research is to alert people that the adversarial machine learning attack could diminish the performance of the machine learning models. Through two different experiments, the research firstly showed how seriously simple noise data could affect the performance of the models. Furthermore, by suggesting the isolation forest and local outlier factors, this paper suggests the possibility of detecting the noise data. Through these experiments, it could be concluded that even though very small values of noise were applied, the effect on the result was tremendous. Therefore, this research showed a possibility that enormous risk could happen in the real world with little noise in the real portfolio. This research could lead to enhanced, various further researches which could deal with the more sophisticated adversarial machine learning attack.

6.2 Limitation

Even though the proposed experiments were quite successful, there still exist some limitations. First of all, this research could propose the detecting algorithm for the noise data, but could not suggest how to filter the noise data to recover the accuracy score. Furthermore, as this paper suggests the effect of the noise on simple binary classification, further researchers should cover more sophisticated classification problems or regressions which could be more related to real-world problems.

7 Conclusion



In order to notify how dangerous, the adversarial machine learning attack is, this paper conducted two different experiments. In the first experiment, numerous machine learning algorithms were utilized to calculate the accuracy score of the classification, whether it was a good investment or not. With the ordinal dataset, which involves “Volatility_Buy,” “expected_return,” “nomina_return,” “NetProfit-Margin_ratio,” “current_ratio,” and “ESG_ranking,” most of the classifiers achieved higher than 90% of accuracy. However, after spreading the noise into the dataset, the performance decreased dramatically, which revealed that the highest accuracy was only 88.5%, which dropped from 99.5%. In the second experiment, the dataset was slightly altered to detect the noise data. LOF and IF yielded 95.156 and 84.1% of the accuracy of the unsupervised anomaly detection model on the test set. Even though this paper involved the limitation that it only conducted very simple binary classification and could not suggest how to recover the accuracy of the classifiers through using filters, the result is still worthwhile in alerting the danger of the attack and how to detect the noise data.

References

1. ADEC INNOVATIONS, What is ESG? <https://www.adececg.com/resources/faq/what-is-esg/> Accessed 28 Oct 2021
2. Nasdaq strong ESG practices can benefit companies and investors: here’s how, <https://www.nasdaq.com/articles/strong-esg-practices-can-benefit-companies-and-investors-2019-03-13> Accessed 28 Oct 2021
3. Toward datascience, Evasion attacks on machine learning (or “adversarial examples”). <https://towardsdatascience.com/evasion-attacks-on-machine-learning-or-adversarial-examples-12f2283e06a1> Accessed 28 Oct 2021
4. IT BUSINESS EDGE, Adversarial machine learning is used to attack machine learning systems. Learn how to identify and combat these cyberattacks. IT business edge, <https://www.itbusinessedge.com/development/adversarial-machine-learning-combating-data-poisoning/>. Accessed 28 Oct 2021
5. ESG STANDARD, <https://esgstandard.com/index.html>, Accessed 27 Oct 2021
6. Yu L, Huang X, Yin H (2020) Can machine learning paradigm improve attribute noise problem in credit risk classification? *Int Rev Econ Finan* 70:440–455
7. Margot V, Geissler C, de Franco C, Monnier B (2021) ESG investments: filtering versus machine learning approaches. *Appl Econ Finan* 8(2)
8. De Lucia C, Paziienza P, Bartlett M (2020) Does good ESG lead to better financial performances by firms? machine learning and logistic regression models of public enterprises in Europe. *Sustainability* 12(13) MDPI
9. Cheon MJ, Lee DH, Joo HS, Lee O (2021) Deep learning based hybrid approach of detecting fraudulent transactions. *J Theor Appl Inf Technol* 99(16):4044–4054
10. Kaggle, 400k NYSE random investments + financial ratios, <https://www.kaggle.com/imanolrecioerquicia/400k-nyse-random-investments-financial-ratios>, Accessed 4 Oct 2021
11. Ke G, Meng Q, Finley T, Wang T, Chen W, Ma W, Ye Q, Liu T (2017) LightGBM: a highly efficient gradient boosting decision tree. *Adv Neural Inf Process Syst* 30:3146–3154
12. Breunig MM, Kriegel HP, Ng RT, Sander J (2000) LOF. In: Proceedings of the 2000 ACM SIGMOD international conference on management of data, pp 93–104

Positioning Comparison Using GIM, Klobuchar, and IRI-2016 Models During the Geomagnetic Storm in 2021



Worachai Srisamoodkham, Kutubuddin Ansari ,
and Punyawati Jamjareegulgarn 

Abstract This paper compares the positioning accuracy obtained from the GIM VTEC, the Klobuchar model, and the IRI-2016 model at Chiang Mai and DPT9 stations, Thailand, during an intense geomagnetic storm of 2021 (on May 12, 2021). The results show that the diurnal variation of the Klobuchar modeled VTECs show the same trend as that of the observed GIM VTECs with the same peaks and the maximum deviation of 22.5% at 05:00 UT. Meanwhile, the IRI2016-predicted VTECs show its peak at 07:00 UT and are not available obviously during 13:00–21:00 UT due to the impact of this intense geomagnetic storm. Most of the ionospheric delays obtained from the Klobuchar model underestimate those of the GIM VTEC, whereas they overestimate those of GIM VTEC during after midnight and pre-sunrise period. At both stations, the mean ionospheric range delays of the GIM VTEC are highest during daytime period while those of the IRI-2016 model are largest during nighttime period. The positioning errors at higher latitude (CHMA station) are larger than those at lower latitude (DPT9 station).

Keywords GIM TEC · IRI-2016 model · Klobuchar model · Positioning

1 Introduction

The ionosphere is a layer of Earth's upper atmosphere ranging from 50 to 1,000 km. It is characterized by highly dynamical plasma density where the free ions and electrons

W. Srisamoodkham

Faculty of Agricultural and Industrial Technology, Phetchabun Rajabhat University, Phetchabun, Thailand

e-mail: hs5xij@pcru.ac.th

K. Ansari

Integrated Geoinformation (IntGeo) Solution Private Limited, New Delhi, India

e-mail: kdansarix@gmail.com

P. Jamjareegulgarn (✉)

Prince of Chumphon Campus, King Mongkut's Institute of Technology Ladkrabang, Chumphon 86160, Thailand

e-mail: kjpunyaw@gmail.com

affect largely both the refraction and the retardation of satellite signals. Hence, the ionospheric delay is one of the main error sources of the global navigation satellite systems (GNSSs), for example, GLONASS, Galileo, Beidou, QZSS, and GPS, etc., which have been employed increasingly for numerous applications. The ionospheric delay is proportional to the total electron content (TEC) along the line of sight (LOS), and inversely proportional to the signal frequency. Generally, the ionospheric error covers from a few meters to tens of meters at the zenith and can increase beyond 100 m under extreme space weather situations. The violence of delay errors relies on time of day, location, season, solar cycle, and other anomalies and irregularities [1–3]. The estimation failures of the ionospheric error make both the cycle slip correction and the ambiguity resolution more difficult and result in positioning errors in long baseline solutions [4]. It is well known that the ionospheric delay can be eliminated by using range measurements for any dual-frequency GNSS receivers. In contrast, it can be compensated to get the actual positions using the ionospheric models for single-frequency GNSS receivers. Models using ionospheric parameters broadcasted along with navigation messages have been widely employed to mitigate the ionospheric influences on signal propagation for the users of single-frequency GNSS receivers.

The ionospheric delay can be mitigated up to different levels based on various ionospheric error mitigation techniques. The main solution to neglect the ionospheric effects can be conducted by considering the GNSS observation during nighttime period, which assumes that the ionospheric condition is almost quiet. The most widely used model for single-frequency GPS receiver is the Klobuchar model [5]. Klobuchar developed an algorithm to give 50% root mean square correction for the ionospheric delay. The coefficients were computed from an empirical model of global ionospheric behaviors as functions of solar activity and time of year and broadcasted through the GPS navigation message. It can be used to compute a slant ionospheric delay to each satellite at various elevation and azimuth directions that are then applied to determine the final pseudoranges. However, the accuracy of Klobuchar model is deteriorated because of the intensities of solar activity and geomagnetic storm.

Thailand is located close to the geomagnetic equator within the equatorial ionization anomaly (EIA) region at $\pm 15^\circ$ in latitude where the TEC often fluctuates and can be affected by the geomagnetic storm. So, the positioning accuracy can be deteriorated at the EIA region [6]. Hence, this paper is aimed to identify the positioning accuracy of Klobuchar model and IRI-2016 model during the most intense geomagnetic storm of this year till now (on May 12, 2021) over Thailand region.

2 Data Used

The raw data on an intense geomagnetic storm of year 2021 (storm level G3, Ap = 42 and Kp = 7 between 12:00 and 18:00 UT) were gathered from the GNSS stations at Chiang Mai, namely CHMA (lat. 18.84°N, long. 98.97°E) and DPT9 (lat. 13.76°N, long. 100.57°E) stations from Department of Public Work and Country

Planning. Afterward, the raw data were converted to the RINEX (Receiver Independent Exchange Format) files which include the navigation data, observation data, and Klobuchar coefficients. This information was employed to estimate the ionospheric delay as described in the next section. Moreover, the vertical total electron content (VTEC) values from Global Ionospheric Model (GIM) map were also employed in this paper that can be downloaded through a web site: <https://urs.earthdata.nasa.gov/>. Likewise, the VTEC values obtained from the IRI-2016 model were also used to compare with the GIM VTEC and Klobuchar modeled VTECs and can download from a web site: https://ccmc.gsfc.nasa.gov/modelweb/models/iri2016_vitmo.php.

3 Ionospheric Delay

Firstly, we start computing the ionospheric delay of Klobuchar model. The Klobuchar model has been utilized to compensate the GPS positioning accuracy for single-frequency GPS users since 1987. The GPS satellites broadcast eight coefficients of the Klobuchar model to estimate the ionospheric delay using two main issues as follows: (a) the electron content is assumed to be concentrated as a thin layer at the height of 350 km and (b) the slant delay is computed from the vertical delay at the ionospheric pierce point (IPP) multiplying by an obliquity factor. Mathematical equations for computing the ionospheric delay of Klobuchar ionospheric model can be thoroughly found in [7]. Here, we show shortly the equation to compute vertical ionospheric time delay (I_d) of L1 frequency (1.575 MHz) in unit: ns as follow:

$$I_{d_klo} = \begin{cases} \left[5 \cdot 10^{-9} + A_I \cdot \left(1 - \frac{X_I^2}{2} + \frac{X_I^4}{24} \right) \right]; |X_I| \leq 1.57 \\ 5 \cdot 10^{-9}; |X_I| > 1.57 \end{cases} \quad (1)$$

where A_I and X_I are the amplitude and phase of the ionospheric delay, respectively.

Afterward, the VTEC of Klobuchar model ($VTEC_{KLO}$) can be computed as follow.

$$VTEC_{KLO} = \frac{I_{d_klo} \cdot c \cdot f_{L1}^2}{40.3} \quad (2)$$

where c is the light velocity.

Secondly, the GIM VTEC data ($VTEC_{GIM}$) are routinely computed and stored in a large database owned by International GNSS Service (IGS). The ionospheric delay based on GIM VTEC data can also be determined by the following expression.

$$I_{d_GIM} = \frac{40.3 \times VTEC_{GIM}}{c \cdot f_{L1}^2} \quad (3)$$

Thirdly, the IRI VTEC data ($VTEC_{IRI}$) are also retrieved from a large database owned by the NASA and NSF organizations. The ionospheric delay based on IRI VTEC data can also be calculated by the below expression.

$$I_{d_IRI} = \frac{40.3 \times VTEC_{IRI}}{c \cdot f_{L1}^2} \quad (4)$$

As for Eqs. (3) and (4), we assume that the GNSS receivers can receive and process the $VTEC_{GIM}$ and $VTEC_{IRI}$ values and employ them to compute the I_{d_GIM} and I_{d_IRI} values, respectively.

4 Ionospheric Range Delay

The ionospheric range delay (I_r) of each considered model can be computed as follow.

$$I_{r*} = I_{d*} \times c \quad (5)$$

where I_{d*} represent the ionospheric delays of the observed GIM VTECs, the Klobuchar modeled VTECs, and the IRI2016-predicted VTECs, respectively; then, the I_{r*} denote the respective ionospheric range delays (i.e., I_{r_GIM} , I_{r_KLO} and I_{r_IRI}).

5 Results and Discussion

As for our experiments, we start calculating the ionospheric delays and VTEC values for all GNSS constellations at both CHMA and DPT9 stations over Thailand region during the most intense storm of year 2021 till now (on May 12, 2021). We find that the QZSS constellations can provide the best Klobuchar modeled VTEC values that behave the same trend as the GIM VTEC values. Therefore, all parameters related to the Klobuchar model in this paper are owned by the QZSS constellations. Firstly, the VTEC values obtained from the observed GIM, the Klobuchar model, and the IRI-2016 model are processed and analyzed. Their VTEC results are shown in Fig. 1 and Table 1. It is seen that the diurnal variation of the Klobuchar modeled VTECs shows the same trend as that of the observed GIM VTECs with the same peaks and the maximum deviation of 22.5% at 05:00 UT (12:00 LT). Meanwhile, the IRI2016-predicted VTEC values show differently its peak at 07:00 UT. Also, they are not available during 13:00–21:00 UT due to the impact of this intense geomagnetic storm.

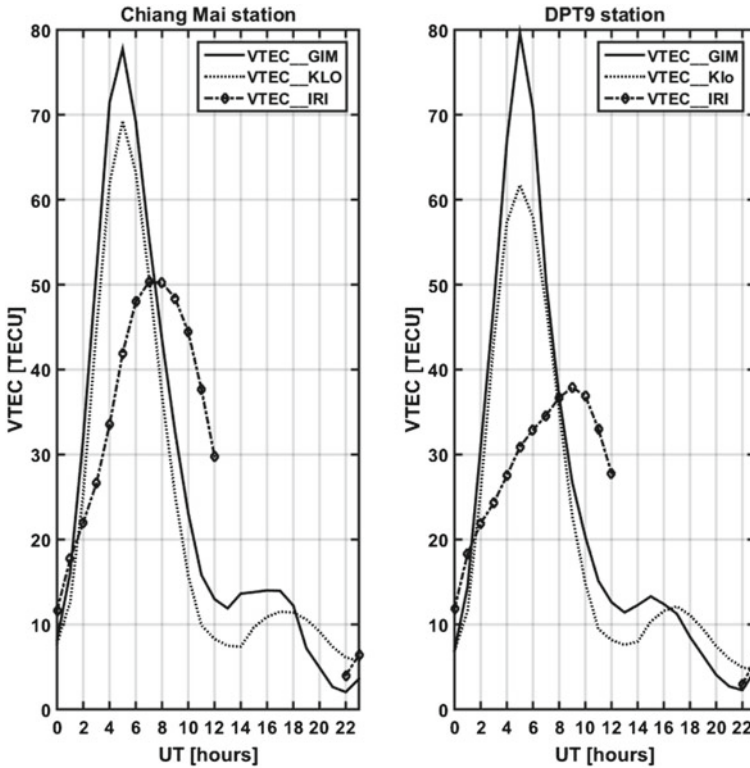


Fig. 1 The studied VTEC values obtained from CHMA and DPT9 on May 12, 2021

Table 1 The VTEC statistics (unit: TECU) of the observed GIM VTEC, Klobuchar model, and IRI-2016 models at Chiang Mai and DPT9 stations on May 12, 2021

Periods	VTEC statistics (occurrence time)	Chiang Mai station		
		Observed GIM	Klobuchar	IRI-2016
Daytime	Max. VTEC	77.75 (5 UT)	69.28 (5 UT)	50.3 (7 UT)
	Min. VTEC	8.16 (0 UT)	7.73 (0 UT)	11.6 (0 UT)
Nighttime	Max. VTEC	14.00 (16 UT)	11.49 (17 UT)	
	Min. VTEC	2.03 (22 UT)	5.72 (23 UT)	-
DPT9 station				
Daytime	Max. VTEC	79.66 (5 UT)	61.73 (5 UT)	37.9 (9 UT)
	Min. VTEC	6.90 (0 UT)	6.90 (0 UT)	11.8 (0 UT)
Nighttime	Max. VTEC	13.29 (15 UT)	12.11 (17 UT)	-
	Min. VTEC	2.26 (22 UT)	4.65 (23 UT)	-

At both stations, the ionospheric delay statistics of the observed GIM are mostly higher than those of the Klobuchar model and the IRI-2016 model during the storm day. However, the ionospheric delays during the whole day at CHMA station are mostly larger than those at DPT9 station as shown in Fig. 2 and Table 2. It can be seen obviously that the ionospheric delays during daytime period are bigger than those during nighttime period because of the existence of photoionization process. Refer to the ionospheric delay differences between the GIM and Klobuchar models of these two stations, the maximum and minimum values during daytime period are about 4.59–9.70 ns around 5:00 UT and about 0.00 ns at 0:00 UT, respectively. Meanwhile, those values during nighttime period are smaller of 2.41–3.38 ns around 13:00 UT and about 0.00 ns at different hours, respectively. It is worthy to note that most of the ionospheric delays obtained from the Klobuchar model underestimate those of the GIM VTEC, whereas they overestimate those of GIM VTEC between 17:00 UT and 23:00 UT (00:00–06:00 LT). These results are good agreements with the results of Jongsintawee et al. [8].

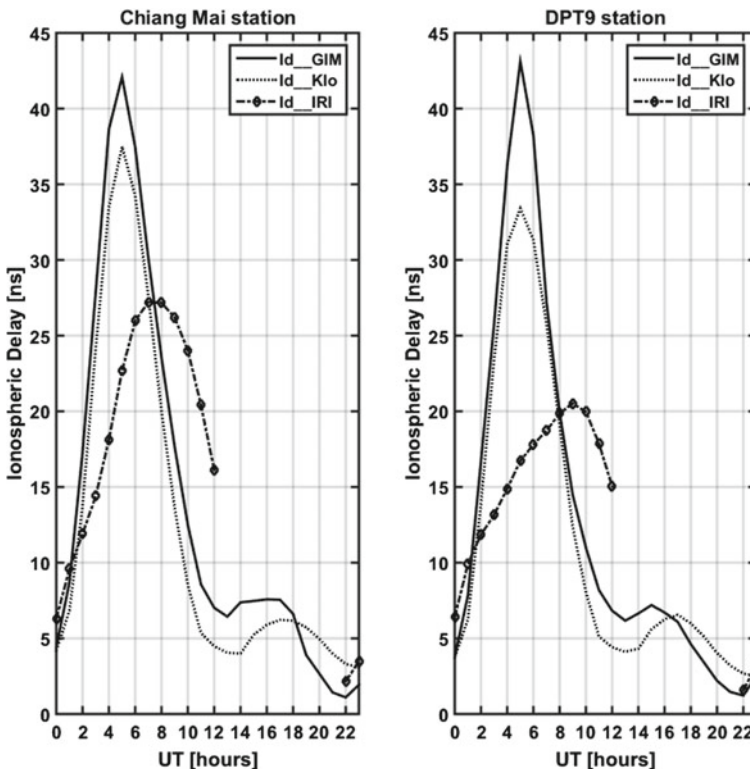


Fig. 2 The studied ionospheric delays for CHMA and DPT9 stations on May 12, 2021

Table 2 The ionospheric delays (unit: ns) of the GIM VTEC, Klobuchar model, and IRI-2016 models at Chiang Mai and DPT9 stations on May 12, 2021

Periods	Ionospheric delay statistics (Id)	Chiang Mai station		
		observed GIM	Klobuchar	IRI-2016
Daytime	Max. Id	42.08	37.50	27.22
	Min. Id	4.42	4.18	6.28
	Mean Id	22.41	19.11	19.50
Nighttime	Max. Id	7.58	6.22	–
	Min. Id	1.10	3.10	–
	Mean Id	5.09	4.76	–
DPT9 station				
Daytime	Max. Id	43.11	33.41	20.51
	Min. Id	3.73	3.73	6.39
	Mean Id	21.02	17.78	15.64
Nighttime	Max. Id	7.20	6.55	–
	Min. Id	1.22	2.51	–
	Mean Id	4.59	4.57	–

As for the ionospheric range delays, the studied results are similar to the results of the ionospheric delays shown in Fig. 3 and Table 3. During daytime, the averaged range delays of the GIM VTEC are highest at both stations, while the averaged range delays of the IRI-2016 model are largest during nighttime period at both stations. Moreover, during daytime, the I_{r_GIM} , I_{r_KLO} and I_{r_IRI} at both stations behave the decreasing trend, respectively, as a result of the orderly lower VTEC values. Meanwhile, during night time period, the I_{r_IRI} values seem to be highest as compared to I_{r_GIM} and I_{r_KLO} , although the IRI-2016 data are not available on this geomagnetic storm. It can be seen clearly that the positioning errors at higher latitude (CHMA station) are larger than those at lower latitude (DPT9 station) as reported in Jongsintawee et al. [8].

6 Conclusion

This paper compares the positioning accuracy obtained from the GIM VTEC, Klobuchar model, and the IRI-2016 model at Chiang Mai and DPT9 stations over Thailand region during an intense storm on May 12, 2021. The VTEC variations of

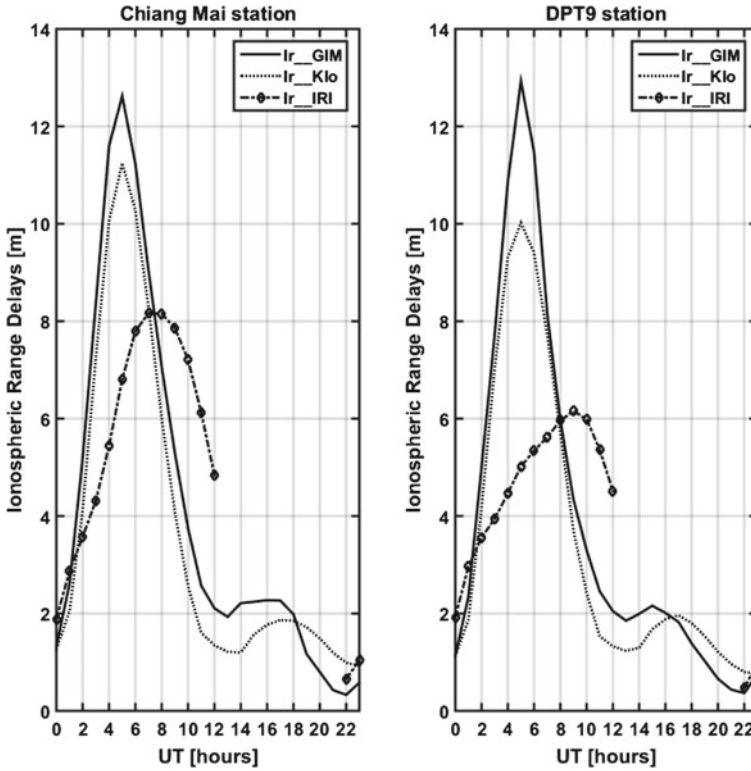


Fig. 3 The studied ionospheric range delays for CHMA and DPT9 stations on May 12, 2021

GIM map behave the same trend as those of the Klobuchar model, so the ionospheric delays and the ionospheric range delays behave the same trends as the VTECs. The results show that the Klobuchar model is suitably employed for compensating the ionospheric delays during 07:00–23:00 LT, whereas it should be improved during 00:00–06:00 LT due to the estimation beyond the GIM VTEC. Moreover, the IRI2016-predicted VTECs should be improved to be higher during daytime period and be more robust during the intense geomagnetic storm.

Table 3 The ionospheric range delays (unit: m) of the GIM VTEC, Klobuchar model, and IRI-2016 models at Chiang Mai and DPT9 stations on May 12, 2021

Periods	Ionospheric range delay statistics (Ir)	Chiang Mai station		
		observed GIM	Klobuchar	IRI-2016
Daytime	Max. Ir	12.62	11.25	8.17
	Min. Ir	1.32	1.26	1.88
	Mean Ir	6.72	5.73	5.85
Nighttime	Max. Ir	2.27	1.87	–
	Min. Ir	0.33	0.93	–
	Mean Ir	1.53	1.43	–
DPT9 station				
Daytime	Max. Ir	12.93	10.02	6.15
	Min. Ir	1.12	1.12	1.92
	Mean Ir	6.31	5.33	4.69
Nighttime	Max. Ir	2.16	1.97	–
	Min. Ir	0.37	0.75	–
	Mean Ir	1.38	1.37	–

Acknowledgements This research is financially supported by Broadcasting and Telecommunications Research and Development Fund for Public Interest (project code: B2-001/6-2-63). The authors would like to express the gratitude to OMNI Web of NASA for the IRI-2016 models and to the EARTHDATA web of NASA for GIM TEC. Moreover, the authors would like to thank to Department of Public Work and Town and Country Planning (DPT) for the RINEX data.

References

- Panda SK, Gedam SS (2016) Evaluation of GPS standard point positioning with various ionospheric error mitigation techniques. *J Appl Geodesy* 10(4):1–11
- Júnior PTS, Alves DBM, da Silva CM, Klobuchar NeQuick G (2019) Ionospheric models comparison for Multi-GNSS single-frequency code point positioning in the Brazilian region. *Bull Geodetic Sci* 25(3):e2019016
- Li J, Wan Q, Ma G, Zhang J, Wang X, Fan J (2017) Evaluation of the Klobuchar model in Taiwan. *Adv Space Res* 60(6):1210–1219
- Mageed KMA (2014) Effect of using Klobuchar, CODE, and no-ionosphere models on processing single frequency GPS static medium baselines. *Int J Sci Eng Res* 5(5)
- Klobuchar JA (1987) Ionospheric time-delay algorithm for single-frequency GPS users. *IEEE T Aero Elec Sys* 23:325–331
- Lee C (2011) Generation of Klobuchar ionospheric error model coefficients using Fourier series and accuracy analysis. *J Astron Space Sci* 28(1):71–77
- Klobuchar Model, https://gssc.esa.int/navipedia/index.php/Klobuchar_Ionospheric_Model Accessed 10 July 2021
- Jongsintawee S, Rungraengwajiake S, Supnithi P, Panachart C (2016) Comparison of GPS positioning accuracy using Klobuchar model and IGS TEC model in Thailand. *KMITL Sci Tech J* 16(1):1–10

Wearable Patch Antennas on Fr4, Rogers and Jeans Fabric Substrates for Biomedical Applications



Regidi Suneetha  and P. V. Sridevi 

Abstract Wearable antennas with dimensions of $24.5 \times 15 \times 1.6 \text{ mm}^3$ for biomedical applications are designed on FR4, Rogers RT/Duroid 5880 and jeans fabric substrate for stroke imaging applications. These antennas accomplish the ease of wearability in daily wear accessories like watches, helmets and pockets with flexibility in antennas' orientation due to omnidirectional radiation pattern. The structure is simple, miniaturization is also attained due to defected ground structure and slots insertion in Fr4 and Rogers RT/Duroid 5880 substrate models, and partial ground is used in the case of jeans fabric substrate. All three antennas are performing well for biomedical applications like patient monitoring and microwave imaging applications.

Keywords Biomedical applications · Body centric communication · Microwave frequencies · Microwave imaging · Wearable antenna

1 Introduction

In the modernized world of communication systems, radiofrequency (RF) devices need to be of small size with productive microwave circuits [1] multiband applications of present-day wireless communication systems [2, 3]. Broadband type and multiband type are the kinds of antennas covering different bands of frequency ranges. Compared to a wideband antenna, a multiband antenna is operating at numerous desired frequency bands. Advanced study of microwaves due to the number of advantages, high accessibility, low attenuation with non-ionizing radiation dispenses more feasible methods of stroke [4] and microwave imaging [5, 6]. Microwave medical imaging is an advanced tool used to diagnose distinguish different deadly diseases like cancer. It uses electromagnetic radiation of 0.3–9.0 GHz frequency range. Due to the limited availability of machinery like computed tomography (CT scan) and magnetic

R. Suneetha (✉) · P. V. Sridevi
Andhra University College of Engineering (A), Andhra University, Visakhapatnam, Andhra Pradesh, India
e-mail: rsuneetha@rocketmail.com

resonance imaging (MRI) with precarious radiation effects owing to ionization especially for patients with long-term usage, microwave imaging presents revolutionary results with more ease at low cost.

A wide range of different types of antennas are being proposed throughout the research. Different kinds of antennas like dipole antennas, slot dipole antennas [7], loop antennas [8] and helical folded dipole [9] are used in patient monitoring systems. Due to low profile, planar constitution, low volume and lightweight with ease of fabrication at low cost allow microstrip patch antennas for wide usage in wireless communication systems applications.

In this paper, microstrip patch antennas using Fr4 substrate, Rogers RT/Duroid 5880 substrate and jeans fabric substrate are discussed with dimensions of $24.5 \times 15 \times 1.6 \text{ mm}^3$ for biomedical applications like stroke imaging in the frequency range between 1–10 GHz.

2 Design of the Antennas

The antennas are designed and simulated using HFSS software. The dimensions of the conventional microstrip patch antenna can be designed using equation [1].

The effective dielectric constant of the substrate is

$$\epsilon_{\text{eff}} = \frac{1}{2} \left(\epsilon_r + 1 + (\epsilon_r - 1) \sqrt{\frac{w + 12h}{w}} \right) \quad (1)$$

$$w = \frac{C}{2fo} \left(\sqrt{\frac{2}{\epsilon_r + 1}} \right) \quad (2)$$

where h is the height of the substrate, w is the width of the patch. ϵ_r is the dielectric constant of the substrate, C is the velocity of light in free space, and fo is the center frequency of the band.

Length of the patch can be obtained from

$$L = L_{\text{eff}} - 2\Delta L \quad (3)$$

where the effective length and ΔL of the patch can be obtained from

$$L_{\text{eff}} = \frac{c}{2fo} \left(\frac{1}{\sqrt{\epsilon_{\text{eff}}}} \right) \quad (4)$$

$$\Delta L = 0.412h \frac{(\epsilon_{\text{eff}} + 0.3) \left(\frac{w}{h} + 0.264 \right)}{(\epsilon_{\text{eff}} - 0.258) \left(\frac{w}{h} + 0.8 \right)} \quad (5)$$

1. The rectangular slotted patch [10, 11] antennas with defected ground structure [12, 13] and microstrip line feeding are proposed with dimensions of $24.5 \times 15 \times 1.6 \text{ mm}^3$ on Fr4 and Rogers RT/Duroid 5880 substrate materials [14, 15] as shown in Fig. 1a, b. It is to escalate the current path and also reduce the lowest resonant frequency, by the addition of slots, miniaturization is being achieved.

- Fr4 substrate with a dielectric constant of 4.4 and loss tangent of 0.02 with 50Ω microstrip line feeding is used.
- Rogers RT/Duroid 5880 substrate with a dielectric constant of 10.2 and loss tangent of 0.02 with 50Ω microstrip line feeding is used.

2. A rectangular patch antenna is proposed with dimensions of $24.5 \times 15 \times 1.6 \text{ mm}^3$ on jeans substrate [15] materials with a dielectric constant of 1.67 and loss tangent of 0.02 with 50Ω microstrip line feeding as shown in Fig. 1c, d

The dimensions of the antennas are mentioned in Table 1.

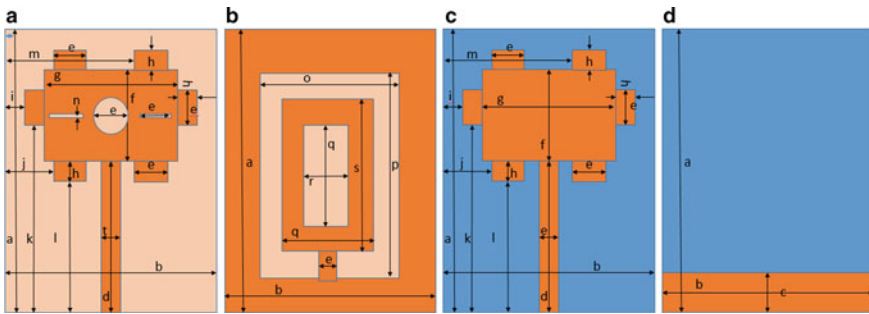


Fig. 1 Proposed design of antennas with dimensions. **a** Patch and **b** ground of Fr4 and Rogers RT/Duroid 5880 **c** patch and **d** ground of jeans substrate antenna

Table 1 Dimensions of the designed antennas

S. No.	Alphabet	Value (mm)	S. No.	Alphabet	Value (mm)
1	a	24.5	11	k	15
2	b	15	12	l	11.2
3	c	2.5	13	m	9.5
4	d	12.6	14	n	0.5
5	e	2	15	o	11
6	f	6	16	p	20
7	g	10.6	17	q	7
8	h	1.4	18	r	3
9	i	0.8	19	s	16
10	j	3.8	20	t	1.6

3 Discussion of Results

The antenna parameters such as reflection coefficient (S_{11}), radiation pattern, VSWR (voltage standing wave ratio), gain, and surface current distribution are being discussed along with SAR (specific absorption rate) analysis. All the parameters are simulated and comparison plots are plotted for designed antennas. The dimensions and structure used for antenna design are the same for all three antennas.

The reflection coefficient in terms of S parameters, S_{11} plot, is as shown in Fig. 2. The minimum value of S_{11} is of -29.32 dB at 5.9 GHz, -21.59 dB at 6.3 GHz and -16.76 dB at 3.9 GHz for Fr4, Rogers RT/Duroid 5880 and jeans substrates, respectively. The minimum value of VSWR is of 1.07 dB at 5.9 GHz, 1.18 dB at 6.3 GHz and 1.33 dB at 3.9 GHz for Fr4, Rogers RT/Duroid 5880 and jeans

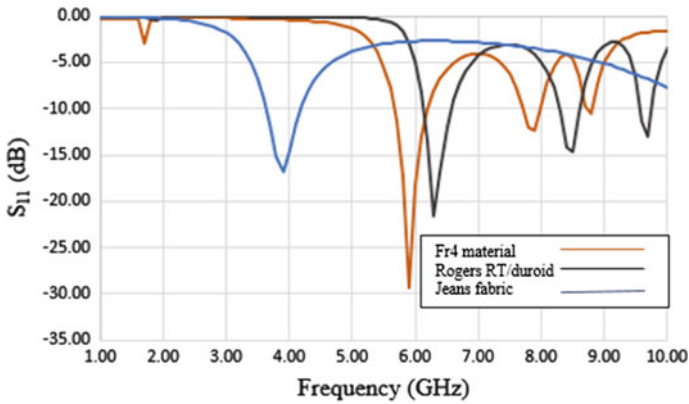


Fig. 2 Simulated S_{11} versus frequency plot of designed antennas in free space

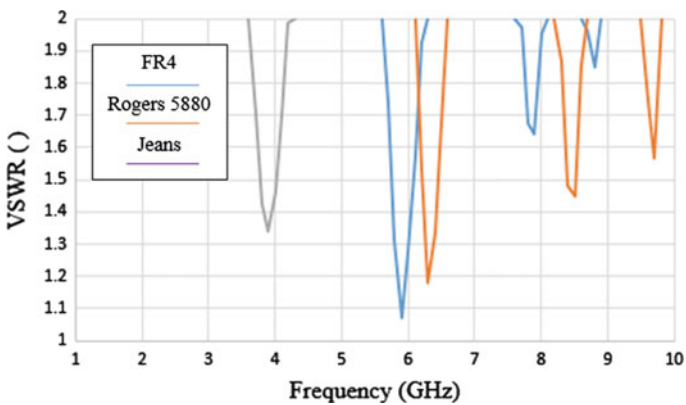


Fig. 3 Simulated VSWR versus frequency plot of designed antennas

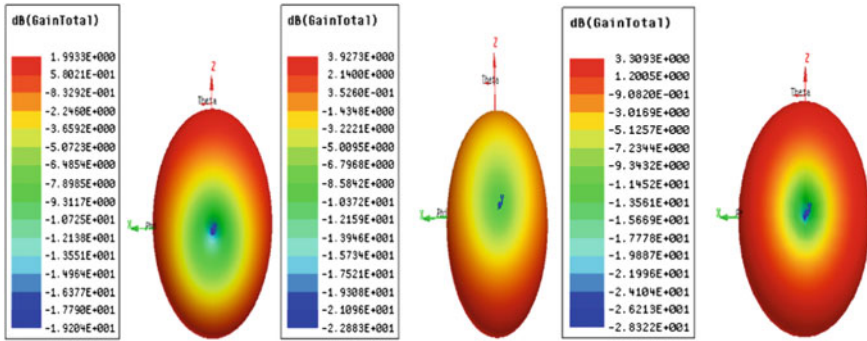


Fig. 4 Simulated radiation pattern of the designed antennas

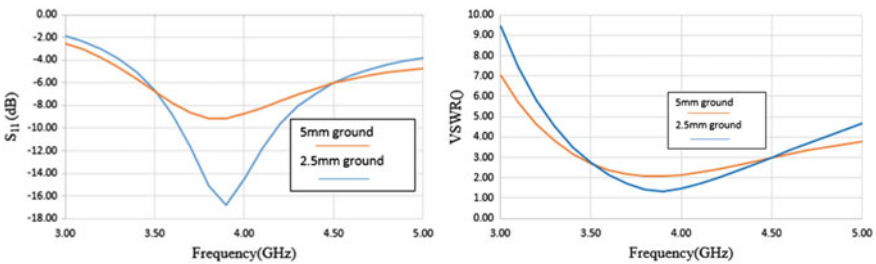


Fig. 5 Simulated S_{11} versus frequency plot and VSWR versus frequency plot of the jeans fabric antenna

substrates, respectively. The VSWR versus frequency plot is shown in Fig. 3. The omnidirectional radiation pattern is obtained for all three antennas as shown in Fig. 4.

Figure 5 shows the simulated results of S_{11} and VSWR plots of jeans substrate patch antenna for different values of ground, and the 2.5-mm ground is considered for the final design of jeans substrate. The surface current distribution is shown in Fig. 6. That can be used to change the shape and structure of the antenna. It is noticed that the current distribution is mostly concentrated at the feed line at lower frequencies in the direction of the X-axis, and the current intensity is maximum at higher frequencies concentrated at the edges of the patch and ground plane across the feed line.

It is the nature of the human body to have different dielectric constant values due to the non-homogenous constitution of the body like skin, bones, blood and skull etc. It exhibits different electromagnetic parameters at different values of frequency. The human phantom model of three layers muscle 10 mm, fat 4 mm and skin 2 mm is applied to obtain SAR analysis. The model is shown in Fig. 7. The electrical properties of tissues used are mentioned in Table 2.

Specific absorption rate (SAR) [16] is the exposure measurement of the human body for wireless devices or electromagnetic radiations.

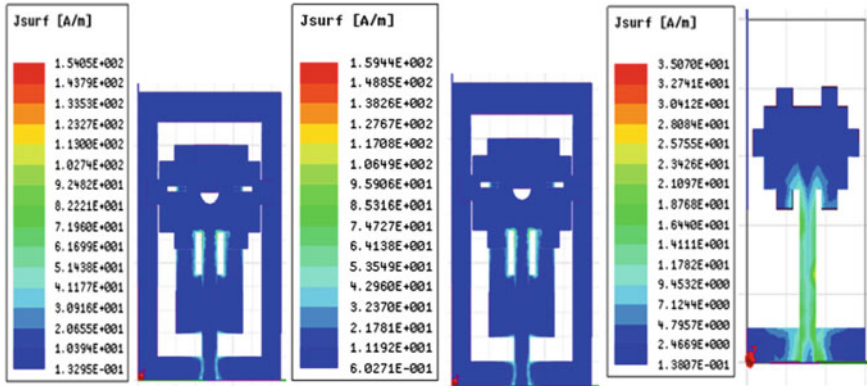


Fig. 6 Surface current distribution of designed antennas

Fig. 7 Three-layer human phantom model

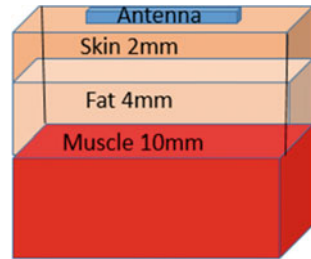


Table 2 Electrical properties of human tissues

Tissue	Source	Permittivity		Elec. Cond. (S/m)		Thickness (mm)
		4 GHz	5.7 GHz	4 GHz	5.7 GHz	
Skin	Skin (Dry)	36.58	35.19	2.34	3.63	2
Fat (Not infiltrated)	Fat (Not infiltrated)	5.12	4.96	0.18	0.28	4
Muscle	Muscles	50.82	48.61	3.01	4.84	10

$$SAR = \frac{1}{V} \oint \frac{\sigma(x)|E(x)|^2}{\rho(x)} dx \tag{6}$$

where

- Sample electrical conductivity— σ
- Electric field intensity— E ,
- Charge density— ρ ,
- Sample volume— V .

The international guidelines of human safety, IEEE C95-1-2005 standards and ICNIRP define maximum permissible SAR to preserve over 10-g tissue must be less than 2 W/kg. All the three antennas are performing well in the limited range of SAR levels with Fr4 substrate of 1.99 W/Kg and Rogers RT/Duroid 5880 substrate of 1.92 W/Kg at 5.7 GHz frequency and jeans substrate antenna of 1.25 W/Kg at 4 GHz as shown in Fig. 9. The S_{11} versus frequency plot when an antenna is placed 2 mm above the human phantom model is shown in Fig. 8. The plot shows that the antennas are performing well in the frequency of operation. The details are tabulated in Table 3. Comparison of different antennas from literature are tabulated in Table 4.

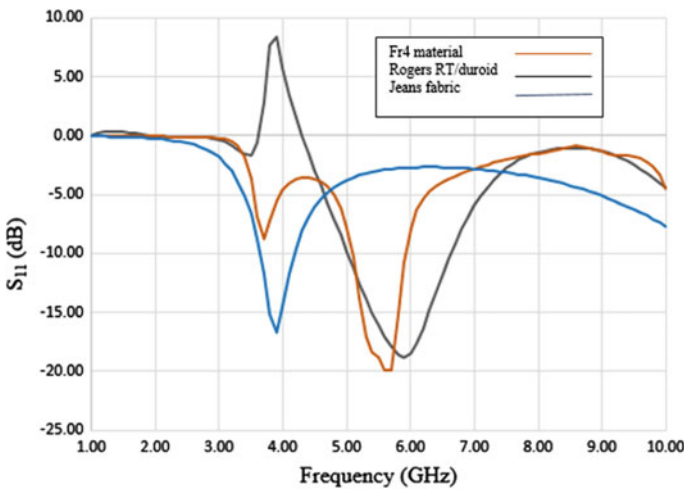


Fig. 8 Simulated S_{11} versus frequency plot of designed antennas when placed above human phantom model

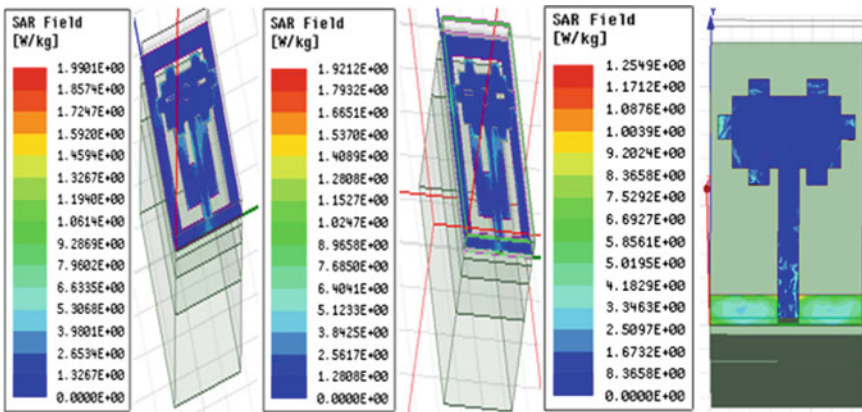


Fig. 9 Simulated SAR values of designed antennas

Table 3 Comparison of designed antennas

Model	Substrate	Dimensions (mm ³)	S ₁₁ (dB)	Gain (dB)	SAR (W/Kg)
Slotted antenna with defected ground structure	Fr4	24.5 × 15 × 1.6	−29.32 at 5.9 GHz	1.99	1.99
Slotted antenna with defected ground structure	Rogers RT/Duroid 5880	24.5 × 15 × 1.6	−21.59 at 6.3 GHz	3.92	1.92
Partial ground	Jeans fabric	24.5 × 15 × 1.6	−16.76 at 3.9 GHz	3.30	1.25

Table 4 Comparison of different antennas from literature

Model	Substrate	Dimensions (mm ²)	S ₁₁ (dB)
Ref [17]	Polyethylene foam	25 × 35.5	−17 dB at 5.5 GHz
Ref [18]	Rogers R04003	95 × 50	−11.5 dB at 5.2 GHz
Ref [19]	Fr4	30 × 16	−20.5 dB at 5.8 GHz
Proposed model	Fr4	24.5 × 15	−29.32 dB at 5.9 GHz

4 Conclusion and Future Scope

In this paper, the study and analysis of miniaturized, compact size wearable antennas for biomedical applications are discussed. The dimensions of the proposed antennas are 24.5 × 15 × 1.6 mm³ implemented on different substrates Fr4, Rogers RT/Duroid 5880 and jeans fabric. The Fr4 material and jeans fabric used are of low cost and easily available. The antennas designed and discussed can be used for various biomedical applications like patient monitoring, and other applications in sports and military for pulse monitoring etc., as it is small in size, easy to operate and can be placed in daily wear accessories like wallets and helmets. Future scope includes the implementation of fabrication and testing the antennas on the human body under different conditions.

Acknowledgements The Unique Awardee Number of corresponding author is MEITY-PHD-2245, acknowledge the Visvesvaraya PhD Scheme, DeitY, New Delhi and AUCE (A), Andhra University for providing funding and necessary support.

References

1. Balanis CA Antenna theory: analysis and design. Wiley India edition, Third Edition, pp 811–876
2. Liu ZG, Guo YX (2013) Dual band low profile antenna for body centric communications. *IEEE Trans Antenna Propag* 61(4):2282–2285
3. Hall PS et al., Antennas and propagation for on-body communication systems. *IEEE Antennas Propag Mag* 49:41–58
4. Feigin VL et al (2014) Global and regional burden of stroke during 1990–2010: findings from the global burden of disease study 2010. *Lancet* 383:245–255
5. Abtahi S, Yang J, Kidborg S (2012) New compact multiband antenna for stroke diagnosis system over 0.5–3 GHz. *Microwave Opt Technol Lett* 54(10):2342–2346
6. Persson M et al (2014) Microwave-based stroke diagnosis making global prehospital thrombolytic treatment possible. *IEEE Trans Biomed Eng* 62:2806–2817
7. Scarpello ML, Kurup D, Rogier H, Ginste DV (2011) Design of an implantable slot dipole conformal flexible antenna for biomedical applications. *IEEE Trans Antennas Propag* 59(10):3556–3564
8. Ullah S, Khan P, Ullah N, Saleem S, Higgins H, Kw KS (2009) A review of wireless body area networks for medical applications. *Int J Commun Network Syst Sci (IJCN)* 2(8):797–803
9. Sirait DC, Zulkifli FY, Rahardjo ET Basari (2013) A helical folded dipole antenna for medical implant communication applications. In: *Proceedings of IEEE MTT-S international microwave workshop series on RF and wireless technologies for biomedical and healthcare applications (IMWS-BIO)* Singapore, pp 09–11
10. Shen W, Yin WY, Member S, Sun W (2011) Compact substrate integrated waveguide (SIW) filter with defected ground structure. *IEEE Microw Wireless Compon Lett* 21(2)
11. Chen JM, Row JS (2014) Wideband circularly polarized slotted-patch antenna with a reflector. In: *Proceedings of ISAP 2014, Kaohsiung, Taiwan*, vol 1, no c, pp 615–616
12. Fields C (2016) Communications asymmetric geometry of defected ground structure for rectangular microstrip: a new approach. *IEEE Trans Antenna Propag* 64(6)
13. Dwivedi RP (2015) High gain antenna with DGS for wireless applications. In: *2nd international conference on signal processing and integrated networks (SPIN)* pp 19–24
14. Kavitha A, Swaminathan JN (2019) Design of flexible textile antenna using FR4, jeans cotton and teflon substrates. *Microsyst Technol* 25(4):1311–1320
15. Gil I, Fern R (2016) Wearable GPS patch antenna on jeans fabric. *Progress in electromagnetic research symposium (PIERS)*, Shanghai, China
16. Amsaveni A, Bharathi M, Swaminathan JN (2019) Design and performance analysis of low SAR hexagonal slot antenna using cotton substrate. *Microsyst Technol* 25(6):2273–2278
17. Chahat N, Zhadobov M, Sauleau R, Mahdjoubi K (2010) Improvement of the on-body performance of a dual-band textile antenna using an EBG structure. In: *2010 Loughbrgh antennas propagation conference LAPC 2010*, pp 465–468
18. Yu D, Liu WL, Zhang ZH (2012) Simple structure multiband patch antenna with three slots. In: *2012 international conference microwave and millimeter wave technology ICMMT 2012—Proc.*, vol 3, pp 1067–1069
19. Araghi A, Khalily M, Ghannad AA, Xiao P, Tafazolli R (2019) Compact dual band antenna for off-body-centric communications. In: *13th European conference antennas and propagation, EuCAP 2019 EuCAP*, pp 1–5

Puzzling Solid–Liquid Phase Transition of Water (mW) from Free Energy Analysis: A Molecular Dynamics Study



Chandan K. Das

Abstract Water shows a very different trend while melting. Estimating the transition point of any material is still problematic. In recent time, several techniques have been involved to simplify things. Due to anomalous behaviors of water, its transition properties are different from conventional substances. This study is an approach to understand the mechanism of phase transition using computer simulation. For better understanding the mechanism, it is vital to have knowledge of interatomic interaction of the water system. There are several potential models available for water like SPC/E, TIP3P, TIP4P, etc. Another potential model, Stillinger–Weber potential model is good enough to predict the properties of water. This model considers two- and three-particle interactions. That model of water is named as monoatomic water (mW). The Anomaly behaviors of water are well predicted using mW model of water. Difference in free energy connecting two phases of water is evaluated using reversible thermodynamic route. Supercritical path is established using more than one path. These thermodynamic paths are reversible. To the best of my knowledge, this is the first approach to apply thermodynamic path for a system where volume of solid state is more compared with volume of liquid during phase transformation. Transition point is determined Gibbs free energy. There is an abrupt change in the density as function of temperature is observed. Hysteresis loop is also observed for potential energy. For temperature higher than 285 K, huge fall in density and potential are noticed, suggesting full transition from one phase to another.

Keywords Molecular dynamics · LAMMPS · Equations of state · Thermodynamic path · Thermodynamic integration

C. K. Das (✉)

Department of Chemical Engineering, National Institute of Technology Rourkela, Rourkela 769008, India

e-mail: dasck@nitrkl.ac.in

1 Introduction

Freezing and melting can be understood by the transition and first principles, even in thermodynamic equilibrium and negotiating through a radial symmetrical pair capacity for relatively simple systems. During phase transformation, it shows anomalous behaviors compared to conventional elements and compounds. Exact mechanism of phase transformation of water becomes a challenging problem and remains unanswered [1–5].

Phase transition of water in confinement is reported [6]. Melting and freezing transition is reported based on Hansen–Verlet criteria [7]. The anomaly character of water is studied at low temperature [8]. Phase diagram is reported with influence of external force field [9]. Solidification of fluidic water is studied using TIP4P model [6, 10–12]. Thermodynamic stability is reported for mW model [13]. Transition temperature can also be evaluated using specific heat capacity information [14, 15]. Another robust technique for determination of transition point is calculation of entropy [16, 17]. Conventional methods include density hysteresis plot, Lindemann parameter [5], and non-Gaussian parameter [14].

Solid to liquid transformation of Lennard–Jones (LJ) system under confinement is reported [15]. Transition point is determined on the basis of density hysteresis plot, Lindemann parameter, and non-Gaussian parameter [15]. For determination of melting transition, change in first and second co-ordination number is important too [14].

Most of the abovementioned methods are not accurate to predict the melting transition [16]. Transition temperature of Lennard–Jones (LJ) and sodium chloride (NaCl) is reported from free energy information [17]. Free energy is evaluated employing thermodynamic integration. The thermodynamic route connecting solid–liquid is constructed employing reversible thermodynamic route [16, 17]. Phase transformation from solid to liquid under slit [18, 19], and cylindrical confinement is studied using free energy analyses [20]. Various melting and freezing criteria are reported using non-equilibrium method [21].

Free energy gap between states is estimated deploying pseudo-supercritical path [17]. Melting temperature can be predicted using Gibbs free energy. Gibbs free energy calculation involves with thermodynamics integration and multiple histogram reweighting (MHR) method [17].

In this work, I evaluate free energy gap connecting solid–liquid transitions. I present briefly the technique. (a) The liquid state is transformed into a poorly interacting liquid with the help of slowly decreasing the interatomic attractions. (b) Gaussian wells are located to the corresponding particles; simultaneously, the volume is enlarged to obtain a poorly interacting oriented state. (c) Gaussian wells are removed gradually, and simultaneously, interatomic attractions are slowly brought back to its whole strength to obtain a crystalline state.

2 Methodology

In this work, I evaluate the free energy connecting solid–liquid state transition. The inclusive technique is described elsewhere [16]. Equation of states are generated using multiple histogram reweighting technique [20]. Free energy computation is performed with the help of pseudo-supercritical transformation path. Then, ultimately with the help of second and third steps evaluation of the transition point is done at zero energy difference [20]. Each step is exclusively elaborated below. Interaction potential of water (mW) is as follows:

$$E = U_{\text{inter}}(r^N) = \sum_i \sum_{j>i} \varphi_2(r_{ij}) + \sum_i \sum_{j \neq i} \sum_{k>j} \varphi_3(r_{ij}, r_{ik}, \theta_{ijk}) \quad (1)$$

$$\varphi_2(r_{ij}) = A_{ij} \epsilon_{ij} \left[B_{ij} \left(\frac{\sigma_{ij}}{r_{ij}} \right)^{p_{ij}} - \left(\frac{\sigma_{ij}}{r_{ij}} \right)^{q_{ij}} \right] \exp\left(\frac{\sigma_{ij}}{r_{ij} - a_{ij} \sigma_{ij}} \right) \quad (2)$$

$$\begin{aligned} \varphi_3(r_{ij}, r_{ik}, \theta_{ijk}) &= \lambda_{ijk} \epsilon_{ijk} [\cos \theta_{ijk} - \cos \theta_{0ijk}]^2 \\ &\exp\left(\frac{\gamma_{ij} \sigma_{ij}}{r_{ij} - a_{ij} \sigma_{ij}} \right) \exp\left(\frac{\gamma_{ik} \sigma_{ik}}{r_{ik} - a_{ik} \sigma_{ik}} \right) \end{aligned} \quad (3)$$

The φ_2 represents two-particle interaction term. The φ_3 presents three-particle attraction expression. The summation in the expression is overall neighbors J and K of atom I within a truncated length a [22]. The A , B , p , and q parameters are employed for two-particle attractions. The λ and $\cos \theta_0$ parameters are used only for three-particle attractions. The ϵ , σ , and a parameter are employed for both cases. γ is employed for three-particle attraction. The others extra parameters are dimensionless [22].

2.1 Estimation of an Estimated Transition Temperature

To detect an approximate transition temperature, gradually heating and quenching simulations are performed for solid and liquid states, respectively [20], by employing *NPT* simulation at $P = 1.0$ bar. Around 36,864 number of particles are used in simulation. Afterward, the estimated transition temperature is chosen within the metastable region at where a sudden change in the density is noticed [20].

2.2 *Solid and Liquid Free Energy Curve with Respect to Their Corresponding Reference States*

The Gibbs energy is expressed in terms of temperature. They are presented for both states with regard to their corresponding standard phase temperature. This free energy curves are obtained over a small region around the estimated transition point at the constant pressure [20]. The temperature range is given in Eq. (16). This is carried out using multiple histogram reweighting (MHR) technique.

2.3 *Computation of Solid–Liquid Free Energy Gap at an Estimated Transition Point*

The Helmholtz free energy gap connecting the solid and liquid states at an estimated transition point is estimated by forming a reversible way connecting the solid and liquid states with the help of other reversible stages [20]. The free energy throughout the connecting route is evaluated using a known integration scheme:

$$\Delta A^{ex} = \int \frac{dU}{d\lambda}_{NVT\lambda} d\lambda \quad (4)$$

while ΔA^{ex} is the gap in Helmholtz free energy. Kirkwood's coupling parameter is used by the symbol λ . Generally, λ changes in between 0 to 1. The value of $\lambda = 0$ system acts as an ideal state [17]. The angled bracket is indication of ensemble average for a specific λ parameter [17]. The three stages pseudo-supercritical conversion method is represented in Fig. 1. Very short explanation of the stages is presented below.

2.3.1 Stage-a

Initially, strongly attracted liquid state is transformed into a poorly interacting liquid using a coupling parameter λ , which controls interatomic potential [17] in the mentioned way:

$$U_a(\lambda) = [1 - \lambda(1 - \eta)]U_{\text{inter}}(r^N) \quad (5)$$

where $U_{\text{inter}}(r^N)$ is the interatomic interaction energy due to location of all N particles [17]. The η is a scaling parameter. The value varies $0 < \eta < 1$. The first derivative of intermolecular interaction relation produces:

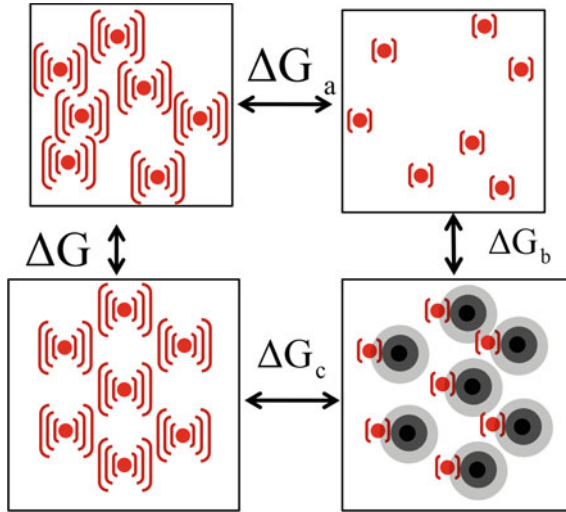


Fig. 1 Presents the three-stage pseudo-supercritical conversion route. **a** The liquid state is transformed into a poorly interacting liquid by slowly increasing the coupling parameter [17]. **b** Gaussian wells are located to the corresponding particles; simultaneously, the volume is enlarged to obtain a poorly interacting oriented state. **c** Gaussian wells are removed gradually while coupling parameter is slowly increasing to bring back its full strength to obtain a crystalline state

$$\frac{\partial U_a}{\partial \lambda} = -(1 - \eta)U_{\text{inter}}(r^N) \tag{6}$$

2.3.2 Stage-b

During second stage, volume of liquid state is enlarged to the volume of solid state unlike other conventional substances. Enlarge volume is clearly visible in Fig. 1. Hence, length of the simulation box (L_x , L_y , and L_z) for a particular system dimension must be predetermined at the estimated transition point, either from the MHR results or hysteresis diagram [17]. Liquid box dimension is 10.318570 nm, and solid phase dimension is 10.401820 nm. The interatomic interaction on the basis of λ in this stage is represented following way:

$$U_b(\lambda) = \eta U_{\text{inter}}[r^N(\lambda)] + \lambda U_{\text{Gauss}}[r^N(\lambda), r_{\text{well}}^N(\lambda)] \tag{7}$$

where $r^N(\lambda)$ and $r_{\text{well}}^N(\lambda)$ are the representation of the positions of atoms and Gaussian wells respectively [17]. U_{Gauss} presents interatomic potential because of the attraction in between the wells and corresponding particles (Eq. 9). The values of parameters ‘a’ and ‘b’ are taken from Gochola’s works [16]. Particles coordinate due

to enlarged volume from liquid to solid conversion in the same manner as did in literature [17]. Equation (8) represents change in box dimension for coupling parameter values.

$$H(\lambda) = (1 - \lambda)H_l + \lambda H_s \quad (8)$$

$$U_{\text{Gauss}}[r^N(\lambda), r_{\text{well}}^N(\lambda)] = \sum_{i=1}^N \sum_{k=1}^{N_{\text{wells}}} a_{ik} \exp[-b_{ik} r_{ik}^2(\lambda)] \quad (9)$$

$$\frac{-U_{\text{inter}}}{\delta H_{xz}} = \sum_y P_{xy}^{\text{ex}} V H_{zy}^{-1} \quad (10)$$

Derived form of potential expression with respect to λ is

$$\frac{\partial U_b}{\partial \lambda} = - \sum_{x,y,z} V(\lambda) H_{zy}^{-1}(\lambda) \Delta H_{xz} \left(\eta P_{xy}^{\text{ex}} + \lambda P_{\text{Gauss},xy}^{\text{ex}} \right) + U_{\text{Gauss}}[r^N(\lambda), r_{\text{well}}^N(\lambda)] \quad (11)$$

2.3.3 Stage-c

Stage-c is ultimate step of the pseudo-supercritical conversion method [17]. In this stage, fully interacting solid configurationally phase is obtained. The interaction potential is presented of this final step in terms of λ

$$U_c(\lambda) = [\eta + (1 - \eta)\lambda]U_{\text{inter}}(r^N) + (1 - \lambda)U_{\text{Gauss}}[r^N(\lambda), r_{\text{well}}^N(\lambda)] \quad (12)$$

And the derivative terms can be rewritten:

$$\frac{\partial U_c}{\partial \lambda} = (1 - \eta)U_{\text{inter}}(r^N) + U_{\text{Gauss}}[r^N(\lambda), r_{\text{well}}^N(\lambda)] \quad (13)$$

2.4 Determination of Transition Point Where ΔG is Zero

The free energy ΔA^{ex} , between phases at the estimated transition point is evaluated by thermodynamic integration [17]. Now transfer of the Helmholtz energy into the Gibbs energy is important. It is obtained using the formulae given, $\Delta G = \Delta A^{\text{ex}} + \Delta A^{\text{id}} + P \Delta V$. Additionally, the histogram reweighting method produces two equations of states. Expression of the liquid state, $[(\beta G)_{T_1,l} - (\beta G)_{T_i,l}]$ is familiar and for the solid state expression is $[(\beta G)_{T_1,s} - (\beta G)_{T_i,s}]$ [17], provided that T_{em} is estimated

transition temperature, at which the reversible thermodynamic path [20] is performed [17], achieved the following:

$$\begin{aligned} & [(\beta G)_{T_{1,s}} - (\beta G)_{T_{em,s}}] + [\beta(G_{T_{em,s}} - G_{T_{em,l}})] - [(\beta G)_{T_{1,l}} - (\beta G)_{T_{em,l}}] \\ & = [(\beta G)_{T_{1,s}} - (\beta G)_{T_{1,l}}] \end{aligned} \quad (14)$$

Equation 10 further can be rearranged as:

$$\begin{aligned} & [(\beta G)_{T_{1,s}} - (\beta G)_{T_{em,s}}] + [\beta(G_{T_{em,s}} - G_{T_{em,l}})] + [(\beta G)_{T_{1,l}} - (\beta G)_{T_{i,l}}] \\ & - [(\beta G)_{T_{1,l}} - (\beta G)_{T_{em,l}}] = [(\beta G)_{T_{1,s}} - (\beta G)_{T_{i,l}}] \end{aligned} \quad (15)$$

3 Simulation Details and Software Work

3.1 Molecular Dynamics Simulation

Molecular dynamics simulation is a technique used to get insights about the movements and properties of a system of atoms and molecules. It is basically a computer simulation where the system of atoms and molecules are given to make interaction among them for a certain amount of period.

3.2 Atomic Potential Used

Stillinger–Weber Potential is a good model for mW water. It considers both two-particle and three-particle interactions. The values of following parameters in metal units have been used. The potential of the water is provided in Eq. (1), (2), and (3). Parameters values are listed in Table 1.

Table 1 Value of parameters used in mW potential (in real unit)

A	B	P	Q	A	λ	γ	$\varepsilon(\text{kcal})$	$\sigma(\text{\AA})$
7.0495562	0.6022245	4	0	1.80	23.15	1.20	6.189	2.3925

3.3 Simulation Details and Potential Model

The NPT MD simulations are conducted with the help of LAMMPS [22]. Integration time step (Δt) is 5 fs. Numbers of particles are simulated around 36,684. The periodic boundary condition is applied for simulations. Cooling process is carried out gradually after each 5000,000 MD time steps. 36,864 Number of particles are used. Throughout the simulation process, applied boundary conditions for all the three dimensions are periodic. The constructions of equations of states for both phases are done employing multiple histogram reweighting diagrams. Histograms are generated from *NPT* simulations based on volume and potential energy of the system. Total 17 simulations are carried out for individual phase. The temperature is selected in accordance with the given formulae

$$T_i = T_{em} + \sum_{n=-8}^8 n\Delta T \quad (16)$$

where T_{em} is the expected transition point computed from the density versus temperature plot hysteresis data; $\Delta T(=5 \text{ }^\circ\text{K})$ is determined based on size of the metastable region.

For the reversible path evaluation (for the three steps of pseudo-supercritical path) as shown in Fig. 1, simulations are carried with *NVT* ensemble. Total run time for each simulation of three stages for each λ value is 60 ns, and time step of integration is 10 fs. Total run time for each coupling parameter value is 60 ns [20].

4 Results and Discussions

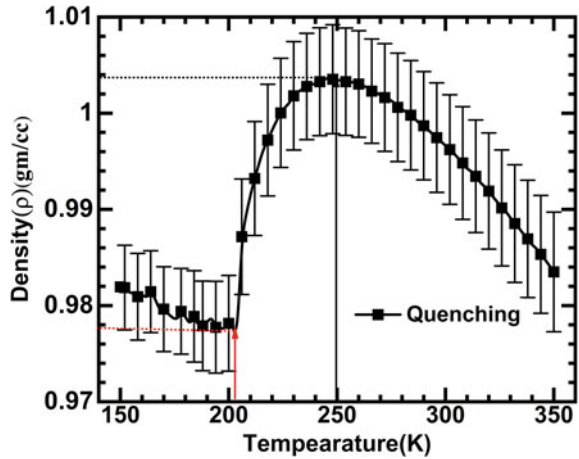
In this portion, I try to describe output results of various parameters like density, potential energy, and free energy with the change in temperature and coupling parameter.

4.1 Density

In this part, I describe the nature of density of the water system as I perform heating and quenching. Sharp density changed is observed for both the heat and quenching case. During cooling the system as shown in Fig. 2. I obtain maximum density 1.003362 gm/cc at temperature 250 K and minimum density 0.977756 gm/cc at temperature 202 K. Results are well agreement with literature value [23].

Figure 2 represents density as function of temperature. The vertical upward arrow black in color indicates maximum density (1.003362 gm/cc) and temperature (TMD

Fig. 2 Representation of density as function of temperature



= 250 K), and the vertical upward arrow red in color indicates minimum density (0.977756 gm/cc) and temperature ($T_{mD} = 202$ K).

Figure 3 represents density versus temperature plot. Filled square black in color represents the quenching system whereas filled circle for heating the system, vertical arrow line indicates an estimated estimate transition point ($T_{em} = 250$ K). Horizontal black line and red line indicate corresponding liquid density (1.003362) and solid density (0.979466), respectively, at estimated melting temperature.

Potential energy also shows the similar kind of behavior like density which is shown in Fig. 4. The red rectangle represents heating process. The black circle represents quenching process.

Fig. 3 Representation of density versus temperature plot

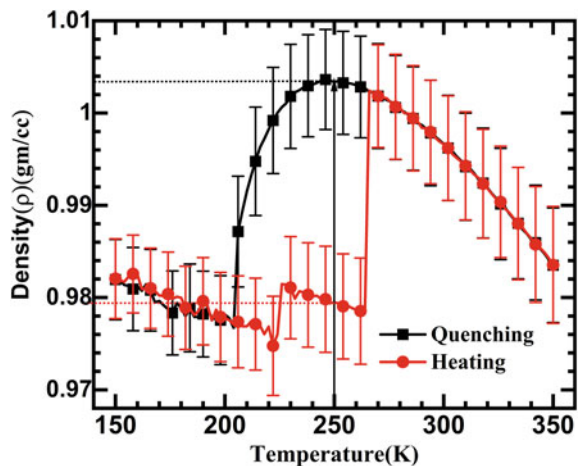
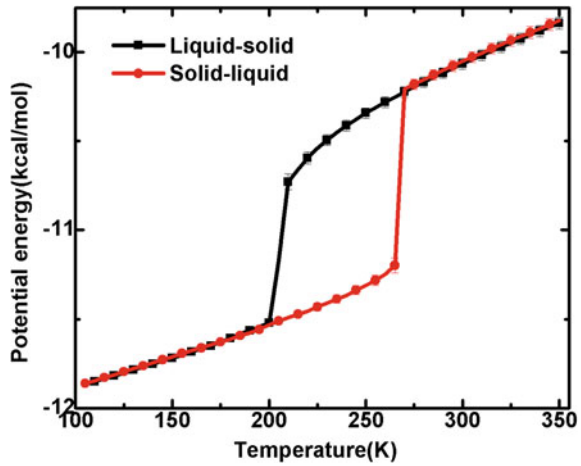


Fig. 4 Potential energy in terms of temperature similar kind of nature is obtained as density versus temperature curve. Filled square black in color represents cooling process whereas red circle represents heating process



4.2 Free Energy

Helmholtz free energy difference between liquid and solid phase is determined using pseudo-supercritical path by constructing reversible thermodynamic paths [16]. Derivative of interaction potential energy with respect to λ is presented in Fig. 5. For the λ values, they coincide as shown in Figs. 5, 7, and 8 and represent for stage-b and stage-c, respectively. The whole thermodynamic path is performed such a way that pressure remain unchanged at beginning and at the end of the path, as shown in Fig. 6. The derivative of potential energy with respect to coupling parameter in stage-b is represented in Fig. 7. The path is reversible and smooth. Similarly, the derivative of potential energy with respect to coupling parameter in stage-c is represented in

Fig. 5 $\langle \partial U_a / \partial \lambda \rangle_{NVT\lambda}$ as a variable of λ for three λ types values (10, 15, and 20) of stage-a for pseudo-supercritical path. Thermodynamic path is smooth and reversible, hence integrable

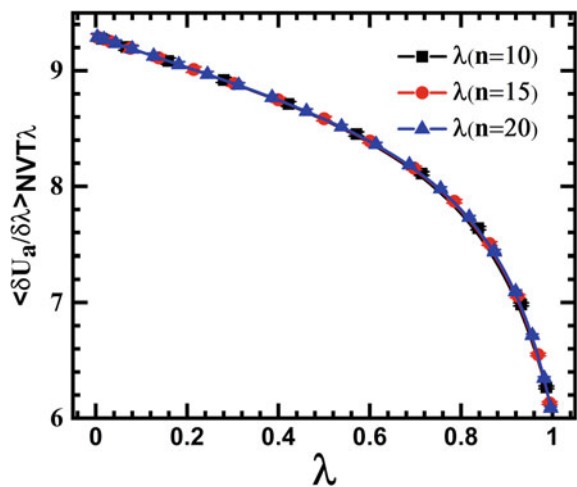


Fig. 6 Pressure at the start of stage-a and at the end of stage-c is constant. This is essential and the sufficient criteria for construction of the thermodynamic reversible paths

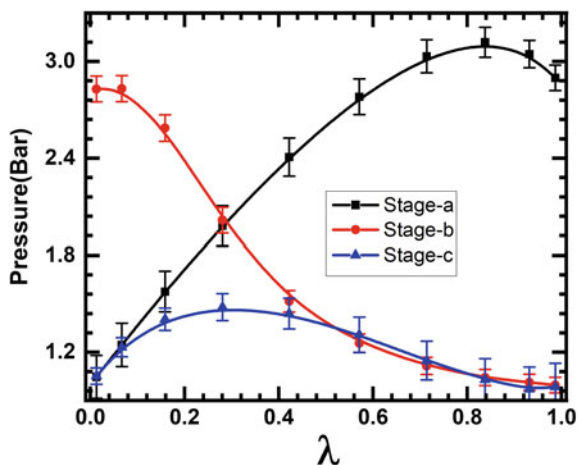


Fig. 7 $\langle \partial U_b / \partial \lambda \rangle_{NVT\lambda}$ as a variable of λ (10, 15, and 20) of stage-b values. Thermodynamic path is smooth and reversible, hence integrable. Error is so small and it submerges with symbol

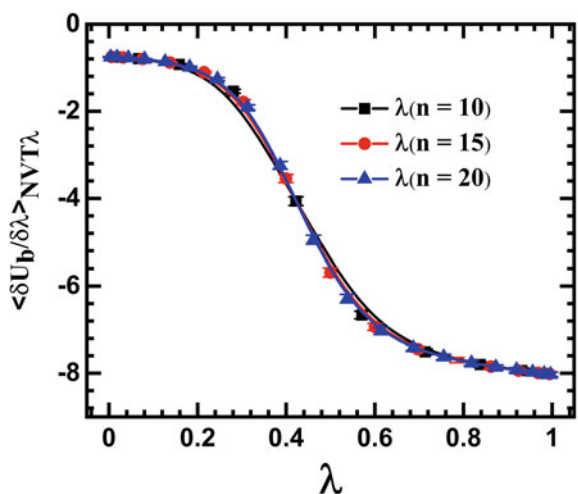


Fig. 8. Figures for all the stages are smooth and reversible, so we can easily integrate it. The Gibbs free energy difference connecting solid–liquid is around 0.101025 ± 0.00135 kcal/mol. Results are reported in Table 2. Using the Gibbs free energy difference between solid–liquid obtained from pseudo-supercritical path, along with equation of state which is obtained from multiple histogram reweighting (MHR) method, the equation of states are converted into single reference state using Eqs. (14) and (15). The gap in free energy connecting two phases is determined and presented in Fig. 9. True thermodynamic melting temperature is the point where ΔG is zero. From Fig. 9, it is clear that true thermodynamic transition point is around 273.9 ± 0.9 °K.

Fig. 8 $\langle \partial U_c / \partial \lambda \rangle_{NVT\lambda}$ as a variable of λ (10, 15, and 20) for stage-c

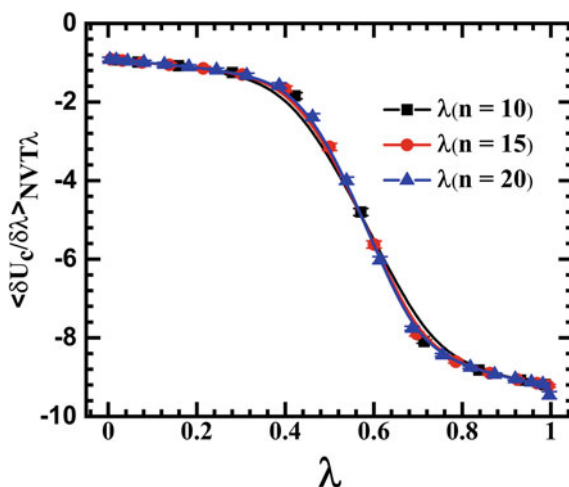
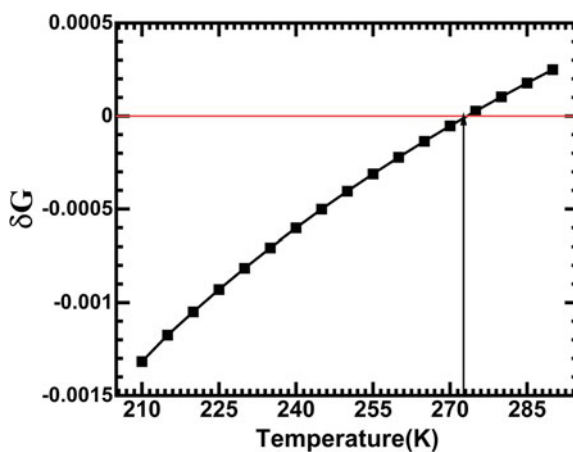


Table 2 Separation of the subscriptions to the gap in Gibbs free energy connecting the two states $T = 250$ °K. The pressure is maintained at $P = 1$ Bar

Free energy terms (kcal/mol)	
$A_s^{ex} - A_l^{ex}$	11.989193 ± 0.00135
$A_s^{id} - A_l^{id}$	-11.931945
$P\Delta V$	0.043769
$G_s - G_l$	0.101025 ± 0.00135

Fig. 9 ΔG as a function of T . Vertical arrow line blue in color indicates solid–liquid transition point temperature or true thermodynamic transition temperature (T_m) of solid where Gibbs free energy difference, ΔG , between solid and liquid is zero



5 Conclusion

Various methods have been employed, and I have been successful in observing the phase transformation of water, depending on various parameters. While simulating with decreasing temperature, this is the value for density after melting occurred at approximately 273.9 ± 0.9 °K, which is slightly higher than the reported melting point for water which is 273.15 K. Phase transition point is determined basing on Gibbs free energy. Estimation of Gibbs free energy is performed with the help of pseudo-supercritical reversible thermodynamic cycle along with the help of multiple histogram reweighting diagrams. The construction of supercritical path is the combination of three stages. The thermodynamic integration is applied using 10, 15, and 20 points. Estimated true thermodynamics melting temperature is around 273.9 ± 0.9 °k, which is in good precision with experimental results.

Acknowledgements This work is supported by National Institute of Technology Rourkela, Government of India.

References

1. Cahn RW (2001) Melting from within. *Nature* 413:582–583
2. Das CK, Singh JK (2013) Effect of confinement on the solid-liquid coexistence of Lennard-Jones fluid. *J Chem Phys* 139(17):174706
3. Das CK, Singh JK (2013) Melting transition of confined Lennard-Jones solids in slit pores. *Theor Chem Acc* 132:1351
4. Das CK, Singh JK (2013) Melting transition of confined Lennard-Jones solids in slit pores. *Theor Chem Acc* 132(4):1351-1-1351–13
5. Das CK, Singh JK (2014) Melting transition of Lennard-Jones fluid in cylindrical pores. *J Chem Phys* 140(20):204703-1-204703–9
6. Eike DM, Brennecke JF, Maginn EJ (2005) Toward a robust and general molecular simulation method for computing solid-liquid coexistence. *J Chem Phys* 122:014115
7. Grochola G (2004) Constrained fluid λ -integration: constructing a reversible thermodynamic path between the solid and liquid state. *J Chem Phys* 120:2122
8. Hansen J-P, Verlet L (1969) Phase transitions of the Lennard-Jones system. *Phys Rev* 184(1):151–161
9. Hoffmann GP, Löwen H (2001) Freezing and melting criteria in non-equilibrium. *J Phys Condens Matter* 13:9197–9206
10. Jacobson LC, Hujo W, Molinero V (2009) Thermodynamic stability and growth of guest-free clathrate hydrates: a low-density crystal phase of water. *J Phys Chem B* 113:10298–10307
11. James T, Wales DJ (2007) Energy landscapes for water clusters in a uniform electric field. *J Chem Phys* 126:054506-1-054506–12
12. Moore EB, Molinero V (2009) Growing correlation length in supercooled water. *J Chem Phys* 130(24):2445505-1-2445505–12
13. Moore EB, Molinero V (2010) Ice crystallization in water's "no-man's land." *J Chem Phys* 132(24):244504-1-244504–10
14. Moore EB, Molinero V (2011) Is it cubic? Ice crystallization from deeply supercooled water. *Phys Chem Chem Phys* 13:20008–20016

15. Moore EB, Molinero V (2011) Structural transformation in supercooled water controls the crystallization rate of ice. *Nature* 479:506–509
16. Moore EB et al (2010) Freezing, melting and structure of ice in a hydrophilic nanopore. *Phys Chem Chem Phys* 12:4124–4134
17. Plimpton S (1995) Fast parallel algorithms for short-range molecular dynamics. *J Comp Phys* 117:1–9
18. Stanley HE et al (2000) The puzzling behavior of water at very low temperature invited lecture. *Phys Chem Chem Phys* 2:1551–1558
19. Stillinger FH (1995) A topographic view of supercooled liquids and glass formation. *Science* 267(5206):1935–1939
20. Stillinger FH, Weber TA (1980) Lindemann melting criterion and the Gaussian core model. *Phys Rev B* 22(8):3790–3794
21. Stillinger FH, Weber TA (1984) Point defects in bcc crystals: structures, transition kinetics, and melting implications. *J Chem Phys* 81(11):5095–5102
22. Stillinger FH, Weber TA (1985) Computer simulation of local order in condensed phases of silicon. *Phys Rev B* 31(8):5262–5271
23. Zhang SL et al (2011) The study of melting stage of bulk silicon using molecular dynamics simulation. *Physica B* 406:2637–2641

Social Media Flood Image Classification Using Transfer Learning with EfficientNet Variants



S. M. Jaisakthi and P. R. Dhanya

Abstract Social media posts consist of a large number of flood-related images and data. These images can be helpful for the flood image classification process and necessary to produce real-time information about road accessibility during rescue and disaster management. In this paper, social media images are classified into flooded and non-flooded categories using EfficientNet variants with MediaEval 2017 Disaster Image Retrieval from the social media dataset. The transfer learning technique is employed to achieve good performance and efficiency. Then different variants are compared using various metrics after and before freezing. The higher variants are intentionally avoided, as it does not perform with better efficiency and accuracy. The B3 variant outperforms all other variance when considering model validation accuracy and false positive rate after unfreezing top layers. But when analysing the classification report, all variants show almost equal performance and weighted average $F1$ -score.

Keywords EfficientNet · Flood · Social media

1 Introduction

Floods are the most common natural calamity which can create devastating effects on properties and lives. There is a considerable increase in the alarming rate of floods due to heavy rainfall, urbanization and population. In this context, it is necessary to produce accurate and timely information about flood assessments before, during and after an event. The information about road conditions and accessibility is crucial for fast responding and rescue operations. The domination of social media is considered for getting fruitful information about flood disasters. Popular social media such

S. M. Jaisakthi (✉) · P. R. Dhanya
School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, India
e-mail: jaisakthi.murugaiyan@vit.ac.in

P. R. Dhanya
e-mail: dhanya.pr@vit.ac.in

as Twitter, Facebook, YouTube and other microblogs provide potential information regarding flooding events. The social media posts contain both visual and text information about an event. This enormous amount of data may or may not directly give information about flood events. In this paper, an efficient algorithm is proposed to classify flooded and non-flooded social media images. Both visual and text features are considered for flood image classification.

Flood images uploaded to social media platforms are becoming a vital source of information about flooding events. It is possible to employ two different approaches for flood detection using social media images. The visual feature extraction approach exploits convolutional neural networks (CNN) architectures. The visual feature is fused with text features for flood event classification [1]. It is possible to identify images that provide evidence for social passability. Satellite images and social media images have to be differentiated to detect passable and non-passable roads. Deep learning models pre-trained with ImageNet and Places dataset are used for object and scene level extraction. CNN [2] and the transfer learning-based classification approach are used to determine whether a vehicle can pass the road [3]. An essential need during floods is accessing open roads for transporting emergency support to the people. A set of tweets posted during floods can be utilized for studying the evidence of road passability using a single double-ended neural network [4]. It is very difficult to move drainage water during heavy rain in several countries. It is possible to detect a flooding event using the method of image segmentation using SegNet. There comes a necessity to take immediate action to reduce the impact of floods [5]. FloodBot is a real-time flood detection and notification system deployed in a flash flood-prone Ellicott City, Maryland. This FloodBot is working with the method of semantic segmentation using U-Net architecture with above 80% of accuracy. The scene level information is augmented with the readings of actual sensor data and weather data [6]. It is preferred to use multimodal data on incident detection for better accuracy of prediction. Multimodal data from social media posts are used to detect emergency incidents using CNN and long short-term memory (LSTM). VGG16 outperforms AlexNet [7], VGG19 and SqueezeNet with better accuracy [8, 9].

The method for summarization of images from social media is detecting images to cluster the images based on features. Each cluster wrapped with textual information gives details about a flooding event. This information can be useful for rescue authorities and smartphone users [10]. A CNN-based system consists of two levels of identification scheme for flood detection using CNN followed by colour-based filtering to obtain the final detection. The system has been tested with recent flood image posts from Twitter, and results are encouraging [11]. Smart communication devices are becoming responsible for sharing multimedia content on social media. In the 2015 Tamil Nadu floods, there was a massive sharing of flood information through social media. It is possible to analyse the severity of floods using Naive Bayesian and SSVM classification. A disaster geographic map is generated based on location to interpolation cluster proximity considering flood-affected areas [12]. A fully automated real-time system is implemented for water level estimation from flood images in social media using deep learning. This method may overcome the label scarcity problem for many regression tasks. In this method, human annotators

can decide which image has a higher level of water compared to the two images [13]. Most of the recent research articles have discussed the automated machine learning approach to extract the features from social media images to get information about flood events. It has been useful for early warning detection, damage assessments, disaster coordination and timely response [14]. Semantic drift demonstrates gradual and long-term changes in meaning and sentiments studying corpora. This semantic drift is used to detect floods and helps to increase the reference data for event monitoring. The deep learning methods are employed to determine whether social media images are associated with semantically drifted social media tags [15].

Social media contains a sufficient amount of posts that may or may not be directly related to flooding events. So in this paper, a novel classification algorithm is used to classify social media posts into two categories; images are related to flooding events and are not related to flooding events. The EfficientNet [16] variants are considered for classifying both visual and text data. The textual features are extracted from metadata that contains information about post, date, time, contents, location, sentiment and emotion, etc. These two features are trained together and classify images into flooded and non-flooded categories.

The structure of the paper is as follows: Sect. 2 discusses proposed methods including datasets used for classification and methodology. Experimental results and evaluations of the proposed classification algorithm are given in Sect. 3. Finally, conclusions and future works are presented in Sect. 4.

2 Proposed Method

2.1 Dataset

MediaEval 2017 Disaster Image Retrieval from social media [17] is used for the classification of flooded and non-flooded images in social media posts. This dataset contains a total number of 6600 Flickr images extracted from the YFC100M dataset and shares a common licence. These images are divided into development sets and test sets. Development set consists of 5280 images with features and class labels while the test set consists of 1320 images with features and class labels. Class vectors 0 represents evidence of flooding and 1 represents no evidence of flooding. Test image folder contains an inconsistent number of images and labels. So in this paper, the training dataset will be split into training data and validation data.

2.2 Methodology

EfficientNet is one of the most efficient image classification models. The EfficientNet provides different variants (B0 to B7) that have different scales of efficiency and

accuracy. The dataset is initially split into training set and validation set with a test size ratio of 0.33 and random state value 54. Test size represents the proportion of the dataset included in the test split. The random state controls the shuffling applied to the data before applying the split function. The input shape taken by different variants also varies. Hence, it becomes necessary to preprocess the input images that are fed to the network. It is necessary to resize images to the same size when the dataset includes different sizes of images. The input images are reshaped according to the variant that has been chosen. All the images are augmented after the reshaping process using Keras preprocessing layers API to increase the diversity of the dataset and to overcome the overfitting problems. The model efficiency increases with the availability of images in the dataset. Hence, data augmentation techniques are used to create new data for increasing the volume of the dataset from existing training data. The images are randomly rotated, translated, flipped and contrast adjusted during the augmentation process. This paper uses transfer learning techniques by initializing EfficientNet with pre-trained ImageNet weights and using them to fine-tune the dataset. Transfer learning is a machine learning method, where the model is reused as the entry point for modelling in the second task. Transfer learning can be considered as an optimization process as it increases the performance in the second task. The detailed representation of the proposed method is shown in Fig. 1.

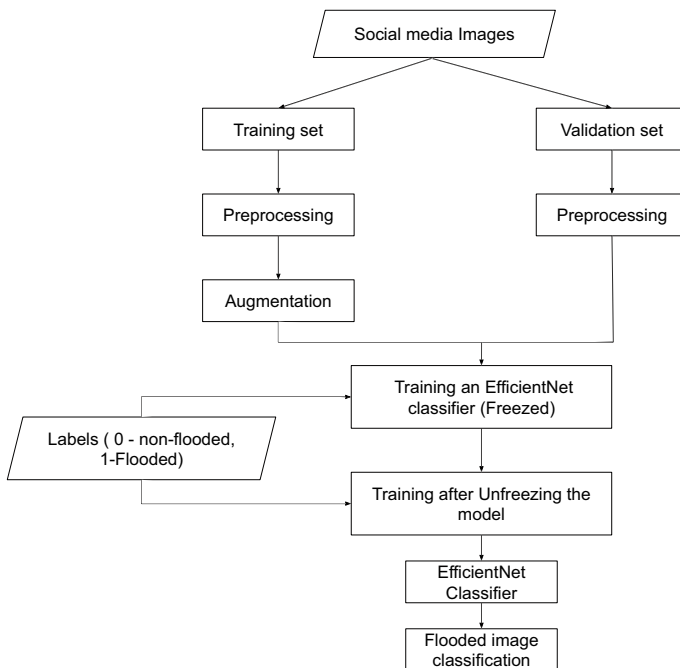


Fig. 1 Block diagram of proposed method

The EfficientNet uses a scaling method that uniformly scales all dimensions of depth, width and resolution using a compound scaling coefficient ϕ in a principled way. So it is possible to increase the network depth by α^n , width by β^n , and image size by γ^n for using 2^n times more computational resources. α , β and γ are the constant coefficients determined by a small grid search on the original small model. The compound scaling method is defined as

$$d = \alpha^\phi, w = \beta^\phi, r = \gamma^\phi, \ni \alpha \cdot \beta^2 \cdot \gamma^2 \approx 2 \quad (1)$$

where α , β , γ and ϕ are depth, width, resolution and user-specified coefficient, respectively. Initially, all layers are frozen, and then only the top layers are trained. The model is compiled with Adam optimizer with a learning rate of 1e-3 and mean-squared logarithmic error. The model is trained for 20 epochs. The validation accuracy has no further improvement by increasing epochs. Secondly, the model is trained by unfreezing some layers with a preferably small learning rate. The unfrozen model is trained with the same optimizers with a learning rate of 1e-4 and the previous loss function. It deals with binary classification in this paper. The BatchNormalization layers must be kept frozen because it uses non-trainable weights to keep track of the mean and variance during training. Otherwise, these updates on non-trainable weights will destroy the learning of the model. Larger variants of EfficientNet do not promise improved efficiency and accuracy especially, for tasks with fewer data and fewer classes. Hence, it is only concentrated on B0 to B3 variants. The B0, B1, B2 and B3 variants take resolutions of 224, 240, 260 and 300, respectively, and input data should range [0, 255]. Normalization is included as part of the EfficientNet model. The ground truth contains class vectors 0 and 1 which is converted into a binary class matrix. The results are produced considering posterior probabilities of an image being flooded or non-flooded.

3 Experimental Results and Evaluation

The accuracy of a model is a suitable metric for both binary and multiclass classification problems. So it will be good to use training and validation accuracy and loss to study the efficiency of our classification model. The increase in validation accuracy is more entertaining than training accuracy. When considering classification problems, it will be using classification metrics such as true positive rate, false positive rate, $F1$ -score, precision, recall and support to evaluate it. It is expected to be the false positive rate to be minimum, ideally less than 10%. A classification report is the summarization of main metrics per class for classification problems. The metrics are given in terms of true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN). The true positives are when the actual class is either 'flooded' or 1 as is the predicted class. The false positives are when the actual class is either 'non-flooded' or 0 but the predicted class is either 'flooded' or 1.

Precision is a metric that shows the exactness of a classifying model. It is the ratio between true positive and the sum of true positives and false positives. The recall shows the completeness of the classifier. It is defined as the ratio of true positives to the sum of true positives and false negatives. The $F1$ -score is defined as the weighted harmonic mean of precision and recall. It is considered as 1 is the best score and 0 is the worst score. The support is the number of actual occurrences of the classes in the specified dataset. It will not change between models but instead diagnoses the evaluation process.

ROC curves and precision–recall curves are omitted because the dataset does not contain an equal number of data for each class, and there is no moderate or class imbalance. The variant B3 outperforms all other models considering training and validation accuracy and loss. The training and validation metrics are compared before and after freezing top layers as shown in Table 1. The graphical representation of these metrics shows the changes per epoch as shown (see Figs. 2 and 3). It is possible to undoubtedly say that there is a significant improvement in efficiency and performance using transfer learning. In a binary classification problem, the false positive rate measures the percentage of false positives against all positive predictions.

The false positive rate is minimum for the B3 variant considering all other variants after unfreezing the top layers of the model. False positive decreases for each model after transfer learning as shown in Table 2. The EfficientNet variants are not showing significant improvement in classification accuracy after transfer learning. The confusion matrix for each variants for different variants is shown (see Figs. 4 and 5).

The freed and unfreed model shows almost the same values for precision, recall and $F1$ -score as shown in Table 3. So it is clear that transfer learning improves the performance and efficiency of EfficientNet variants in social media image classification. The proposed system outperforms [18, 19] which have $F1$ -score of 35% and 51%, respectively. It uses a basic SVM classifier and CNN network for image classification, respectively. The CNN architecture is used with pre-trained models DenseNet2021, InceptionV3 and InceptionResnetV2 as basic feature extractors. In

Table 1 Accuracy and loss of freed and unfreed EfficientNet variants for training and validation

	Variants	Training		Validation	
		Accuracy (%)	Loss (%)	Accuracy (%)	Loss (%)
Freed	B0	64.66	11.19	61.33	12.37
	B1	62.34	11.10	55.54	12.59
	B2	61.29	11.27	54.96	12.46
	B3	63.22	10.87	58.84	12.21
Unfreed	B0	77.73	7.47	80.15	15.92
	B1	70.60	9.25	80.44	14.19
	B2	71.42	9.05	84.18	14.92
	B3	71.16	9.23	88.41	15.25

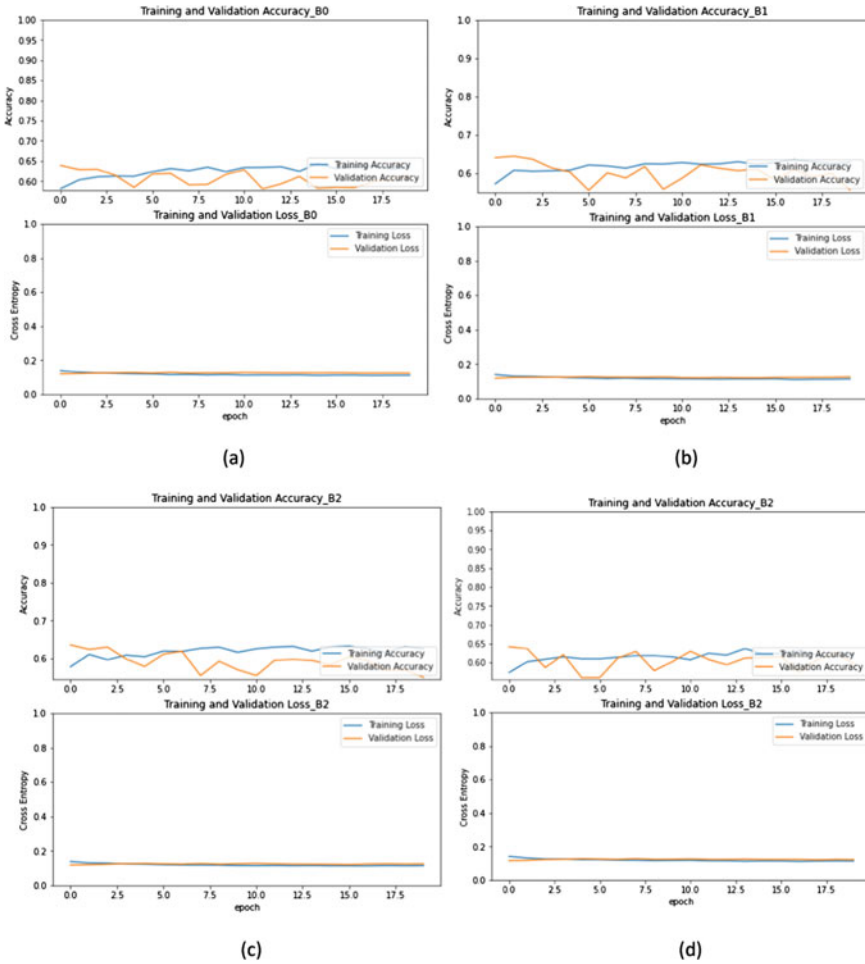


Fig. 2 Accuracy and loss for training and validation of freezed models a B0, b B1, c B2, d B3

the proposed method, the $F1$ -score for freezed and unfreezed variants has been almost the same. It is also evident that the false positivity rate for B0 increases after transfer learning compared to other variants. Even though considering low false positivity rate and high accuracy rate of variant B3, it can be selected as the preferable model for social media flood image classification. There is an adequate improvement in training and validation accuracy and a decrease in loss after transfer learning.

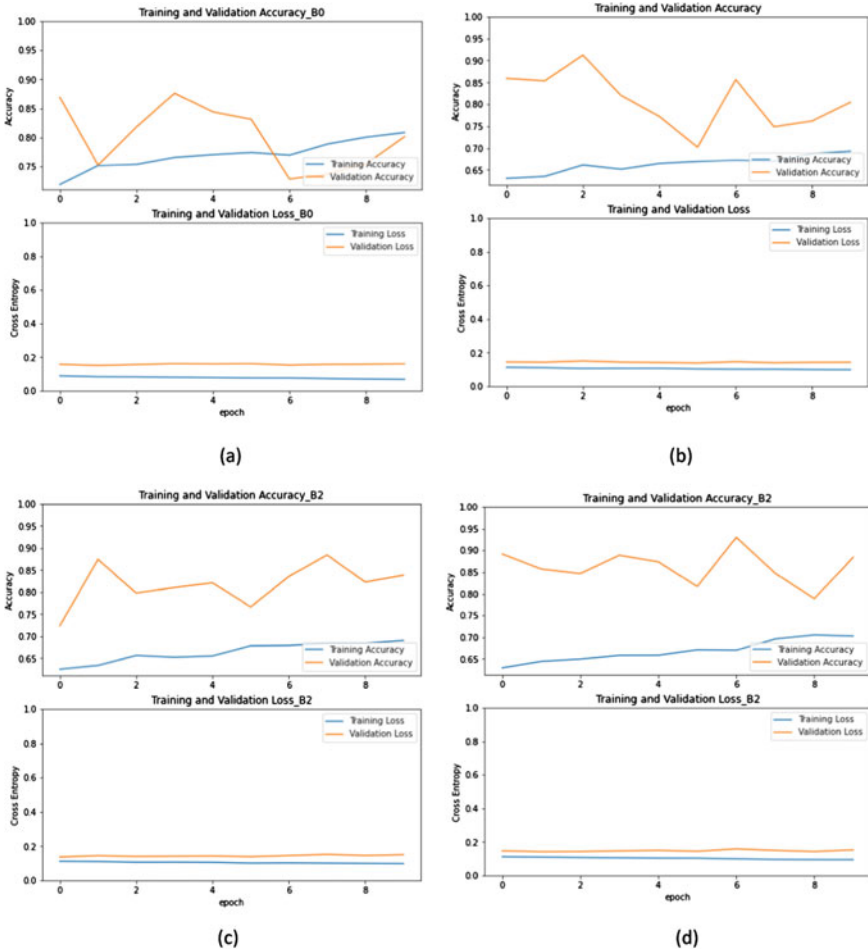


Fig. 3 Accuracy and loss for training and validation of unfreezed models **a** B0, **b** B1, **c** B2, **d** B3

Table 2 Classification accuracy, FP rate and FN rate of freezed and unfreezed EfficientNet variants

	Variants	Accuracy (%)	FP rate (%)	FN rate (%)
Freezed	B0	61	8.40	92.8
	B1	55.54	29.96	70.40
	B2	54.96	30.14	71.68
	B3	59.84	15.74	83.84
Unfreezed	B0	58	20.03	80.48
	B1	58	19.85	80.96
	B2	59	16.36	84.32
	B3	62	10.55	86.56

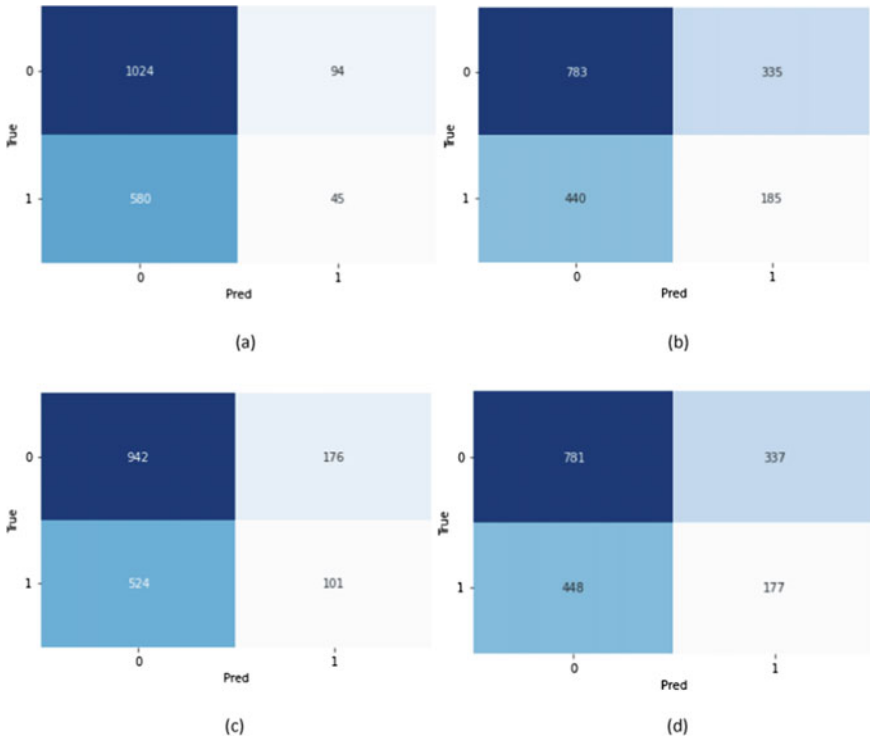


Fig. 4 Confusion matrices for frozeed **a** B0, **b** B1, **c** B2 and **d** B3

4 Conclusion

The EfficientNet variants are appropriate models for social media flood image classification. It improves its performance with transfer learning. EfficientNet variant B3 shows better performance compared to other variants. In this paper, the images and labels are used for image classification. The dataset is providing information on metadata and image features. It will be better to train the model with these features and will improve the classification accuracy and performance.

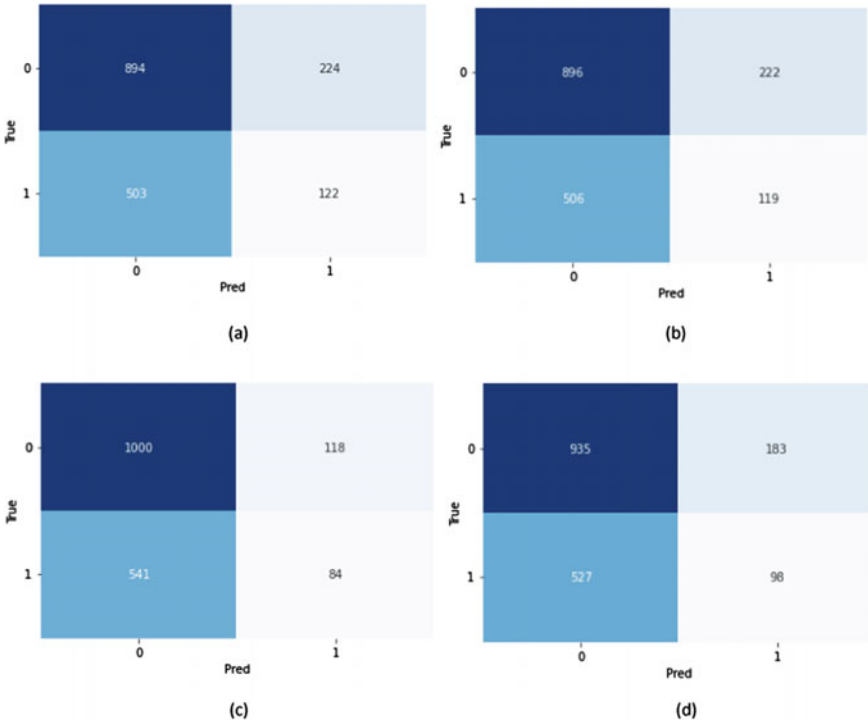


Fig. 5 Confusion matrices for unfreezed **a** B0, **b** B1, **c** B2 and **d** B3

Table 3 Weighted average metrics of freezed and unfreezed EfficientNet variants

	Variants	Precision (%)	Recall (%)	F1-score(%)	Support (%)
Freezed	B0	53	61	52	1743
	B1	54	56	54	1743
	B2	53	55	54	1743
	B3	64	60	55	1743
Unfreezed	B0	54	58	55	1743
	B1	54	58	54	1743
	B2	54	59	54	1743
	B3	67	62	56	1743

Acknowledgements The authors express their sincere thanks to the managements of Department of Science and Technology, Science and Engineering Re-search Board, Government of India (DST-SERB), and Vellore Institute of Technology, Vellore, India, for funding the respective research laboratories where this research was carried out.

References

1. Jony RI, Woodley A, Perrin D (2019) Flood detection in social media images using visual features and metadata. In: 2019 Digital image computing: techniques and applications (DICTA), pp 1–8. <https://doi.org/10.1109/DICTA47822.2019.8946007>
2. Albawi S, Mohammed TA, Al-Zawi S (2017) Understanding of a convolutional neural network. In: 2017 International conference on engineering and technology (ICET) pp 1–6. <https://doi.org/10.1109/ICEngTechnol.2017.8308186>
3. Ahmad K, Pogorelov K, Riegler M, Ostroukhova O, Halvorsen P, Conci N, Dahyot R (2019) Automatic detection of passable roads after floods in remote sensed and social media data. *Sign Process: Image Commun* 74:110–118
4. Lopez-Fuentes L, Farasin A, Zaffaroni M, Skinnemoen H, Garza P (2020) Deep learning models for road passability detection during flood events using social media data. *Appl Sci* 10:8783. <https://doi.org/10.3390/app10248783>
5. Baydargil HB, Serdaroglu S, Park J, Park K, Shin H (2018) Flood detection and control using deep convolutional encoder-decoder architecture. In: 2018 International conference on information and communication technology robotics (ICT-ROBOT), pp 1–3. <https://doi.org/10.1109/ICT-ROBOT.2018.8549916>
6. Basnyat B, Roy N, Gangopadhyay A (2021) Flood detection using semantic segmentation and multimodal data fusion. In: 2021 IEEE international conference on pervasive computing and communications workshops and other affiliated events (PerCom Workshops), pp 135–140. <https://doi.org/10.1109/PerComWorkshops51409.2021.9430985>
7. Almisreb AA, Jamil N, Din NM (2018) Utilizing AlexNet deep transfer learning for ear recognition. In: 2018 Fourth international conference on information retrieval and knowledge management (CAMP), pp 1–5. <https://doi.org/10.1109/INFRKM.2018.8464769>
8. Fathichah C, Sammy Wiyadi PD, Adni Navastara D, Suciati N, Munif A (2020) Incident detection based on multimodal data from social media using deep learning methods. In: 2020 International conference on ICT for smart society (ICISS), pp 1–6. <https://doi.org/10.1109/ICISS50791.2020.9307555>
9. Swasono DI, Tjandrasa H, Fathichah C (2019) Classification of tobacco leaf pests using VGG16 transfer learning. In: 2019 12th international conference on information & communication technology and system (ICTS), pp 176–181. <https://doi.org/10.1109/ICTS.2019.8850946>
10. Layek AK, Pal A, Saha R, Mandal S (2018) DETSApp: an app for disaster event tweets summarization using images posted on twitter. In: 2018 Fifth international conference on emerging applications of information technology (EAIT), pp 1–4. <https://doi.org/10.1109/EAIT.2018.8470398>
11. Layek AK, Poddar S, Mandal S (2019) Detection of flood images posted on online social media for disaster response. In: 2019 Second international conference on advanced computational and communication paradigms (ICACCP), pp 1–6. <https://doi.org/10.1109/ICACCP.2019.8882877>
12. Anbalagan B, Valliyammai C (2016) ChennaiFloods: leveraging human and machine learning for crisis mapping during disasters using social media. In: 2016 IEEE 23rd international conference on high performance computing workshops (HiPCW), pp 50–59. <https://doi.org/10.1109/HiPCW.2016.016>
13. Chaudhary P, D'Aronco S, Leitao JP, Schindler K, Wegner JD (2020) Water level prediction from social media images with a multi-task ranking approach. *ISPRS J Photogram Remote Sens* 167:252–262

14. Dwarakanath L, Kamsin A, Rasheed RA, Anandhan A, Huib L (2021) Automated machine learning approaches for emergency response and coordination via social media in the aftermath of a disaster: a review 9:68917–68931. <https://doi.org/10.1109/ACCESS.2021.3074819>
15. Tkachenko N, Procter R, Jarvis S (2021) Quantifying people's experience during flood events with implications for hazard risk communication. *PLoS One* 16(1):e0244801. <https://doi.org/10.1371/journal.pone.0244801>
16. Zhang D, Liu Z, Shi X, (2020) Transfer learning on EfficientNet for remote sensing image classification. In: 5th International conference on mechanical, control and computer engineering (ICMCCE), pp 2255–2258. Harbin, China. <https://doi.org/10.1109/ICMCCE51767.2020.00489>
17. Bischke B, Helber P, Schulze C, Srinivasan V, Dengel A, Borth D (2017) The multimedia satellite task at MediaEval 2017 Emergency Response for Flooding Events, MediaEval'17. Dublin, Ireland
18. Kirchknof A, Slijepcevic D, Zeppelzauer M, Seidl M (2018) Detection of road passability from social media and satellite images, MediaEval'18, vol 2283. Sophia, Antipolic, France
19. Feng Y, Shebotnov S, Brenner C, Sester M (2018) Ensembled convolutional neural network models for retrieving flood relevant tweets. In: Image, vol 10, p 1. Sophia, Antipolic, France

A Survey-Based Study to Understand Various Aspects of Kanban



Anupam Kumar, Nilesh Kumar, Sayani Mondal, and Tarun Biswas

Abstract Kanban is a Japanese coined system for workflow through cards, and the method is used in Kanban software for a real-time flow of work with optimum transparency implementing Agile and DevOps; it helps to surface the message good or bad about scope and complexity of any project. The key highlight about Kanban is it motivates the team members at the bottom level by empowering each member with an opportunity to self-commitment which in turn sets the right expectation with the stakeholders. In this paper, the survey has been done worldwide considering professional industrialists, and based on those critical conclusions have been drawn. The survey was conducted in the software industry across the globe who is extensively working on Kanban. The statistical and analytical approach has concluded that time is the best factor for Kanban, and it plays a vital role in achieving the delivery date and minimization of cycle time. A comparative study has been performed for running factors like time, scope, cost, risk, quality, and team. Quality of teamwork and time aspects after the implementation of Kanban have been explored. Jira has proven to be the best and widely used tool in terms of productivity, effectiveness, communication, and metrics. A study on reported benefits and challenges of Kanban has been identified, and the nature of the project best suited for Kanban has been investigated. Lack of experience with Kanban board and progress tracking is considered as the biggest challenges in implementing Kanban.

Keywords Kanban · Agile · DevOps · Project · Survey · Industry

A. Kumar (✉) · T. Biswas
Department of CSE, National Institute of Technology, Sikkim, India
e-mail: anupam10accent@gmail.com

N. Kumar
Energy Institute, Centre of Rajiv Gandhi Institute of Petroleum Technology, Bengaluru,
Karnataka, India

S. Mondal
Indian Institute of Technology, Kharagpur, West Bengal, India

1 Introduction

Software Engineering is a branch of engineering that deals with the design, development, and maintenance of software. It was introduced to address the issues of low-quality project software. Problems occur when software exceeds budgets, timelines, and to compensate that the quality of the software gets reduced. Software engineering ensures that the product is built consistently, accurately, correctly, on budget, on time, and within requirements. The waterfall model is one of the first and linear sequential life cycle models [1] that are based on the factory assembly-line process introduced in the year 1970. It uses rigid phases; where when one phase completes, the next phase begins. It follows sequential steps and the developers do not have the option to go from an unmodified waterfall model to previous steps. The steps involved are Requirement Specification, Design, Development, Testing, and Maintenance. The rigidity of the waterfall model restricts flexibility, and there is no scope of user feedback until software development has been finished. Agile methodology [2] evolved in the mid-1990s to overcome the traditional waterfall issues. Agile methodology is known for its flexibility, and it follows an incremental approach. Project development requirement changes are allowed in Agile whereas waterfall has no scope to change accordingly once the development of the project has started. Kanban is one of the methodologies that is considered to be Agile [3].

The concept of Kanban in Software Engineering was first introduced in the year 2010 by Anderson [4], and later with its increase in popularity, people in the software industry have started using Kanban. Initially, the demand was less as people were not much aware of Kanban and they do not know how it will impact their software development and maintenance projects. Earlier software industry people used the old traditional waterfall model in their projects that consist of 5 or 7 sequential stages in the waterfall model. If any of the sequential structures leads to problems, it would lead the development of the project to start from the beginning.

In this paper, survey research analysis has been done to investigate the overall success and satisfaction level of Kanban users and to find the best and widely used Kanban tools which are most suited for software projects. To evaluate the running factors—performance, deadline, cost, quality, team, and risk, using Likert scale for factor response has been taken into consideration, to identify the nature of the project that is most preferred for Kanban, and to identify features of Kanban benefit, challenge, and its attraction in the software industry.

2 Kanban as Project Development Software in Industries

Kanban is solely dedicated to finishing the work item that is actively in progress, the next item is picked up from the top of the backlog which is assisted by its Planning flexibility approach [5]. The product owner has the freedom to reprioritize the work as per their need without disrupting the Kanban team.

This is possible because any change outside the running work will not impact the team working on any other task. As the product owner arranges the work item on the backlog, the development team is confident that they are giving the best value to the business and the client. Kanban is based on the pull approach instead of the push approach [6]. Whenever there is demand and the ability to handle the demand, then only the assigned team pulls the work. Kanban system works on the principle of a pull-based approach. ‘In Progress (IP)’ and ‘Done’ column is used in every process state. Once the Code Review task is over, it moves to Code Review (Done) column. The done column is used as a Queue state. No work is performed in the Queue states. The testing pulls the work from Code Review (Done) column when the capacity to handle the demand. Pull-based approach reduces the piling of the work and wait times. Resource balancing and quality are maintained with the pull-based approach [7].

Cycle time is optimized so that team can forecast the future work delivery [8]. A bottleneck is identified and resolved collectively by the team to minimize cycle time. It gives the concept of continuous delivery of products or software to customers in a short release cycle [9]. Charts play a vital role for a team to improve team efficiency and effectiveness. Control charts and cumulative flow diagrams are the two reports that a Kanban team uses in their project. It is easier to see the bottlenecks in any industrial project by seeing the data [10]. Efficiency has increased because of continuous delivery, WIP, pull-based approach, and visualizing the workflow with the help of the Kanban board [11].

2.1 Challenges of Kanban

The progress monitoring task at the time of the development process of software projects is the major challenge of Kanban. Software practitioners face challenges in determining the WIP limits for each stage of the Kanban board because of the manpower and skill issues. This further leads to late delivery and project failures in certain cases [12]. Challenges like lack of experience with Kanban visualization board, team members, motivation to use old methods, and not being willing to follow Kanban.

The goal in the software industry is to complete a project with proper tracking and management of complete work more efficiently and successfully. The Literature study as per Table 1 reveals that there has not been a prior study on the most widely used available tools of Kanban in the software industry. Also, there has not been any significant survey and statistical evaluation of Kanban impact in software engineering for a long period, although Kanban has become very popular day by day in the recent software industry. It has been observed that a survey on Kanban with various aspects like time, cost, quality, team, risk, and performance with various parameters has not been performed, especially for Kanban. There are some additional factors for Kanban challenges, benefits, and its attraction to use that have not yet been explored.

Table 1 List of papers that were already published in journals and conferences were taken into consideration to prepare the actual aim to perform in this paper

Author	Work done
Millstein et al. [8]	Result shows that scrum and Kanban both lead to successful project and Kanban is better than Scrum
Ikonen et al. [11]	Different factors pertaining to motivation, benefits, and challenges were calculated
Lei et al. [13]	An empirical study that investigates how value is interpreted and prioritized, and how value is assured and measured
Polk et al. [10]	The usefulness of descriptions of issues and comments for predicting issue success was studied
Ahmad et al. [14]	To determine the main factors to consider during the selection of Kanban and scrum method

A detailed analytical approach has not yet been done to understand the real impact analysis of Kanban in software engineering.

Study shows that there are multiple tools to track the task and project management. Jira, Microsoft Azure DevOps, CA Agile, and Version One are few tools that have featured most suitable for the projects. Jira is the most competitive and highly adaptable. It can integrate with external apps, and additional functionality can be customized. There is a variation among all the tools few support Agile workflows and almost all support task management. Most of the tools are easy to set up and use. Integration with the external app is the most important feature among all the tools but there are complexities in terms of using these features. The other Kanban tools are Trello, GitHub, Notion, Redmine, Bugzilla, Pivotal Tracker, Mantis Bug Tracker, Asana, YouTrack, Basecamp, OpenProject, Monday, and TeamLeader.

3 Survey Methodology

Survey research has been performed on various aspects of Kanban, and the obtained result has been studied to find the impact analysis of Kanban in the software industry across the globe (locations such as India, United States of America, United Kingdom, Europe, and Australia). An online survey has been conducted with the help of Google forms from 27th October 2020 to 29th November 2020 to find the real data from the actual users who have worked on Kanban, and accordingly statistical and analytical approach has been performed.

The online survey form was developed for capturing the responses with 4 different sections—**1st section** is Kanban: Benefit challenges and available options (this section is to understand the overall experience while working with Kanban). **2nd section** is to capture the observation of workers after the implementation of Kanban (this section captures the effects in the quality of teamwork and in terms of time aspect). **3rd section** is to evaluate Kanban using a statistical approach by capturing

the important factors that are attracted to use Kanban, important benefits that have been achieved from Kanban, and important challenges faced from Kanban in the software industry using a score ranging from 0 to 5. **4th section** is to capture the observation of Kanban while working in your project in terms of Performance, Deadline, Cost, quality, Team, and Risk by using a like scale response ranging from 1 to 5 (Strongly Disagree, Disagree, Neutral, Agree, Strongly Agree). A total of 16 questions were framed consisting of a single choice, multiple choices, providing comments, rating the factor, and selecting the Likert response, and the duration of filling the survey was set to be approximately 30 min.

3.1 Selection of Participants for the Survey

The participants were selected based on their experience and the nature of work they have performed with the help of Kanban. A list of CMMI level 5 companies (TCS, Accenture, Wipro, Microsoft, etc.) were prepared, and accordingly, their employees were approached through LinkedIn, Email, Phone, Social Networking, Professional contacts, and academic contacts. All participants were informed why they have been selected for the study and detail about the purpose of doing this research. They are holding some of the key positions in their organization, and they are among the decision making of the successful projects executed through Kanban in their organization. The participants are holding the designations—Senior Manager, Analyst, Software Engineer, Technical Lead, Consultant, and Agile Lead. Most of the participants are of 1–5 years of experience.

3.2 Effect on Quality of Teamwork After the Implementation of Kanban

In this section, comments have been captured and studied to know the quality of teamwork after the implementation of Kanban. The results of the online survey for this section can be classified into four main themes: code quality has increased, Bugs and defects have reduced, Error has been fixed quickly, and Delivery date has been achieved.

3.2.1 Code Quality

Comments obtained from the online survey on code quality were studied and analyzed. Around 50% of respondents have confirmed that code quality has increased, and very few respondents have confirmed that there is no impact on code quality. To explain the impact on code quality, some of the Interviewee gave very positive

comments “Code is reviewed before it’s getting released and making code quality improvement” and “It provides better avenues to improve code quality through well-defined CI/CD pipeline.” Additionally, “Code has more value-added comments and required split of methods are appropriate to segregate the tasks.” Furthermore, one interviewee mentioned, “Board was modified in a way to call out for reviews, in turn, helping improve quality.” One of the respondents gave negative comments about its impact on code quality, i.e., “Can’t implement purely Kanban for IT development projects due to vast technologies requirement and limited resource.”

3.2.2 Bugs and Defects

Comments obtained from the online survey on Bugs and Defects were studied and analyzed. Around 50% of respondents have confirmed that Bugs and Defects have been reduced, and very few respondents have confirmed that there is no impact on the reduction of Bugs and Defects. To explain the impact on Bugs and Defects, some of the Interviewee gave many positive comments “Yes certainly the defects has reduced with correct planning and implementation of requirements,” “Yes—due to transparency among all team members,” “It also helps in identifying the bugs and defects at an earlier stage of the project and that further results in fewer number bugs and defects at the delivery stage.” Additionally, “Little bit at the cost of resource tracking as who developed that piece of code.” Furthermore, one interviewee mentioned, “Since the team used to describe acceptance criteria for the entire user stories that helped in testing and bugs identification throughout the development in the early stages. Also testing column was used to put the tasks along with their estimated time to be done. Overall, it helped in identifying bugs at an early stage and with good time allocation.” One of the respondents gave comments that reduction depends on “complexity and quality.” None of the respondents have given any kind of negative comments on the reduction of Bugs and Defects.

3.2.3 Error Fixation

Comments obtained from the online survey on Error fixing were studied and analyzed. Around 75% of respondents have confirmed that the error has been fixed quickly, and a very few respondents have confirmed that there is no impact on fixing of error. To explain the impact on fixing of error, some of the Interviewee gave very positive comments “Identified defects are easy to fix as root cause are identified,” “Its highlighted well to the concerned team, takes lesser time for actions.”

Additionally, “Yes. We can check the past similar type of bugs and accordingly time and priority we can set.”, “Yes—due to ownership being taken by team members.” Furthermore, one interviewee mentioned, “It certainly helps in drawing the attention to undergoing error in development. Since team members can label certain tasks with the difficulty being faced and another member can help with it by giving their input. It helps in bringing the transparency and facilitates better

communication among the team members.” One of the respondents confirmed that fixing time has decreased. None of the respondents have given any kind of negative comments on fixing the error.

3.2.4 Delivery Date

Comments obtained from the online survey on achieving the delivery date were studied and analyzed. Many respondents have confirmed that delivery date has achieved but they have given different kinds of opinion like some has confirmed that date has achieved to some extent, partially, almost 80–90%, good impact, completed in the estimated time and a very few respondents have confirmed that there is no impact on achieving the delivery date. To explain the impact on Delivery Date, some of the Interviewee gave very positive comments *“As a daily check-up of the progress of tasks, it’s easy to manage the deliverable dates of each task”, “Helped to plan the project and meet project delivery timelines.”* Additionally, *“It offers many features that help in meeting the deadline goals. Prioritization helps in implementing the most important features first and allocating the time to tasks based on the story points also helps in achieving the deadline.”* Furthermore, one interviewee mentioned, *“was easy to track dates.”* One of the respondents confirmed that it has improved the throughput.

3.3 Time Aspects After the Implementation of Kanban

In this section, comments have been captured and studied to understand the impact of time aspects after the implementation of Kanban. The results of the online survey for this section can be classified into four main themes: tracking of work items, lead time, to meet the customer demand, and delay of product delivery.

3.3.1 Tracking of Work Items

Comments obtained from the online survey on tracking of work items were studied and analyzed. All respondents have confirmed that Kanban has efficiently helped in the tracking of work items. Respondents have confirmed that tracking of work items has become faster, efficient, easier, positive, and good in this measure. To explain the impact on tracking of work items, some of the Interviewee gave very positive comments *“Touch time spent per task went down. Average wait time went down,” “Has improved. Sprint velocity, burn down charts are quite helpful.”* Additionally, *“List increased to manifold as multiple resources for short time were engaged.”*, *“Each deliverable is divided into multiple tasks and tracked effectively with help of Kanban.”* Furthermore, one interviewee mentioned, *“Every activity has a specific time and within that time we can manage to finish work. If there is any delay, we can mention an appropriate reason for more clarity.”* Respondents have confirmed that

tracking of work items becomes smooth in tracking, and it is improved leading to lesser confusion and more productivity, project status is transparent.

3.3.2 Lead Time

Comments obtained from the online survey on lead time were studied and analyzed. Most of the respondents have confirmed that lead time has reduced, and very few respondents have confirmed that lead time has increased. To explain the impact on lead time, some of the Interviewee gave very positive comments *“Onboarding and offboarding resource based on skill requirement equalized the time saved with optimal mapping.”*, *“When we track the task closely the time between commitment time and deliverable time gets estimated with lead time.”* Additionally, *“Overall effective management of the development process helps in delivering the product in the pre-decided time.”* Respondents have confirmed that Kanban helps in managing lead time. Lead time depended on plan and based on work.

3.3.3 Meeting Customer Demand

Comments obtained from the online survey on meeting the customer demand were studied and analyzed. Almost all the respondents have confirmed that Kanban has met the customer demand and only one has confirmed that it is not up to the mark. To explain the impact on meeting the customer demand, some of the Interviewee gave very positive comments *“Can track items and see the progress and plan accordingly if ruining behind schedule,”* *“Is easier and releasing MVPs is quite a good technique. The whole iterative approach is amazing.”* Additionally, *“Quality of project got a little bit better however ongoing maintenance got drastic effect as multiple resources have different documentation style and assumptions which lead to a little bit extra efforts to standardize design and maintenance documentation.”*, *“Any changes while we progress to deliver can get accommodated which helps to meet customer demand better.”* Furthermore, one interviewee mentioned, *“customer demands are well accepted and implemented—thanks to pulling strategy being implemented in Kanban,”* *“Regular meeting with the clients certainly helped in getting their input on the product as a development progressed on and implement any necessary changes.”* Respondents have confirmed that it helps to achieve goals and continuous collaboration with business partners and brings clarity in terms of tasks, dependencies, and management.

3.3.4 Delay of Product Delivery

Comments obtained from the online survey on a delay of product delivery were studied and analyzed. Almost all the respondents have confirmed that either there is no delay, or the delay has been reduced and minimized. To explain the impact on

the delay of product delivery, some of the Interviewee gave very positive comments “Delay of product delivery could happen due to multiple reasons but with the help of Kanban, it could be reduced by following proper monitoring at Kanban board.”, “Can be predicated based on the progress of items inboard.” Additionally, it “Can be better forecasted and managed.”, “If there is a delay of product delivery, can be identified in early stage.” Furthermore, one interviewee mentioned, “Overall time and team management are improved significantly which helps in meeting the deadline goal and avoid any unnecessary delays.” Respondents have confirmed that delay is in control, very less, and sometimes only. One of the respondents gave a negative comment on delay he confirmed that delay has occurred more because tasks pile up.

4 Results and Discussion

A total of 46 respondents working in some of the leading software organizations of CMMI level 5 and holding some of the important designation and roles have filled the survey consisting of abovementioned 4 sections. Kanban is used in versatile projects—Development (78.3%), Maintenance (39.1%), Enhancement (39.1%), and Run the Firm (2.2%). It was found that the planning flexibility (76.1%) and Continuous delivery (71.7%) approach of Kanban are best suited for the software project whereas the pull approach (26.1%) and minimizing cycle time (39.1%) are less suited compared to other factors. 50% of users have recommended visual metrics, and 52.2% have recommended Focus and efficiency for their software project. The above study represents the fraction of nature of work domain specialists advocating various key factors in Kanban that best suit the software project. It can be observed that the most favored nature of the project was “Development Project” followed with “Enhancement and Maintenance Project,” The development focused people preferred “Planning Flexibility” as a leading key factor for Kanban, whereas these groups showed appropriate approach toward “Continuous Delivery” as the key factor for Kanban, but “Pull up approach” was not much recommended by “Development Project” favored group. “Minimized cycle time” was highly recommended by the “Development and Enhancement Project” group of specialists, whereas “Visual Metrics” were appropriately suggested by “Development Project” groups although they were in the highest population, this key had sounding votes from “Maintenance Project” groups. On the contrary, only a few specialists were there in the group of “Development, Maintenance Project” who also run their firm, were as “Development, Maintenance and Enhancement Project” group specialists preferred “Planning Flexibility” as the best key factor for Kanban but “Visual Metrics” were least preferred by them as shown in Fig. 1.

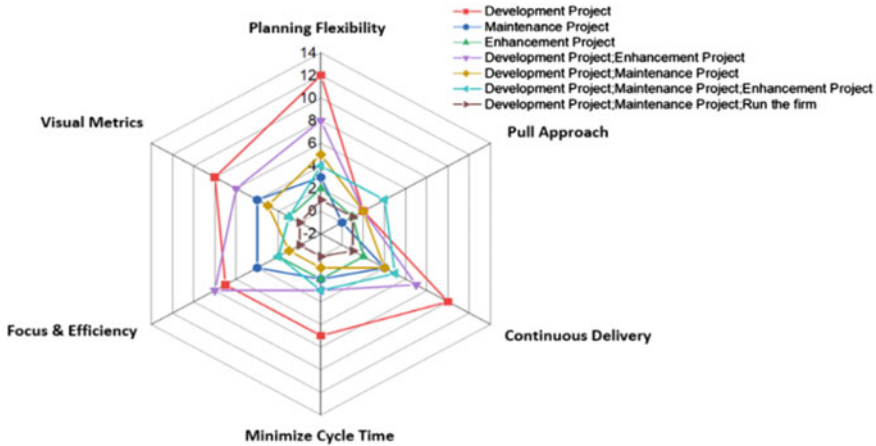


Fig. 1 Key factors in various projects

4.1 Methodology Used

It was found that most of the users use the Kanban software as an agile method in their projects and found it soothing in the management of their projects, whereas “scrum” is also utilized as a method in addition to Kanban for specific projects in their domain. Few engineers do use “XP and Lean” in addition to Kanban, particularly for new projects. Unified process, Team Software Process, RAD, and SAFe are other agile methods to be used along with Kanban for the specific domain. On the contrary, few users found Kanban as not a suitable agile method and found it slow and complicated for their respective projects although they used “scrum” independently or along with “Lean, XP, etc.” for their projects as an agile method.

It was found that 17.4% of the users preferred Kanban software good for new projects, of which 40% found it suitable only for new projects and not good for existing projects as per their past experiences, whereas 26.1% of the users found it suitable for existing projects only, whereas 56.5% of the users found Kanban suitable for new projects as well as for the existing projects as shown by the above Venn diagram in Fig. 2.

4.2 Statistical Approach to Evaluate Kanban

To evaluate the observation of Kanban while working on a software project considering the running factors—Performance, Deadline, Cost, Quality, Team, and Risk, a Likert scale factor as given in Table 2 has been used to capture the numerical response of factors.

Fig. 2 Sections of people preferring Kanban for new and existing projects

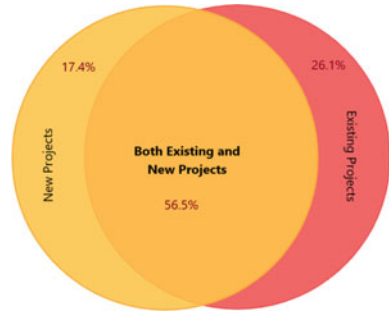


Table 2 Likert scale for factor response

Likert scale for factor response	Score
Strongly disagree	1
Disagree	2
Neutral	3
Agree	4
Strongly agree	5

This section consists of how different factors are impacting the Kanban methodology. Statistical evaluation has been used to conclude the effect of various factors on Kanban. Table 3 shows the observation of Kanban for all the parameters considered for various factor. This also includes the Avg Score for total response count including the percentage for all the Likert scale response. There are 46 respondents of various designations working in various locations and different kinds of project who have given their working observation in terms of Time, Scope, Cost, quality, Team, Risk, and Performance factors of software project. The results show that Average Score is in between 3.5 and 4.0 in all factors.

In our study, we have done the checking to find the correlation factor of Quality concerning Time, Cost, Team, Risk, Scope, and Performance as given in Table 4. All the correlation values come to be positive and this will be very beneficial in examining how Kanban affects the individual factors that could affect the expectation of the client. Correlation values are significant and range from 0.46 to 0.80.

In the earlier study [13], it has been observed that among all the factors Cost, Team, Risk, Scope, and Schedule, the correlation value is positive and there is a very small difference among all values. Hence, it is very difficult to access these factors among themselves. Also, this correlation value was calculated by considering the responses of both Scrum and Kanban users. This gives a comparison between Scrum and Kanban and does not give a clear and valuable indication for Kanban specifically.

We have performed our study specifically for Kanban users and to understand the impact of these running factors Cost, Time, Team, Risk, and Scope. Our calculated result on Pearson’s coefficient gave us an interesting result that helps us to identify the

Table 3 Factors considered for the observation of Kanban

Questions	Strongly disagree	Disagree	Neutral	Agree	Strongly agree	Avg. score
Time: Delivery date has achieved (2.1)	0(0%)	2(4.3%)	7(15.2%)	27(58.7%)	10(21.7%)	3.98
Cycle time has minimized (1.1)	1(2.2%)	3(6.5%)	8(17.4%)	28(60.9%)	6(13%)	3.76
Cost: Completion of project is within estimated cost (3.1)	1(2.2%)	5(10.9%)	10(21.7%)	27(58.7%)	3(6.5%)	3.56
Profit margin has increased with respect to investment (3.2)	1(2.2%)	3(6.5%)	22(47.8%)	15(32.6%)	5(10.9%)	3.44
Quality: Quality of project has increased (4.1)	2(4.3%)	1(2.2%)	12(26%)	24(52.2%)	7(15.2%)	3.72
Expectation of the client is met (4.2)	2(4.3%)	0(0%)	10(21.7%)	25(54.3%)	9(19.6%)	3.85
Team: Efficiency of team has increased (5.1)	2(4.3%)	2(4.3%)	10(21.7%)	23(50%)	9(19.6%)	3.76
Collaboration within team has achieved (5.2)	2(4.3%)	2(4.3%)	8(17.4%)	25(54.3%)	9(19.6%)	3.8
Overburden of team has managed (5.3)	3(6.5%)	1(2.2%)	14(30.4%)	25(54.3%)	3(6.5%)	3.52
Risk: Risk can be easily identified and managed (6.1)	2(4.3%)	4(8.7%)	11(23.9%)	20(43.5%)	9(19.6%)	3.65
Risk can be treated on time (6.2)	1(2.2%)	1(2.2%)	12(26.1%)	24(52.2%)	8(17.4%)	3.81
Occurrence of risk has reduced with the progress of the project (6.3)	3(6.5%)	3(6.5%)	9(19.6%)	22(47.8%)	9(19.6%)	3.68
Scope: Constantly changing requirements are easily adapted in the project team (2.2)	2(4.3%)	1(2.2%)	13(28.3%)	24(52.2%)	6(13%)	3.67
Performance: Throughput has increased (1.2)	2(4.3%)	2(4.3%)	8(17.4%)	23(50%)	11(23.9%)	3.85

contribution of these factors for quality that is an indication for meeting the client's expectation.

The overall study of factors and calculation of Pearson's coefficient for all factors for quality confirms that cost has the least correlation value of 0.46. Hence, the contribution of cost concerning quality in terms of client expectation is minimal in comparison with all other factors. This further indicates that the role and effect of cost for quality that is an indicator of meeting the client expectation are less.

Table 4 Correlation between project success and other factors for Kanban

Factor	Pearson’s correlation with quality factor
Time	0.64
Cost	0.46
Team	0.80
Risk	0.69
Scope	0.64
Performance	0.68

On the other side of our study, we have found that the factor team has a maximum correlation value of 0.80. Hence, the contribution of the team for quality in terms of client expectation is maximum in comparison with other factors. This further indicated that the role and effect of the team are topmost concerning the quality that is an indicator for meeting the client satisfaction.

We have also analyzed each of the individual factors and found the mean and standard deviation for each question across all the respondents. This gave us an important result that time has the least standard deviation of 0.68 in comparison with all other factors. This is an indication among all the factors that time is the best factor for Kanban, and it plays a vital role in achieving the Delivery date and minimization of cycle time.

In the earlier study [13], time factor was not taken into consideration to identify the impact of Kanban. To understand the time factor, we have taken two important parameters Cycle time and Delivery Date.

Our study and overall calculation of Average and Standard deviation values of scores for each factor as given in Table 5 show that Time/Deadline has the least Standard Deviation of 0.68 in comparison with all other factors and this confirms that Time is the most important factor to Kanban and it enables the project to achieve the delivery date and further results in minimization of Cycle time.

Table 5 Average and Standard deviation values of scores for each factor

Factor	Mean	Standard deviation
Time	3.85	0.68
Cost	3.45	0.94
Quality	3.78	0.86
Team	3.71	0.82
Risk	3.71	0.87
Scope	3.67	0.90
Performance	3.83	0.97

Table 6 Nature of project with respect to year of experience for all employee

Yr. of experience (year)	Development	Maintenance	Enhancement	Run the firm
< 1	2	2	3	0
1–3	11	6	6	0
3–5	15	5	5	1
5–10	6	4	3	0
> 10	2	1	1	0

4.3 Preferred Project for Kanban With Respect to Year of Experience

The overall survey and collected data as given in Table 6 reflect that among employees having 1–10 years of experience—development projects are the preferred maximum for Kanban. On the other side, Maintenance and Enhancement are equally preferred by 1–5 years of experience and less in comparison with Development projects.

Maintenance projects are preferred more than Enhancement projects by 5–10 years of experienced employees. One of the respondents in 3–5 years has given his preference for the project that belongs to run the firm. There is no specific kind of preference in < 1 year and > 10 years. In this section, there is almost the same kind of liking for the Development, Maintenance, and Enhancement Project. It concludes that Kanban is more preferred for Development projects in comparison with Maintenance and Enhancement projects.

4.4 Most Favorable Tool for Kanban

Online responses were studied and analyzed to find the best management tools for Kanban. It was found that 84.8% of the users used “Jira” and preferred to it as best tool for processing in Kanban; also, users found it most suitable in ease to setup on compared to other setups of Kanban, whereas some users find Jira as the only tool for Kanban as it is the only setup their firm preferred as best. It has the best project management tool; it holds maximum response record of being user friendly than any other tools as this has direct access of usable items directly on dashboard, which is unlike to some other similar tools. It provides live statuses of progress of the project and signifies the goodness of it. Particularly, the workflow feature in the Streamed line processes has resulted in high accuracy, consistency, and rapid, and the handling becomes much easier with the local accessing featured without much involvement of the administration. It is preferred for bundling similar kind of processes which makes it easier for sorting out the issues of same kind. Due to automated features, it becomes more effecting, and less burden is left with users, as it alarms the notifications on regular basis and real-time assessment is done as shown in Fig. 3.

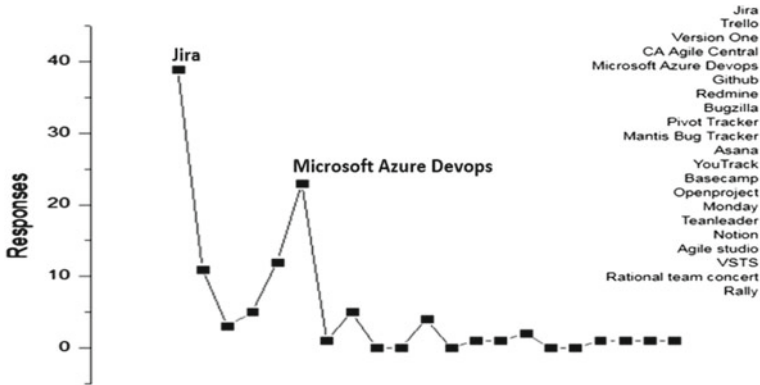


Fig. 3 Preferences on basis of difference in functionality of tools for Kanban

Whereas, almost 50% of users preferred “GitHub” in addition to Jira and other setups, over it 26.1% of them preferred “Microsoft Azure DevOps.” It was found that Microsoft Azure DevOps was found to be suitable in the domain of reliability and flexibility as it provides various functionality options for processing. Most of the users used different tools in Kanban and preferred those based on the flexibility and suitability of their respective projects. Some attention was also drawn to Trello, CA Agile Central, and Bugzilla as feasible setup tools at Kanban, few preferred Trello as best for the processing, whereas least attention was drawn toward Notion, TeamLeader, YouTrack, Pivotal Tracker, etc., as almost negligible users were found for this tool.

4.5 Rate the Important Factors (Attracted, Benefited, Challenges) for Kanban

The software technocrats and managers who are working with Kanban in their real works are requested to rate the important factors of Kanban that have attracted, benefited, and created challenges to them. Tables are created based on 46 online responses.

The result in Table 7 shows that all the above factors that have attracted to use of Kanban are equally impacted in their software development work. This means that Kanban is overall attracted the software developers, project managers, and team members to use Kanban in all the abovementioned factors.

The result in Table 8 shows that all the above factors that are beneficial to use Kanban are equally impacted in their software development work. This means that Kanban is overall benefiting the software developers, project managers, and team members to use Kanban in all the abovementioned factors.

Table 7 Mean and median for factors attracted to use Kanban

Attracted to use Kanban	Mean	Median
Planning flexibility	3.8	4
Pull approach	3.7	4
Minimize cycle time	3.8	4
Continuous delivery	4.2	4
Visual metrics	3.9	4
Focus and efficiency	4.1	4
Full transparency of work	4.2	4
Real-time communication	4.2	4
Client satisfaction	4	4
Removing unwanted activities	4	4
Business and product value	4	4
Development cost	3.5	4
Product quality	3.8	4
Delivery time	4	4

Table 8 Mean and median for important benefits observed while working with Kanban

Benefits	Mean	Median
Better visibility	4.2	4
Improved EFFICIENCY	4.1	4
Increased productivity	4	4
Overburdening of team has reduced	3.7	4
Team focus and attention	4.1	4
Reduced waste	3.7	4
Flexibility	4.2	4
Improved collaboration	4.1	4
More predictability	4	4
Flow control	3.9	4
Responsiveness	3.8	4
Better process understanding	4	4
Better project management	4	4

The result in Table 9 shows that among all the factors of challenges—Progress tracking and working experience of the Kanban board are the biggest challenges developers and managers are observing with Kanban. The remaining all other challenging factors are equally impacting but are less than the progress tracking and working experience of the Kanban board [7].

Table 9 Mean and median for important challenges observed while working with Kanban

Challenges	Mean	Median
Progress tracking	3.4	4
Working experience of Kanban board	3.5	4
Many tasks need to be performed manually	3.4	3
Determining WIP limits	3.4	3
Visualizing workflow	3.3	3
Difficult to prioritize the task	3	3
Specialized training is required	2.9	3
Team unwillingness to follow Kanban	2.8	3
Communication burden within team	2.8	3
No proper planning to adopt Kanban	2.7	3
Management support issue	2.7	3
Budget constraints	2.6	3
Predictability	2.8	3

5 Conclusion

There have not been prior studies to understand the best and widely used tool for Kanban, and survey research analysis has not been performed for a long time to understand the overall experience of Kanban in real software development and maintenance projects. In this research, survey research analysis has been performed specifically on Kanban to evaluate various impacts of Kanban benefits, challenges, and attraction and analyzed the time, cost, quality, team, risk, and performance factors while working with Kanban.

The statistical and analytical approach has concluded that time is the best factor for Kanban, and it plays a vital role in achieving the Delivery date and minimization of cycle time. The correlation coefficient concerning the quality that is an indicator for client expectation indicates that the role and effect of cost for quality that is an indicator of meeting the client expectation are less and effect of Team is topmost for quality that is an indicator for meeting the client satisfaction. This will play a key role in the software community to utilize these factors for the successful execution of projects. The study reflects that Kanban is more preferred for a development project in comparison with the Maintenance and Enhancement Project.

Kanban has proved to be very effective in decreasing the lead time and saving time for further stations. Around 70% have confirmed that Kanban is meeting the customer demand and Jira has proven to be the best and widely used tool in terms of productivity, effectiveness, communication, and metrics.

The study concludes that lack of experience with the Kanban board and progress tracking is the biggest challenge in implementing Kanban.

References

1. Mousaei M Review on role of quality assurance in waterfall and agile software development, p 8
2. Beerbaum D (2021) Applying agile methodology to regulatory compliance projects in the financial industry: a case study research, social science research network, Rochester, NY, SSRN Scholarly Paper ID 3834205. <https://doi.org/10.2139/ssrn.3834205>
3. Agile Methodologies http://www.umsl.edu/~sauterv/analysis/6840_f09_papers/Nat/Agile.html (Accessed Aug 20 2021)
4. Anderson DJ (2010) Kanban: successful evolutionary change for your technology business. Blue Hole Press
5. Birkeland JO (2010) From a timebox tangle to a more flexible flow. In: Agile processes in software engineering and extreme programming, Berlin, Heidelberg, pp 325–334. https://doi.org/10.1007/978-3-642-13054-0_35
6. Senapath (2021) Systems thinking approach to implementing Kanban: a case study. J Softw Evol Process—Wiley online library. <https://doi.org/10.1002/smr.2322> (Accessed 20 Aug 2021)
7. Ahmad MO, Dennehy D, Conboy K, Oivo M (2018) Kanban in software engineering: a systematic mapping study. J Syst Softw 137:96–113. <https://doi.org/10.1016/j.jss.2017.11.045>
8. Takt Time Grouping: implementing kanban-flow manufacturing in an unbalanced, high variation cycle-time process with moving constraints. Int J Prod Res 52(23). <https://doi.org/10.1080/00207543.2014.910621> (Accessed 20 Aug 2021)
9. Neely S, Stolt S (2013) Continuous delivery? Easy! just change everything (well, maybe it is not that easy). In: 2013 Agile conference, pp 121–128. <https://doi.org/10.1109/AGILE.2013.17>
10. Polk R (2011) Agile and Kanban in coordination. In: 2011 agile conference, pp 263–268. <https://doi.org/10.1109/AGILE.2011.10>
11. Ikonen M, Pirinen E, Fagerholm F, Kettunen P, Abrahamsson P (2011) On the impact of kanban on software project work: an empirical case study investigation. In 2011 16th IEEE international conference on engineering of complex computer systems, pp 305–314. <https://doi.org/10.1109/ICECCS.2011.37>
12. Alaidaros H, Omar M, Romli R (2018) Towards an improved software project monitoring task model of agile kanban method 7:118–125
13. Lei H, Ganjeizadeh F, Jayachandran PK, Ozcan P (2017) A statistical analysis of the effects of scrum and Kanban on software development projects. Robot Comput-Integr Manuf 43:59–67. <https://doi.org/10.1016/j.rcim.2015.12.001>
14. (PDF) Usage of Kanban in software companies an empirical study on motivation, benefits and challenges. https://www.researchgate.net/publication/267515017_Usage_of_Kanban_in_Software_Companies_An_empirical_study_on_motivation_benefits_and_challenges#read (Accessed 20 Aug 2021)

Integrated Bioinformatics Analysis to Identify the Potential Molecular Biomarkers for Neuropathic Pain Among Patient of Lumbar Disc Prolapse and COVID-19



Manisha Chaudhary and Veena Puri

Abstract Lumbar disc prolapse (LDP) is a major cause of low back pain that impacts almost 9% population around the world, with a great tendency to increase with the age. The purpose of this study is to better understand the underlying mechanism of LDP by identifying the pathways linked with neuropathic pain and potential molecular biomarkers of such pain in LDP patients and Coronavirus Disease 2019 (COVID-19) by integrating transcriptome bioinformatics analysis. From the Gene Expression Omnibus repository, the gene expression profile of GSE124272 was extracted for analysis. The Bioconductor package named Limma was employed in Rstudio to identify differential genes, and a total of 669 dysregulated genes between the disease group and the control group were identified (424 upregulated and 245 downregulated) which were then subjected to Gene Ontology pathway enrichment analysis using clusterProfiler package that revealed 751 upregulated pathways and 901 downregulated pathways. Most of the upregulated pathways were involved in cell killing and inflammatory or immune response; however, downregulated pathways were involved in response to mechanical stimulus, regulation of cell cycle, and ion transport. Differentially expressed genes of LDP were compared with the gene list of COVID-19 obtained from the National Center for Biotechnology Institute Gene database, and the intersecting genes CEACAM1, FCGR3A, IL1RN, and REN between upregulated LDP and COVID-19 genes and PLG and NR1P1 in between downregulated LDP and COVID-19 genes were identified. These were the six potential molecular biomarkers screened for chronic neuropathic pain in patients suffering from LDP and COVID-19.

Keywords Chronic pain · Neuropathic pain · Lumbar disc prolapse · COVID-19 · SARS-CoV-2

M. Chaudhary · V. Puri (✉)

Centre for Systems Biology and Bioinformatics (U.I.E.A.S.T), Panjab University, Chandigarh 160014, India

e-mail: vpuri1825@pu.ac.in

1 Introduction

The occurrence of neuropathic pain is near about 7–10% worldwide. Neuropathic pain is quite possibly the most common symptom affecting lumbar disc prolapse (LDP) patients. This condition might lead to a lifetime impairment, affecting the standard of living and an increase in the death rate [1]. Around eighty percent of the populace suffers from low back pain (LBP) once in the course of their life. Because of its pervasiveness and huge involvement in dysfunction, it causes a yearly expense exceeding \$100 billion in the USA [2]. It becomes necessary to elucidate the molecular mechanism for the remedy of pain after LDP.

The most ordinary levels for a prolapsed disc are L4-5 and L5-S1. The beginning of symptoms is described by a sharp, burning, agonizing pain transmitting down the posterior or lateral part of the leg, to beneath the knee. Pain is usually superficial and localized and is often related to numbness or itching [3]. The specific mechanisms underlying LDP, a particular type of intervertebral disk degeneration disease (DDD), have rarely been explored exclusively, most of the studies on intervertebral disc degeneration (IDD) depend upon the evaluation of tissues of the intervertebral disc, which is a complicated layout comprised of annulus fibrosus and nucleus pulposus inside the spinal region, and LDP has been related to disruption of the annulus fibrosus (AF), the expulsion of the nucleus pulposus (NP), and stimulation of nerve fibers, which leads to pain.

With increasing age, an individual becomes more vulnerable to disc injuries over a while [4]. It is generally common, with 5–20 cases for every 1000 adults yearly, and relatively pervasive in the third to the fifth decade of life, with a male to female proportion of 2 to 1 [5]. The medical remedy of the present situation has been restricted due to the current constraint in the comprehension of its etiology. Hence, enormous expertise in the fundamental working of LDP is essential for the improvement of successful restorative systems.

In December 2019, a new agent, named severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2), brought about a pandemic of acute respiratory illness and was identified as the cause of this epidemic originally from the region of Wuhan, China [6]. The World Health Organization (WHO) designated the disease caused by SARS-CoV-2 as Coronavirus Disease 2019 (COVID-19) [7]. Additionally, conceivable neuropathic torment has been represented in approximately 2.3% of patients admitted to the hospital with coronavirus disease in the initial course; nevertheless, the pervasiveness of such pain is still underrated [8].

Taking into consideration the significance of the neuronal problems of COVID-19, it is assumed many patients experiencing this disease will have neuropathic agony in less than a fortnight or months or that sufferers will show the failure of their neurological difficulties or elevation of their torment. However, various prospective studies are in progress in France in previously admitted ICU patients, and in Canada, Internet-based surveys have been directed these days to determine the coronavirus risk in victims of persistent pain [9]. Thus, it will be important to tentatively follow such people who are suffering from persistent agony impacted due to coronavirus

disease and differentiate regardless if the torment threat is intensifying in individuals with neuropathic in contrast with non-neuropathic.

In this study, bioinformatics methods were utilized to explore the mRNA transcriptome features of LDP patients acquired through the GEO repository; in addition, the dysregulated genes were discovered in people without or with neuropathic torment thereafter LDP and subjected to functional analyses. We focused to identify the new molecular biomarkers for neuropathic agony in patients suffering from LDP and COVID-19. A significant relationship between neuropathic pain with LDP and COVID-19 was obtained via evaluating their intersecting genes suggesting its role in the mechanism of the inflammatory and immune response. Potential molecular biomarkers of chronic neuropathic pain in LDP and COVID-19 were also obtained using various bioinformatics tools in the Rstudio.

2 Methods

2.1 Microarray Analysis

The gene articulation profile GSE124272 based on the GPL21185-Agilent-072363 SurePrint G3 Human GE v3 8 × 60 K Microarray 039494 platform was extracted from the GEO repository [10]. We used “neuropathic pain” as a keyword for data retrieval. This dataset has eight patient samples with lumbar disc prolapse that were confirmed by magnetic resonance imaging (MRI) and eight healthy samples that do not have any clinical evidence of low back pain or sciatica.

2.2 Identification of DEGs

The scan data of the microarray chip was already log₂ normalized for comparative analyses. The raw data downloaded from the GEO database was converted into the form of an expression matrix CSV file and imported into R [11]. The Limma package [12] in Bioconductor was utilized to discover differential genes [13]. Additionally, it provides *t*-statistics and *p*-values. In this study, dysregulated genes amid neuropathic torment (disease category) and healthy volunteers (control category) were filtered and chosen with the threshold criteria of *p*-value less than 0.05 and log₂ fold change (log₂ FC).

2.3 *Functional Enrichment Analysis*

Enrichment analysis of differentially expressed genes was assessed based on Gene Ontology (GO) [14] over-representation using the ClusterProfiler package [15] in Rstudio. Gene ontology is a very common and useful method to annotate gene and gene products and for identifying characteristic biological attributes of high-throughput transcriptome data that consists of three categories: biological process (GO-BP term), cellular component (GO-CC term), and molecular function (GO-MF term) [16]. The gene list must contain these four features: Log2 fold change values, a list of gene names, Not a Number (NaN) values should be removed, and gene list should be sorted in decreasing order.

The criteria for filtering the GO terms, p -value equals 1, and gene Set-size ≥ 5 was set as a cut-off. Benjamini and Hochberg's method for the rectification of the extent of all bogus discoveries was implemented to the natural p -value to obtain an adjusted p -value.

2.4 *Validation of Core Genes*

COVID-19 gene list was extracted from the National Center for Biotechnology Institute (NCBI) Gene database [17] and then imported into Rstudio in CSV format. Two data frames were generated for upregulated and downregulated gene symbols of LDP, and their NaN values were removed. Then, the gplots package [18] was used in R to generate Venn diagrams between coronavirus and LDP genes. Heatmaps were constructed using the Pheatmap package [19] in R to further analyze the expression pattern of key genes across the groups. RColorBrewer package [20] provides a color palette to generate a colorful heatmap according to our choice.

3 **Results and Discussion**

3.1 *DEG Analysis*

Neuropathic pain is an exceptionally disabling disorder, for which treatments are inadequate, and puts an extensive financial burden on society and the health administration. Drug discovery efforts are expensive and highly risky due to an absence of a clear understanding of the systems of the development and maintenance of pain. Transcriptomic studies have turned into a common approach to understanding diseases and drug mechanisms [21]. Various studies have been reported on transcriptome analysis of neuropathic pain in dorsal root ganglion after spinal cord injury [22] or peripheral nerve injury [23].

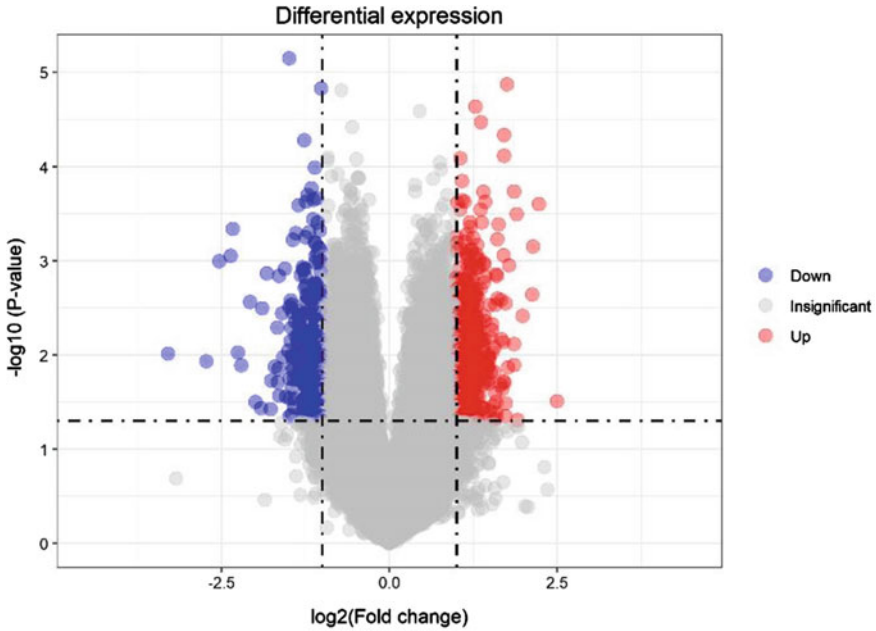


Fig. 1 Volcano plot indicating the log₂ fold change (x-axis) of gene expression and the statistical significance (y-axis) of the differential expression (DE) analysis performed between disease v/s control group for GSE124272. Each point represents one gene. The horizontal dashed line shows the *p*-value cutoff of <0.05, and the vertical dashed line shows log₂ fold change between ±1

In this study, 58,341 probes were acquired using the Agilent technology. As illustrated in Fig. 1, there were 5923 insignificant genes, and an aggregate of 669 genes was recognized to be dysregulated between lumbar disc prolapse (LDP) patients and the control group with the cutoff of log₂ fold change ±1 and *p*-value < 0.05. Among them, there were 424 overexpressed genes displayed through red color; however, 245 were repressed genes shown with blue color, while the rest were in gray. A volcano plot was constructed for these differentially expressed genes (DEGs) identified. Log₂ fold change values were used as they can reveal the genes dramatically dysregulated in an experiment. However, many researchers use log₂ fold change as one of the several criteria to apply to the microarray data analysis [24].

Table 1 depicts the top significantly enriched differentially expressed genes (DEGs). The genes screened with positive log₂ fold change value were termed as upregulated, whereas the genes screened with negative log₂ fold change were termed as downregulated. The log₂ fold change for the upregulated genes ranges from 1.000333 to 2.496979, and the *p*-value ranges from 0.04953086 to 1.336839e-05. For downregulated genes, log₂ fold change range between -1.000921 to -3.290563 and the *p*-value from 0.049592207 to 7.083443e-06.

Table 1 Top 20 up and downregulated genes based on the above-mentioned threshold with their gene expression ratios

S. No.	ID	Log ₂ FC	P-value	T-test	Expression
1	A_24_P181254	2.496979	3.096585e-02	2.376824	Up
2	A_33_P3369944	2.230679	2.511422e-04	4.741740	Up
3	A_22_P00005118	2.140569	7.111626e-04	4.224611	Up
4	A_33_P3276845	2.129714	2.276188e-03	3.658007	Up
5	A_23_P71649	1.985203	3.857398e-03	3.402777	Up
6	A_23_P19291	1.911638	4.905737e-02	2.138477	Up
7	A_21_P0011503	1.899523	3.222180e-04	4.616744	Up
8	A_23_P130961	1.860880	1.281991e-02	2.818058	Up
9	A_22_P00002179	1.860696	7.687096e-03	3.068336	Up
10	A_23_P126540	1.858416	1.848077e-04	4.896762	Up
11	A_21_P0002758	-1.000921	2.054566e-02	-2.584080	Down
12	A_21_P0000515	-1.002780	2.783796e-03	-3.560574	Down
13	A_22_P00008939	-1.003268	1.135551e-02	-2.877677	Down
14	A_24_P944222	-1.003819	1.006201e-02	-2.936940	Down
15	A_22_P00016879	-1.004407	4.843097e-03	-3.292617	Down
16	A_33_P3214239	-1.004548	1.792222e-02	-2.652253	Down
17	A_23_P134663	-1.006300	4.484163e-02	-2.185616	Down
18	A_23_P257971	-1.006532	1.307808e-02	-2.808240	Down
19	A_21_P0008575	-1.008004	4.959221e-02	-2.132767	Down
20	A_24_P377775	-1.008638	3.550334e-02	-2.306731	Down

3.2 Functional Enrichment Analysis

The Gene Ontology (GO) enrichment analysis revealed that 751 pathways were found to be upregulated, which were enriched in 553 biological processes, 111 cellular components, and 87 molecular functions terms, and 901 pathways were found to be downregulated, which were enriched in 710 biological processes, 107 cellular components, and 84 molecular functions terms. As illustrated in Figs. 2 and 3, clusterProfiler performed over-representation analysis using dot plot, which pinpoints different pathways of DEGs related to different ontologies [25]. *P*-values and adjusted *p*-values were filtered out by the threshold given by the parameter of *p*-value cut-off equals 1, minimum count set at 5 and FDR can be estimated by the Benjamini–Hochberg method.

Here, dots address the term enrichment with a color gradient that ranges from red to blue corresponding to increasing *p*-values: Red color indicates the high function enrichment and less *p*-value, and blue demonstrates the low enrichment and elevated *p*-value. The enrichment score shows how much a gene set is excessively present at

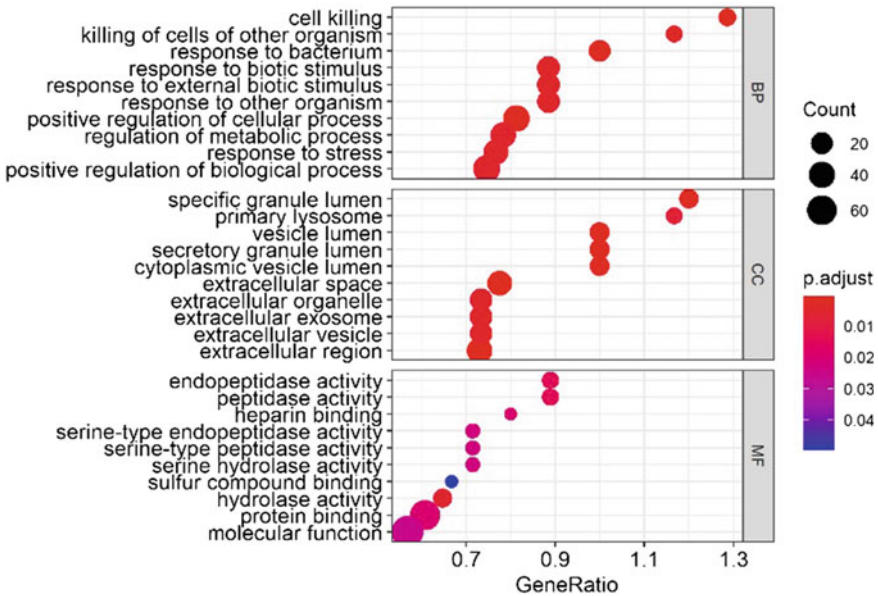


Fig. 2 Functional enrichment analysis visualized as dot plot using clusterProfiler for upregulated genes. Here, the GO terms of the 424 upregulated genes related to each ontology were ranked according to the gene ratio. The gene ratio was defined as follows: Count divided by Set-size. The term “count” indicates the range of genes that have a place with a given gene set, whereas “Set-size” is the complete number of genes inside the gene set. In this figure, the x-axis is represented by gene ratio, dot color by *p*-adjusted value, and count as the dot size

the top or lower part of an ordered list of genes. The sizes of the dots represent the sum total of genes belonging to a gene set. As we can see that upregulated pathways majority display red color dots, this suggests that a greater number of upregulated pathways have less *p*-adjusted value and are highly enriched in the neuropathic pain in lumbar disc prolapse patients. However, downregulated pathways majority display blue color dots, which suggest that a greater number of downregulated pathways have high *p*-adjusted value and are less enriched in the neuropathic pain in lumbar disc prolapse patients. Further study of these pathways can yield productive results to get a better understanding of the mechanism. This analysis provides significant data for further investigations.

The GO enrichment analysis indicated that upregulated DEGs were primarily associated with “cell-killing”, “regulation of cellular activity inflammatory response or immunity”, “signal transduction”, “defense response” [26] in the category of GO-BP terms, “specific granule lumen [25]”, “extracellular space [2]” in the category of GO-CC terms and “protein binding”, “peptidase activity”, “ion channel activity” in the category of GO-MF terms. Most of the biological pathways in upregulated genes were about the inflammatory response or immunity, and the excessively critical enrichment GO category was “leukocyte mediated cytotoxicity (GO:0, 001, 909)”

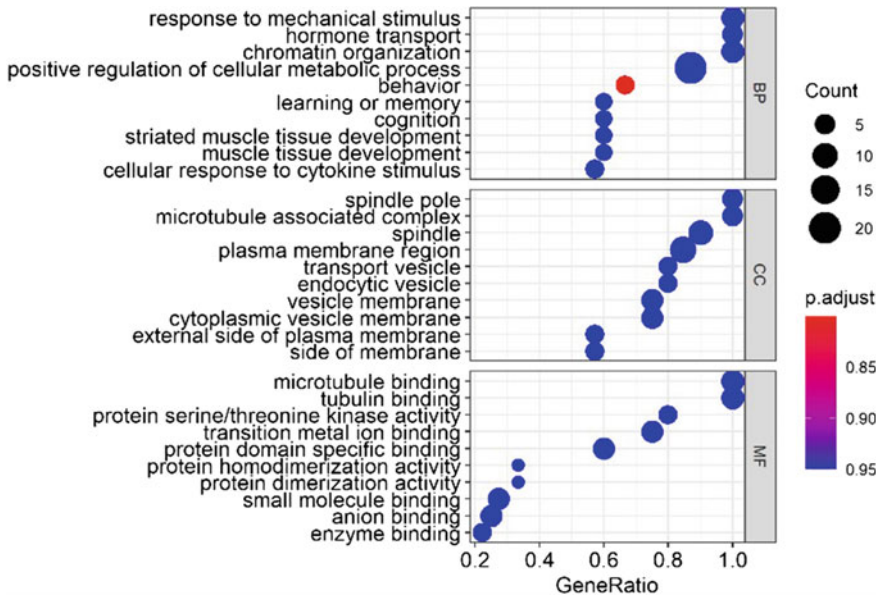


Fig. 3 Functional enrichment analysis visualized as dot plot using clusterProfiler for downregulated genes. Here, the GO terms of the 245 downregulated genes related to each ontology were ranked according to the gene ratio. The gene ratio was defined as follows: Count divided by Set-size. The term “count” indicates the range of genes that have a place within a given gene set, whereas “Set-size” is the complete number of genes inside the gene set. In this figure, the x-axis is represented by gene ratio, dot color by *p*-adjusted value, and count as the dot size

[27] with a *p*-value of 0.0005490003. Other significant categories were related to response to stimulus, regulation of signal transduction, and positive regulation of cellular processes. These results confirmed that immune response is very important in neuropathic pain in LDP [22, 26–28].

Previous models based on neuropathic torment displayed the verification of an important technical role for immune cells in the constancy of torment [29]. However, human imaging researches are steady with preclinical disclosures, with glial enactment obvious inside the brain of sufferers encountering persistent pain. Thence, insusceptible cells are not generally viewed as uninvolved bystanders in the sensory system; an understanding is emerging that, through their correspondence with neurons, the two of them can engender and stay aware of sickness states, including neuropathic torment. In the spinal cord, an immoderate neuronal firing influences the critical enactment of immune cells, to be explicit microglia, that prompts neuroinflammation. The key effector systems of the cellular immune reaction are named inflammation and cytotoxicity [30]. The symptoms like harm, pain, swelling, and redness are in huge part the result of the roles of inflammatory mediators, for example, cytokines, chemokines, and vasodilators delivered from immune cells in an interaction intended to secure and ease the maintenance of harmed tissue. On the opposite side, the job of cytotoxicity is to destroy alive cells as well as savage

tissues through apoptotic techniques and is a significant function of immunity against malignant growth.

The over-represented enrichment analysis demonstrated that downregulated genes were primarily associated with “response to mechanical stimulus”, “transport”, “regulation of cell cycle” in the category of GO-BP terms, “plasma membrane”, “cell junction” in the category of GO-CC terms and “tubulin-binding”, “cell adhesion and molecular binding” [1] in the category of GO-MF terms. Most of the biological pathways in downregulated genes were related to the cell cycle, and the excessively critical enrichment GO category was “response to a mechanical stimulus (GO:0, 009, 612) [5, 23]” with a p -value of 0.0003956992. These results confirmed that response to external stimuli is very important in neuropathic pain in LDP.

The nervous system recognizes and deciphers a broad variety of warm and mechanical stimuli including ecological and intrinsic synthetic aggravations. At the point when extreme, these boosts produce intense pain, and on account of tenacious agony, both peripheral and central sensory systems as parts of the aggravation transmission pathway show enormous versatility, expanding torment flags and creating touchiness [31]. Such stimuli produce ongoing aches and two sorts of hyperalgesia: primary and secondary. Primary hyperalgesia takes place at the location of tissue injury and is intervened in element with the aid of the sensitization of number one afferent nociceptors, which is mediated through the enhanced reaction to warmth upgrades, as an instance. Secondary hyperalgesia happens in the unharmed group of cells encompassing the area of the wound and is believed to be because of stimulation in the CNS. Secondary hyperalgesia is portrayed by utilizing hyperalgesia to mechanical, however no longer warm, upgrades. It is akin to the hyperalgesia found inside the victim with neuropathic torment [32]. There are two sorts of mechanical hyperalgesia noticed: agony to light-stroking boosts which is allodynia and increased agony to punctate improvements. A couple of psychophysical investigations in the participants give solid proof that optional hyperalgesia is because of sensitization inside the CNS.

3.3 Validation of Core Genes

The Venn diagram summarizes the overlap among upregulated genes, downregulated genes in lumbar disc prolapse, and the list of COVID-19 genes derived from NCBI’s Gene Database (Fig. 4). The number of genes common between them was indicated by the overlap between the two circles. There were four common genes, “CEACAM1”, “FCGR3A”, “IL1RN”, “REN”, found between upregulated and COVID-19, whereas only two common genes, “PLG”, “NRP1”, were found in downregulated and COVID-19. However, no intersecting genes are found among all three groups. The number of genes involved in upregulated genes (UP) was 223, downregulated genes (DOWN) were 178, and the number of genes extracted from NCBI’s GENE database for coronavirus disease (COVID) was 220.

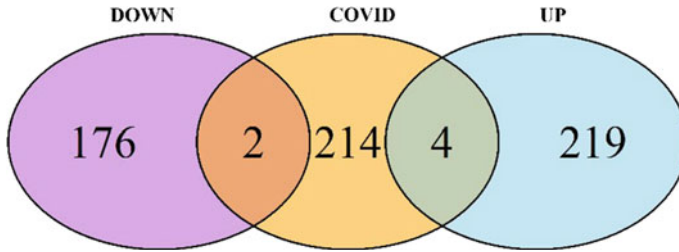


Fig. 4 Venn diagram for neuropathic pain in dysregulated genes of LDP and the list of genes derived for coronavirus from NCBI's Gene Database

3.4 Molecular Biomarkers of Neuropathic Pain in LDP and COVID-19

The molecular biomarkers of neuropathic pain caused by LDP and COVID-19 were identified in this study including CEACAM1, FCGR3A, IL1RN, REN, PLG, and NRP1. These genes and their roles in neuropathic pain are defined as below:

Carcinoembryonic antigen-related cell adhesion molecule 1 (CEACAM1) gene cipher from the family of carcinoembryonic antigen (CEA) that has a place with the immune globulin superfamily. Various researches have revealed the significant role of CEACAM1 in the progression of COVID-19 by regulating the cell–cell communication of developing neutrophils and type II pneumocytes in peripheral blood mononuclear cells and bronchoalveolar lavage fluid, respectively [33]. The encoded protein intervenes in cell adhesion through homophilic and heterophilic binding to different proteins of the subgroup and is recognized on leukocytes, epithelial, and endothelial.

Numerous cellular exercises seem credited toward the ciphered protein, also comprising jobs inside the separation along with the course of action of tissue three-layered design, apoptosis, cancer elimination, angiogenesis, metastasis, and the adjustment of adaptive and innate immune reactions. CEACAM1 performs a crucial role in the immune response of T cells, natural killer (NK), and neutrophils [34]. In addition, it hinders T cell multiplication and cytokine formation via restraints of the JNK cascade and plays a pivotal role in controlling autoimmunity and against cancer resistance by repressing T cells via its response with HAVCR2.

Fc fragment of IgG receptor IIIa (FCGR3A) gene engages with the expulsion of antigen–antibody complexes and a few other responses like antibody-dependent and natural killer cell-mediated cytotoxicity and antibody-dependent improvement of the virus infections. Studies have confirmed that the FCGR3A gene mediates phagocytosis and inflammatory molecule plasminogen (PLG) activates host defense response and immunological pathways in low back pain [35].

This FCGR3A gene is profoundly like one more nearby gene FCGR3B situated upon chromosome 1. FCGR3A receptor expresses on natural killer (NK) cells

in the form of integral membrane glycoprotein that is anchored through the transmembrane peptide. This gene has also been reviewed for its involvement in SARS-CoV-2 infection. Alterations in this gene are related to immunodeficiency, and it is connected to sensitivity to repetitive viral diseases, vulnerability to alloimmune neonatal neutropenia, and fundamental lupus erythematosus [36].

The protein encoded by the interleukin 1 receptor antagonist (IL1RN) gene is a member of the interleukin 1 cytokine family. It modulates a variety of interleukin 1-related immune and inflammatory reactions, especially in the acute phase of infection and inflammation. It is a strong anti-inflammatory particle that works on the ordinary actions of the interleukin 1, proinflammatory cytokines [37]. IL1RN was at first depicted as a normally happening antagonist for the IL-1 receptor. Conservation of a balance among the IL-1 & IL1RN ranges in the localized group of cells impacts the corresponding inflammatory impacts of IL-1, and variation of the equilibrium inclines cells to the generation of a variety of infections. IL1RN has been observed to be related to the risk of lumbar disc prolapse in Finland samples, and it may play a defensive part against the risk of LDP, which can be further utilized to find targets for clinical prevention and/or treatment [38]. Its protein level was likewise observed to be raised in the plasma of COVID-19 patients [39].

The REN (renin) gene encodes renin, an aspartic protease that is discharged by the kidneys. Aside from the appropriately recorded function of the RAAS (renin-angiotensin-aldosterone system) in managing blood pressure with additional associated factors, its job in regulating diverse physiological or pathological features which include neuropathic agony [40]. Like its double function in regulating pressure-related tension and psychological capabilities, its additional function has been recorded in pain balance in diverse disorder conditions.

Medications restricting the RAAS enactment, via aldosterone opponents, AT1 receptor enemies, angiotensin-converting enzyme (ACE) restraints, and renin constraints, were displayed for delivering useful outcomes in headache, neuropathic and nociceptive torment. Its useful influences were specifically attributed to the hindrance of the inflammatory course of responses through repressing the creation of key cytokines, such as tumor necrosis factor (TNF)- α [41]. RAAS is involved in the COVID-19 pandemic through the crucial function of ACE2 during viral infection. Modification inside the RAAS gene may be related to the heterogeneous response to SARS-CoV-2 infection [42].

The plasminogen (PLG) gene provides guidelines for making a protein known as plasminogen, which is created in the liver. As an expansive range protease, plasminogen has multiple roles that add to the guideline of an inflammatory reaction [43]. Plasminogen additionally plays a part in cell relocation and enrollment of insusceptible cells toward the location of the wound. The increased plasminogen level lowers the plasmin activity and decreased fibrinolysis, which provides some explanation for the fibrin-mediated increase in blood viscosity and hypercoagulability in COVID-19 patients [44]. It is a critical intermediary of perivascular macrophage relocation which acts as a pivotal part in the correspondence of inflammatory signs among the CNS and periphery [45]. Depletion of PLG just in the periphery region causes a

reduction in this neuroimmune reaction, prompting a significant discovery about the guideline of neuroinflammation from fundamental particles [46].

Neuropilin-1 is a member of a signaling family, became displayed to fill as an access factor and potentiate SARS coronavirus 2 (SARS-CoV-2) infectivity in vitro [47]. The expression of this cell surface receptor is significant in angiogenesis, multiplication of the tumor, the passage of virus, axonal direction, and immune role. It is involved in various respect of SARS-CoV-2 contamination along with conceivable spread via the olfactory bulb and into the CNS and extended NRP-1 RNA articulation in lungs of extreme Coronavirus Disease 2019 (COVID-19). The role of NRP-1 in insusceptible characteristics is convincing, considering the function of an overstated safe reaction in ailment seriousness and demise because of coronavirus disease. It is recommended to be an insusceptible spot of T cell memory [48]. It is obscure whether the contribution and elevation of NRP-1 in coronavirus disease might change over into infection results and long-haul reactions, consisting of feasible immune disorder. Prior studies revealed that a huge decline of Nrp-1 in the degenerate human intervertebral disc (IVD) causes expanded neural ingrowth and propose that Nrp-1 would possibly have a regulatory function in disc degeneration [49].

Construction of heatmaps. The heatmaps were generated to analyze the pattern of dysregulation of genes as well as how their expression was correlated with each other to get a superior comprehension of the pathophysiology of LDP.

The dendrogram was created using the Pearson correlation coefficient. The genes were clustered in terms of expression. In the hierarchical grouping dendrogram of the gene articulation, the x-axis at the lower part of the map displays the label of genes, and the y-axis on the right facet addresses the type of sample in LDP. The y-axis on the left facet tends to the two forms of shading band, i.e., “orange” for diseased samples and “green” for control samples, and the x-axis at the top shows the level of bunching of samples. Here, the red shading represents overexpression; however, the blue shading represents repression. The more obscure red demonstrates a robust upregulation in the articulation, and the stronger blue shows a robust downregulation in the articulation. Moreover, we determined the Pearson relationship to generate the gap in the middle of the genes and samples which executed the progressive grouping based totally on the used common technique for the association.

The intersecting genes involved in LDP upregulated and COVID-19 have been presented in Fig. 5. Here, we can observe genes REN, CEACAM1, and IL1RN that show similar expression patterns across the groups. The genes were upregulated in the disease group (patient with lumbar disc prolapse). Out of these genes, CEACAM1 and IL1RN were closely grouped. FCGR3A gene expression was slightly decreasing in the control group. This gene is not clustered with others as it shows quite different expression patterns.

REN, IL1RN, and FCGR3A show difference in their expression values from CEACAM1. They tend to show neutral gene expression scores, REN displays it in the disease group, however, IL1RN and FCGR3A in the control group. Thus, it can be said that these genes start to get dysregulated in patients suffering from lumbar disc

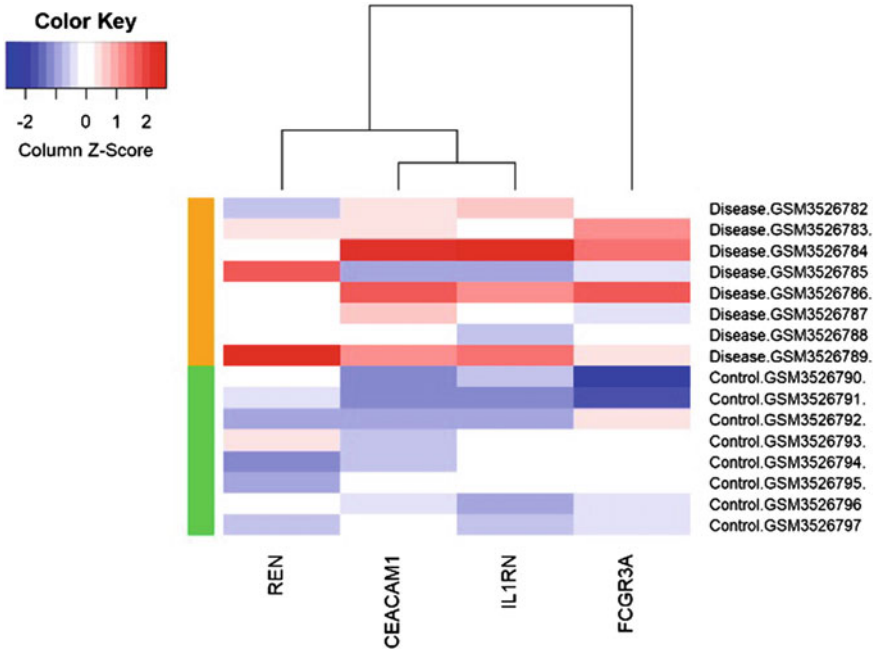


Fig. 5 Heatmap to study the gene expression pattern for the intersecting genes involved in LDP upregulated and COVID-19 across the groups

prolapse with no or little neuropathic pain (NP) and get infected with the coronavirus. Hence, these genes can function as the markers for the early analysis of neuropathic pain followed by COVID-19 and lumbar disc prolapse (LDP) and can be targeted to develop potential therapies or medicines for the treatment of chronic neuropathic pain.

The genes in Fig. 6 show similar gene expression patterns. It can be seen that genes in the disease group seem to be slightly downregulated. The NRP1 gene displayed neutral gene expression scores in the control group. It can be inferred from this observation that the process of dysregulation in the patients suffering from lumbar disc prolapse with no or little neuropathic pain (NP) and get infected with the coronavirus. Hence, these genes can act as the markers for the early identification of neuropathic pain followed by COVID-19 and lumbar disc prolapse (LDP) and can be targeted to develop potential therapies or medicines for the treatment of chronic neuropathic pain.

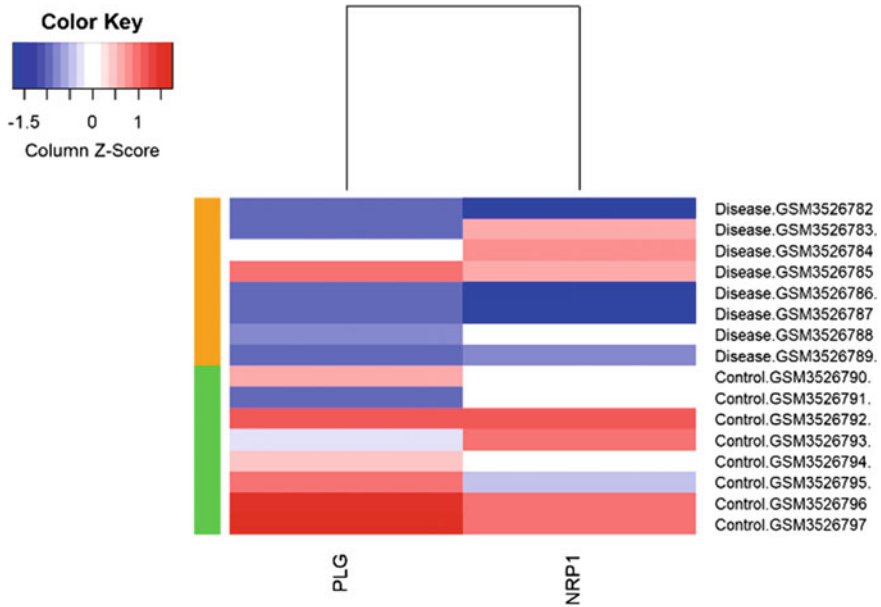


Fig. 6 Heatmap to study the gene expression pattern for the intersecting genes involved in LDP downregulated and COVID-19 across the groups

4 Conclusion and Future Prospects

The present study demonstrated that CEACAM1, FCGR3A, IL1RN, REN, PLG, and NRP1 were identified as potential biomarkers of LDP and COVID-19 using various bioinformatic tools and techniques. LDP depends on intervertebral disc degeneration, and the degree of histopathologic degeneration of lumbar intervertebral discs varies. On account of such groupings, DEGs may be associated with other related mechanisms and not just IDD. Thus, the convergence of the dysregulated LDP and COVID-19 genes was focused on obtaining molecular markers.

To recapitulate, we discovered inflammatory or immune attributes as a peripheral blood transcriptomic signature for lumbar disc prolapse and Coronavirus Disease 2019 and recognized genes that can be crucial for mRNA dysregulation in patients with LDP. Concerning now, they have no reliable information related to the pervasiveness and clinical highlights of neuropathic pain in patients having the coronavirus disease; however, several potential kinds of research are in progress in France and Canada. Future investigations should further portray the role of the peripheral inflammatory and immune response played in the pathology of LDP and COVID-19 and regulate its importance to overall disease progression to help millions of people suffering.

References

1. Wang Y, Dai G, Li L, Liu L, Jiang L, Li S, Liao S, Wang F, Du W, Li Y (2019) Transcriptome signatures reveal candidate key genes in the whole blood of patients with lumbar disc prolapse. *Exp Ther Med* 18(6):4591–4602. <https://doi.org/10.3892/ETM.2019.8137>
2. Liu H, Xia T, Xu F, Ma Z, Gu X (2018) Identification of the key genes associated with neuropathic pain. *Mol Med Rep* 17(5):6371–6378. <https://doi.org/10.3892/MMR.2018.8718>
3. Amin RM, Andrade NS, Neuman BJ (2017) Lumbar disc herniation. *Curr Rev Musculoskelet Med* 10(4):507. <https://doi.org/10.1007/S12178-017-9441-4>
4. Gugliotta M, Da Costa BR, Dabis E, Theiler R, Jüni P, Reichenbach S, Landolt H, Hasler P (2016) Surgical versus conservative treatment for lumbar disc herniation: a prospective cohort study. *BMJ Open* 6(12). <https://doi.org/10.1136/BMJOPEN-2016-012938>
5. Wang Y, Ye F, Huang C, Xue F, Li Y, Gao S, Qiu Z, Li S, Chen Q, Zhou H, Song Y, Huang W, Tan W, Wang Z (2018) Bioinformatic analysis of potential biomarkers for spinal cord-injured patients with intractable neuropathic pain. *Clin J Pain* 34(9):825–830. <https://doi.org/10.1097/AJP.0000000000000608>
6. Lei Z, Cao H, Jie Y, Huang Z, Guo X, Chen J, Peng L, Cao H, Dai X, Liu J, Li X, Zhu J, Xu W, Chen D, Gao Z, He JR, Lin BL (2020) A cross-sectional comparison of epidemiological and clinical features of patients with coronavirus disease (COVID-19) in Wuhan and outside Wuhan, China. *Travel Med Infect Dis* 35:101664. <https://doi.org/10.1016/J.TMAID.2020.101664>
7. Zhu N, Zhang D, Wang W, Li X, Yang B, Song J, Zhao X, Huang B, Shi W, Lu R, Niu P, Zhan F, Ma X, Wang D, Xu W, Wu G, Gao GF, Tan W (2020) A novel coronavirus from patients with pneumonia in China 382(8):727–733. <https://doi.org/10.1056/NEJMOA2001017>
8. Attal N, Martinez V, Bouhassira D (2021) Potential for increased prevalence of neuropathic pain after the COVID-19 pandemic. *Pain Reports* 6(1). <https://doi.org/10.1097/PR9.0000000000000884>
9. Moisset X, Bouhassira D, Couturier JA, Alchaar H, Conradi S, Delmotte MH, Lanteri-Minet M, Lefaucheur JP, Mick G, Piano V, Pickering G (2020) Pharmacological and non-pharmacological treatments for neuropathic pain: systematic review and French recommendations. *Revue Neurologique* 176(5):325–352. <https://doi.org/10.1016/J.NEUROL.2020.01.361>
10. The Gene Expression Omnibus database (n.d.) Retrieved December 8, 2021, from <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4944384/>
11. R: The R Project for Statistical Computing (n.d.) Retrieved December 8, 2021, from <https://www.r-project.org/>
12. Ritchie ME, Phipson B, Wu D, Hu Y, Law CW, Shi W, Smyth GK (2015) limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Res* 43(7):e47–e47. <https://doi.org/10.1093/NAR/GKV007>
13. Smyth GK (2004) Linear models and empirical Bayes methods for assessing differential expression in microarray experiments. *Stat Appl Genet Mol Biol* 3(1). <https://doi.org/10.2202/1544-6115.1027/MACHINEREADABLECITATION/RIS>
14. Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, Davis AP, Dolinski K, Dwight SS, Eppig JT, Harris MA, Hill DP, Issel-Tarver L, Kasarskis A, Lewis S, Matese JC, Richardson JE, Ringwald M, Rubin GM, Sherlock G (2000) Gene ontology: tool for the unification of biology. *Nature Genet* 25(1):25–29. <https://doi.org/10.1038/75556>
15. Yu G, Wang L-G, Han Y, He Q-Y (2012) clusterProfiler: an R package for comparing biological themes among gene clusters 16(5):284–287. <https://doi.org/10.1089/OMI.2011.0118>, <https://Home.Liebertpub.Com/Omi>
16. Consortium GO (2006) The Gene Ontology (GO) project in 2006. *Nucleic Acids Res* 34(suppl_1):D322–D326. <https://doi.org/10.1093/NAR/GKJ021>
17. Home-Genes-NCBI (n.d.) Retrieved December 8, 2021, from <https://www.ncbi.nlm.nih.gov/gene/>
18. gplots package-RDocumentation (n.d.) Retrieved July 29, 2021, from <https://www.rdocumentation.org/packages/gplots/versions/3.1.1>

19. Package “pheatmap” (2019)
20. Neuwirth E (2014) ColorBrewer Palettes [R package RColorBrewer version 1.1–2]. <https://cran.r-project.org/package=RColorBrewer>
21. Pokhilko A, Nash A, Cader MZ (2020) Common transcriptional signatures of neuropathic pain. *Pain* 161(7):1542–1554. <https://doi.org/10.1097/J.PAIN.0000000000001847>
22. Zhang G, Yang P (2017) Bioinformatics genes and pathway analysis for chronic neuropathic pain after spinal cord injury. *BioMed Res Int.* <https://doi.org/10.1155/2017/6423021>
23. Sun W, Kou D, Yu Z, Yang S, Jiang C, Xiong D, Xiao L, Deng Q, Xie H, Hao Y (2017) A transcriptomic analysis of neuropathic pain in rat dorsal root ganglia following peripheral nerve injury. *NeuroMol Med* 22:250–263. <https://doi.org/10.1007/s12017-019-08581-3>
24. Ji F, Sadreyev RI (2018) RNA-seq: basic bioinformatics analysis. *Curr Protoc Mol Biol* 124(1):e68. <https://doi.org/10.1002/CPMB.68>
25. Huang R, Meng T, Zhu R, Zhao L, Song D, Yin H, Huang Z, Cheng L, Zhang J (2020) The integrated transcriptome bioinformatics analysis identifies key genes and cellular components for spinal cord injury-related neuropathic pain. *Front Bioeng Biotechnol* 10. <https://doi.org/10.3389/FBIOE.2020.00101>
26. Chen M, Zeng J, Yang Y, Wu B (2020) Diagnostic biomarker candidates for pulpitis revealed by bioinformatics analysis of merged microarray gene expression datasets. *BMC Oral Health* 20(1):1–13. <https://doi.org/10.1186/S12903-020-01266-5>
27. Wang Y, Dai G, Jiang L, Liao S, Xia J (2021) Microarray analysis reveals an inflammatory transcriptomic signature in peripheral blood for sciatica. *BMC Neurol* 21(1):1–11. <https://doi.org/10.1186/S12883-021-02078-Y>
28. Du Z, Yin S, Song X, Zhang L, Yue S, Jia X, Zhang Y (2020) Identification of differentially expressed genes and key pathways in the dorsal root ganglion after chronic compression. *Front Mol Neurosci* 71. <https://doi.org/10.3389/FNMOL.2020.00071>
29. Malcangio M (2019) Role of the immune system in neuropathic pain. *Scandinavian J Pain* 20(1). <https://doi.org/10.1515/SJPAIN-2019-0138>
30. Davies AJ, Rinaldi S, Costigan M, Oh SB (2020) Cytotoxic immunity in peripheral nerve injury and pain. *Front Neurosci* 14. <https://doi.org/10.3389/FNINS.2020.00142>
31. Campbell JN, Meyer RA (2006) Mechanisms of neuropathic pain. *Neuron* 52(1):77. <https://doi.org/10.1016/J.NEURON.2006.09.021>
32. Basbaum AI, Bautista DM, Scherrer G, Julius D (2009) Cellular and molecular mechanisms of pain. *Cell* 139(2):267. <https://doi.org/10.1016/J.CELL.2009.09.028>
33. Huang R, Meng T, Zha Q, Cheng K, Zhou X, Zheng J, Zhang D, Liu R (2021) The predicting roles of carcinoembryonic antigen and its underlying mechanism in the progression of coronavirus disease 2019. *Crit Care* 25(1):1–20. <https://doi.org/10.1186/S13054-021-03661-Y>
34. Hosomi S, Chen Z, Baker K, Chen L, Huang YH, Olszak T, Zeissig S, Wang JH, Mandelboim O, Beauchemin N, Lanier LL, Blumberg RS (2013) CEACAM1 on activated NK cells inhibits NKG2D-mediated cytolytic function and signaling. *Eur J Immunol* 43(9):2473–2483. <https://doi.org/10.1002/EJL.201242676>
35. Rajasekaran S, Soundararajan DCR, Nayagam SM, Tangavel C, Raveendran M, Thippeswamy PB, Djuric N, Anand SV, Shetty AP, Kanna RM (2021) Modic changes are associated with activation of intense inflammatory and host defense response pathways – molecular insights from proteomic analysis of human intervertebral discs. *Spine J.* <https://doi.org/10.1016/J.SPINEE.2021.07.003>
36. Seidel UJ, Schlegel P, Lang P (2013) Natural killer cell mediated antibody-dependent cellular cytotoxicity in tumor immunotherapy with therapeutic antibodies. *Front Immunol* 4(MAR):76. <https://doi.org/10.3389/fimmu.2013.00076>
37. Shiiba M, Saito K, Yamagami H, Nakashima D, Higo M, Kasamatsu A, Sakamoto Y, Ogawara K, Uzawa K, Takiguchi Y, Tanzawa H (2015) Interleukin-1 receptor antagonist (IL1RN) is associated with suppression of early carcinogenic events in human oral malignancies. *Int J Oncol* 46(5):1978–1984. <https://doi.org/10.3892/IJO.2015.2917>

38. Tai A, Zhu M, Qilimuge H, Rong H, He X, Bai M, Jin T (2020) Genetic polymorphisms of IL1RN were associated with lumbar disk herniation risk in a Chinese Han population. *Mol Genet Genomic Med* 8(6):1247. <https://doi.org/10.1002/MGG3.1247>
39. Zhou Z, Ren L, Zhang L, Zhong J, Xiao Y, Jia Z, Guo L, Yang J, Wang C, Jiang S, Yang D, Zhang G, Li H, Chen F, Xu Y, Chen M, Gao Z, Yang J, Dong J, Wang J (2020) Heightened innate immune responses in the respiratory tract of COVID-19 patients. *Cell Host Microbe* 27(6):883–890.e2. <https://doi.org/10.1016/J.CHOM.2020.04.017>
40. Bali A, Singh N, Jaggi AS (2014) Renin–angiotensin system in pain: existing in a double life? 15(4):329–340. <https://doi.org/10.1177/1470320313503694>
41. Kukkar A, Singh N, Jaggi AS (2013) Neuropathic pain-attenuating potential of aliskiren in chronic constriction injury model in rats. *JRAAS—J Renin-Angiotensin-Aldosterone Syst* 14(2):116–123. <https://doi.org/10.1177/1470320312460899>
42. Coto E, Avanzas P, Gómez J (2021) The Renin–Angiotensin–Aldosterone system and coronavirus disease 2019. *Euro Cardiol Rev* 16. <https://doi.org/10.15420/ECR.2020.30>
43. Ny L, Parmer RJ, Shen Y, Holmberg S, Baik N, Bäckman A, Broden J, Wilczynska M, Ny T, Miles LA (2020) The plasminogen receptor, Plg-R KT , plays a role in inflammation and fibrinolysis during cutaneous wound healing in mice. *Cell Death Disease* 11(12):1–15. <https://doi.org/10.1038/s41419-020-03230-1>
44. Cabrera-Garcia D, Miltiades A, Parsons S, Elisman K, Mansouri MT, Wagener G, Harrison NL (2021) High levels of plasminogen activator inhibitor-1, tissue plasminogen activator and fibrinogen in patients with severe COVID-19. *MedRxiv* 2020.12.29.20248869. <https://doi.org/10.1101/2020.12.29.20248869>
45. Baker SK, Strickland S (2020) A critical role for plasminogen in inflammation. *J Exp Med* 217(4). <https://doi.org/10.1084/JEM.20191865>
46. Baker SK, Chen Z-L, Norris EH, Strickland S (2019) Plasminogen mediates communication between the peripheral and central immune systems during systemic immune challenge with lipopolysaccharide. *J Neuroinflammat* 16(1):1–8. <https://doi.org/10.1186/S12974-019-1560-Y>
47. Mayi BS, Leibowitz JA, Woods AT, Ammon KA, Liu AE, Raja A (2021) The role of neuropilin-1 in COVID-19. *PLoS Pathog* 17(1):e1009153. <https://doi.org/10.1371/JOURNAL.PPAT.1009153>
48. Cantuti-Castelvetri L, Ojha R, Pedro LD, Djannatian M, Franz J, Kuivanen S, van der Meer F, Kallio K, Kaya T, Anastasina M, Smura T, Levanov L, Szirovicza L, Tobi A, Kallio-Kokko H, Österlund P, Joensuu M, Meunier FA, Butcher SJ, Simons M (2020) Neuropilin-1 facilitates SARS-CoV-2 cell entry and infectivity. *Science* 370(6518). <https://doi.org/10.1126/SCIENCE.ABD2985>
49. Tang X, Jing L, Chen J (2012) Changes in the molecular phenotype of nucleus pulposus cells with intervertebral disc aging. *PLoS ONE* 7(12):e52020. <https://doi.org/10.1371/JOURNAL.PONE.00520>

Political Optimizer Algorithm for Optimal Location and Sizing of Photovoltaic Distribution Generation in Electrical Distribution Network



D. Sreenivasulu Reddy , Varapasad Janamala ,
and Pappu Soundarya Lahari 

Abstract In this paper, the political optimizer (PO), a new and efficient socio-inspired meta-heuristic search algorithm, is proposed for the first time in this research for determining the ideal locations and capacities of photovoltaic (PV) distribution generation (DG) in electrical distribution networks (EDN). A multi-objective function is designed to lower distribution losses and voltage deviation indexes and maximize voltage stability, among other objectives. The computational efficiency of PO when solving the optimal allocation of PV systems in EDN is investigated on an IEEE 33-bus EDN. The results indicate that integrating small DGs at multiple locations has a better EDN performance than integrating a single significant DG in the network. The results also suggest that, as demonstrated by a comparative analysis of PO results and those of other related literature works, PO can deal with complex multi-variable optimization problems.

Keywords Electrical distribution network · Distribution generation · Photovoltaic system · Political optimizer · Loss minimization · Voltage profile · Voltage stability index

1 Introduction

De-carbonization is given higher priority than sustainability. Renewable energy (RE)-based distribution generation (DGs) is used to meet the load demand. Because the distribution system is a more complex network, it has higher power losses when compared to the transmission network, which leads to voltage instability, poor

D. Sreenivasulu Reddy (✉) · V. Janamala · P. S. Lahari
Department of Electrical and Electronics Engineering, School of Engineering and Technology,
Christ (Deemed to Be University), Bengaluru, Karnataka 560074, India
e-mail: sreenivasulu.d@res.christuniversity.in

V. Janamala
e-mail: varapasad.janamala@christuniversity.in

P. S. Lahari
e-mail: pappusoundarya.lahari@res.christuniversity.in

voltage regulation, and an increase in economic aspects. The potential benefits of DG with proper sizing and allocation in a network will include the following: improving voltage profile, reducing losses in a system, reliable supply, transmission system stress reduction, environmental friendliness, lower greenhouse gas emissions, improving power quality, and reduction in peak power requirements. By definition [1], DG is the generation of electricity near where it is used. This includes conventional-type generation and RE, mainly photovoltaic (PV) and wind turbines (WT) in small generation units. DG can also be called distributed energy, embedded generation, on-site generation, or district/decentralized energy.

In [2], loss minimization is achieved by optimizing the location and sizing of three solar PV systems using the Bat Algorithm (BA). In [3], the gbest-guided artificial bee colony (GABC) algorithm is proposed for solving the optimum allocation of different types of DGs considering different loading levels and different types of loads. The impact of DGs is analyzed for technical and economic benefits in distribution system operation. The author uses an improved differential search algorithm (IDSA) in [4] to select optimal sizes and locations for three different numbers of DGs, thereby minimizing power loss, improving voltage profile, and reducing economic aspects. In [5], a new bio-inspired algorithm, manta-ray foraging optimization (MRFO), is used, considering the reduction in losses as the primary objective by selecting an appropriate size and optimal placement. Simulation results of three different radial systems with three different numbers of DGs are implemented, and the results are compared with other heuristic techniques. In [6], a new meta-heuristic algorithm, the hybrid grey wolf optimizer, is used. The objective function is to reduce the losses and enhance the voltage profile on each bus. Results obtained are compared with the other algorithms and matched to get the best possible outcome. Different numbers of DGs are simulated on the IEEE-33, 69, and 85 bus systems. In [7], a constraint-based multi-objective approach using the comprehensive teaching learning-based optimization (CTLBO) algorithm is proposed to allocate multiple DGs and also for network loadability enhancement by reconfiguring them, reduction in active power losses, enhanced voltage profile, voltage stability index, and qualified loadability index under all constraints in different radial networks. In [8], multiple DG allocations with multiple objective functions are considered. A bio-inspired Cuckoo search algorithm is (CSO) used as an optimization technique that reduces power losses. The performance evaluation of the proposed system is done by comparing it with the PSO technique. In [9], the Whale Optimization Technique (WOT) is used to optimize multiple DG placements to reduce power loss, improve voltage profile, and increase economic benefits. The results are tested on two different radial systems compared with other heuristic methods. In [10], Stud Krill Algorithm (SKHA), a new meta-heuristic algorithm, is introduced for multiple DG allocation and sizing in two different radial systems and one real-time system for improving loss minimization and hence voltage stability achievement under various loading conditions. The performance evaluation compares the proposed method with other methods and proves that a considerable loss reduction has been obtained. The optimal allocation of the DGs problem is one of many meta-heuristic algorithms that have been proposed in the literature [11] for solving this problem.

The No-free-launch (NFL) theorem states that no such algorithm can solve all types of optimization problems [12]. As a result, researchers continue to be inspired to develop simple and efficient heuristic algorithms for solving optimization problems of various types. The Political Optimizer [13] is a type of recent socio-inspired meta-heuristic algorithm for global optimization problems, and it is one of several such algorithms. In this paper, PO is used for the first time to solve the DG allocation problem in the EDN. The primary objectives are to emphasize loss minimization, voltage deviation index minimization, and voltage stability enhancement.

2 Problem Formulation

The key operational concerns to optimize EDN operation and control are distribution losses, voltage profile, and stability. A multi-objective function is defined to optimize such performance indices, resulting in an efficient EDN.

$$f_1 = P_{\text{loss}} = \sum_k I_k^2 r_k \tag{1}$$

$$f_2 = \text{AVDI} = \frac{1}{nb} \sum_{nb} (|V_r| - |V_i|)^2 \tag{2}$$

$$f_3 = \frac{1}{\text{VSI}_j} = \frac{1}{|V_j^4| - 4(P_j x_k - Q_j r_k) - 4(P_j r_k + Q_j x_k) |V_i^2|} \tag{3}$$

$$\text{OF} = \min\{f_1 + f_2 + f_3\} \tag{4}$$

The suggested multi-objective function is solved with the voltage, current, and DGs power restrictions in mind.

$$|V_i|_{\min} \leq |V_i| \leq |V_i|_{\max} \quad i = 1, 2, \dots, nb \tag{5}$$

$$|I_i| \leq |I_i|_{\max} \quad i = 1, 2, \dots, nbr \tag{6}$$

$$\sum_{i=1}^{ndg} P_{dg(i)} \leq P_D \tag{7}$$

where f_1, f_2 , and f_3 define the distribution losses, voltage profile, voltage stability of the EDN, and overall objective function, respectively. I_k, r_k , and x_k are the current flow, resistance, and reactance of a branch- k connected between bus- i and bus- j , respectively; V_i and V_j are the voltage magnitudes of reference and other buses, respectively; the real and reactive power loads of bus j are denoted by

P_j and Q_j , respectively. $P_{dg}(i)$, PD is the DG real power and total real power demand, respectively; ndg , nb , and nbr are the number of DGs, buses, and branches, respectively.

3 Political Optimizer

The construction of this newly designed meta-heuristic political optimizer (PO) [13] was inspired by strategic political games during the election period in western countries. The heuristic method can depict the following by breaking the entire election process into crucial phases: (i) party formation and constituency allocation represent population initialization; (ii) election campaign represents exploration and exploitation phases; (iii) interparty election, party switching represents balancing between exploration and exploitation phases, including population updating; and (iv) election representation. PO is a highly competitive optimization strategy for solving nonlinear, complex optimization problems in real-time engineering.

The PO addresses the process of nomination by the number of political parties (n), with its number of candidates (n) in a constituency and the same process in several constituencies/dimensions (d) simultaneously in the first phase, as with any heuristic method for population initialization.

$$M = \{M_1, M_2, \dots, M_i, \dots, M_n\} \tag{8}$$

$$M_i = \{m_i^1, m_i^2, \dots, m_i^j, \dots, m_i^n\} \tag{9}$$

$$m_i^j = \{m_{i,1}^1, m_{i,2}^2, \dots, m_{i,j}^j, \dots, m_{i,d}^n\}^T \tag{10}$$

Now, there could be n election candidates representing various political parties for each electoral district, with the j th candidate representing a single solution, as proposed.

$$K = \{K_1, K_2, \dots, K_j, \dots, K_n\} \tag{11}$$

$$K_j = \{m_1^j, m_2^j, \dots, m_j^j, \dots, m_n^j\} \tag{12}$$

The best fitness candidate for a given party can serve as the constituency’s leader and is assessed by,

$$m_i^* = m_i^q \quad \text{where} \quad q = \underset{1 \leq j \leq n}{\operatorname{argmin}} f(m_i^j), \forall i = \{1, \dots, n\} \tag{13}$$

Following the election phase, each constituency’s party leaders can now be chosen as a solution vector, as specified by

$$m^* = \{m_1^*, m_2^*, \dots, m_i^*, \dots, m_n^*\} \tag{14}$$

Members of parliament are elected from a variety of constituencies and are classified as

$$K^* = \{k_1^*, k_2^*, \dots, k_j^*, \dots, k_n^*\} \tag{15}$$

The exploration and exploitation phases are developed in the second phase using the strategies employed by each candidate during the election campaign. When fitness improves, Eq. (16) is used to update the variables; Eq. (17) is used

$$m_{i,k}^j(t+1) = \begin{cases} l^* + r(l^* - m_{i,k}^j(t)) & \text{if } m_{i,k}^j(t-1) \leq m_{i,k}^j(t) \leq l^* \text{ or } \\ & m_{i,k}^j(t-1) \geq m_{i,k}^j(t) \geq l^* \\ l^* + (2r-1)|l^* - m_{i,k}^j(t)| & \text{if } m_{i,k}^j(t-1) \leq l^* \leq m_{i,k}^j(t) \text{ or } \\ & m_{i,k}^j(t-1) \geq l^* \geq m_{i,k}^j(t) \\ l^* + (2r-1)|l^* - m_{i,k}^j(t-1)| & \text{if } l^* \leq m_{i,k}^j(t-1) \leq m_{i,k}^j(t) \text{ or } \\ & l^* \geq m_{i,k}^j(t-1) \geq m_{i,k}^j(t) \end{cases} \tag{16}$$

Both are updating the position refer to the party leader (m_i^*) and the constituency winner (k_j^*). Here, a random number with a uniform distribution in the range [0, 1] l^* is used to address the party leader position first and then the constituency winner position.

$$m_{i,k}^j(t+1) = \begin{cases} l^* + (2r-1)|m^* - m_{i,k}^j(t)| & \text{if } m_{i,k}^j(t-1) \leq m_{i,k}^j(t) \leq l^* \text{ or } \\ & m_{i,k}^j(t-1) \geq m_{i,k}^j(t) \geq l^* \\ m_{i,k}^j(t-1) + r(m_{i,k}^j(t) - m_{i,k}^j(t-1)) & \text{if } m_{i,k}^j(t-1) \leq l^* \leq m_{i,k}^j(t) \text{ or } \\ & m_{i,k}^j(t-1) \geq l^* \geq m_{i,k}^j(t) \\ l^* + (2r-1)|l^* - m_{i,k}^j(t-1)| & \text{if } l^* \leq m_{i,k}^j(t-1) \leq m_{i,k}^j(t) \text{ or } \\ & l^* \geq m_{i,k}^j(t-1) \geq m_{i,k}^j(t) \end{cases} \tag{17}$$

The third phase proposes a proper balance between exploration and exploitation using the scenario of an electoral candidate switching parties. While party switching may occur concurrently with the election campaign during generation, it is implemented in PO after the election campaign phase through an adaptive parameter called the party switching rate (λ), which is defined as decreasing linearly from one to zero throughout the iteration phase.

$$q = \underset{1 \leq j \leq n}{\operatorname{argmax}} f(m_r^j) \quad (18)$$

The fourth stage maps the election phase to calculate the fitness value using winners from all constituencies, as specified by,

$$q = \underset{1 \leq j \leq n}{\operatorname{argmin}} f(m_i^j) \quad , \text{ where } k_j^* = m_q^j \quad (19)$$

In the fifth phase, the exploitation and convergence stages are modeled using Eqs. (13) and (14) to create a synergistic scenario of parliamentary affairs following elections (19). When the winner of the j th constituency (k_j^*) improves its fitness, it updates its position and fitness by selecting the reference of a random parliamentarian (k_r^*). Additionally, it causes the position of that parliamentarian (k_j^*) in the vector of winning party members (M_i) to be updated appropriately.

PO distinguishes itself by emulating this simple strategic gaming for election victory, and it has been developed so that it is suitable for determining a global solution.

4 Results and Discussion

The proposed methodology's effectiveness is evaluated using the widely used IEEE 33-bus system. The proposed PO was implemented in MATLAB on a PC with an Intel Core i5-4210U processor running at up to 1.7 GHz and 8 GB of RAM. The simulations are run for four different scenarios: (1) no DGs, (2) optimal allocation of one PV type DG, (3) optimal allocation of two PV type DGs, and (4) optimal allocation of three PV type DGs. For all the case studies, the maximum number of iterations = 50, number of populations/dimensions = 50, number of political parties/search variables = $2 \times$ number of DG's (one for location and another for size), maximum limit of party fixing rate/lambda = 1.

Case 1: The test system has a base caseload of 3715 kW and 2300 kVAr and suffers from distribution losses of 210.9983 kW + j 143.0329 kVAr. The overall VSI of the system is 0.666, and it has the minimum voltage magnitude at bus-18 of 0.904 p.u.

Case 2: The optimal PV location is bus-6, and correspondingly, the optimal size is 2588 kW, respectively. The EDN performance is changed as follows: losses are decreased to 111.0299 kW + j 81.6838 kVAr. The overall VSI of the system is raised to 0.7926, and it has the minimum voltage magnitude at bus-33 of 0.9435 p.u.

Case 3: The optimal PV locations are bus-13, and bus-30, and correspondingly, the optimal sizes are 854 kW, and 1170 kW, respectively. The EDN performance is changed as follows: losses are decreased 85.708 kW + j 58.908 kVAr. The overall

VSI of the system is raised 0.884, and it has the minimum voltage magnitude at bus-33 of 0.9697 p.u.

Case 4: The optimal PV location is bus-13, bus-24, and bus-30, and correspondingly, the optimal sizes are 800 kW, 1069 kW, and 1064 kW, respectively. The EDN performance is changed as follows: losses are decreased to 71.841 kW + j 50.0287 kVAr. The overall VSI of the system is raised to 0.8831, and it has the minimum voltage magnitude at bus-33 of 0.9694 p.u.

The results of PO are compared with literature for one, two, and three PV type DGs in Tables 1, 2, and 3, respectively. From these results, PO outperformed GABC [3], MRFO [5], MRFO [5], CTLBO [7], HGWO [6], WOA [9], SKHA [10], KHA [10], PFA [14] algorithms.

Table 1 Comparison of PO results for one PV allocation

Method	Locations (bus #)	Size (kW)	P _{loss} (kW)	Q _{loss} (kVAr)	V _{min} (p.u)	AVDI	VSI
Base			210.9983	143.0329	0.9039	0.0546	0.6675
GABC [3]	30	1543	125.15	87.734	0.927	1.1791	0.740
MRFO [5]	6	2590.2	111.021	81.266	0.9427	0.0277	0.7898
CTLBO[7]	8	3609.8	192.72	155.393	0.977	0.0038	0.8797
HGWO [6]	6	2590	111.018	81.2661	0.9427	0.0277	0.7898
WOA [9]	6	2589.6	111	81.69	0.9424	0.0277	0.7898
SKHA [10]	6	2590.215	111.018	81.716	0.9424	0.0277	0.7898
KHA [10]	6	2590.216	111.018	81.716	0.9424	0.0277	0.7898
PFA[14]	6	2590.264	111.03	81.684	0.9424	0.0280	0.7684
Proposed PO algorithm	6	2588	111.0299	81.6838	0.9435	0.0272	0.7926

Table 2 Comparison of PO results for two PV allocation

Method	Locations (bus #)		Size (kW)		P _{loss} (kW)	Q _{loss} (kVAr)	V _{min} (p.u)	AVDI	VSI
Base					210.548	142.7439	0.9039	0.0546	0.6675
GABC [3]	29	30	1015	626	119.5937	86.6530	0.9294	0.0321	0.7462
MRFO [5]	13	30	851.51	1158	85.7163	58.879	0.9693	0.020	0.8828
CTLBO[7]	15	30	1430	2568	86.576	66.509	0.9861	0	0.9456
HGWO [6]	13	30	852	1158	85.7163	58.879	0.9693	0.02	0.8828
SKHA [10]	13	30	851.63	1157.58	87.165	59.812	0.9685	0.02	0.8828
KHA [10]	13	29	824.49	1241.71	87.426	60.209	0.9667	0.02	0.8761
Proposed PO algorithm	13	30	854	1170	85.708	58.908	0.9697	0.0198	0.8840

Table 3 Comparison of PO results for three PV allocation

Method	Locations (bus #)			Size (kW)			P_{loss} (kW)	Q_{loss} (kVAr)	V_{min} (p.u)	AVDI	VSI
Base							210.548	142.743	0.9039	0.0546	0.6675
GABC [3]	28	29	30	1098	132	609	95.29	83.125	0.9629	0.6977	0.8596
SKHA [10]	13	24	30	1054	1091	802	75.507	52.240	0.9647	0.0164	0.8660
IDSA [4]	13	25	30	800	1037	969	73.321	51.010	0.9664	0.0194	0.8722
CTLBO[7]	13	16	30	967.7	563.7	2508	83.897	61.97	0.9863	0.5302	0.9465
BA [2]	13	24	30	720	1020	980	72.781	50.625	0.9638	0.0211	0.8628
WOA [9]	13	25	30	857	772	1073	72.664	50.304	0.9696	0.0179	0.8838
MRFO [5]	13	24	30	788	1017	1035	71.964	50.046	0.9682	0.0190	0.8786
CSA [8]	14	24	30	750	1100	1050	71.891	50.079	0.9686	0.0189	0.8801
HGWO [6]	13	24	30	802	1090	1054	71.845	50.033	0.9694	0.0182	0.8831
Proposed PO algorithm	13	24	30	800	1069	1064	71.841	50.0287	0.9694	0.0182	0.8831

On the other hand, compared to the base case, one PV installation at bus-6 resulted in an $x\%$ loss reduction, two PV DGs caused $y\%$ loss reduction, and three PV DGs resulted in $z\%$ loss reduction. Similarly, the voltage deviation index also decreases for the number of PV DGs, from a, b, c, and d, respectively. On the other hand, the voltage stability index improves from a to b, c, and d with no DG, one DG, two DGs, and three DGs, respectively. From this, one can say that the EDN performance improves in all aspects by having dispersed DGs with small-scale levels at as many locations as possible instead of a large DG at one place. Thus, determining the optimal number of DGs is also a critical issue and can extend the research problem in this area.

5 Conclusion

Using a new and efficient socio-inspired meta-heuristic search method, the political optimizer (PO) determines PV distribution generation’s ideal locations and capacities in electrical distribution networks. A multi-objective function reduces distribution losses and the voltage deviation index while increasing voltage stability. This work analyzes the computing efficiency of PO while determining the optimal PV system allocation in an EDN. In the study, smaller distributed generation units (DGs) performed better than larger distributed generation units (DGs) at different locations. Comparing PO’s results to other similar literature shows that PO can handle challenging multi-variable optimization problems of a complicated nature.

References

1. Pepermans G et al (2005) Distributed generation: definition, benefits, and issues. *Energy Policy* 33(6):787–798
2. Yuvaraj T, Devabalaji KR, Ravi K (2018) Optimal allocation of DG in the radial distribution network using a bat optimization algorithm. In: *Advances in power systems and energy management*. Springer, Singapore, pp 563–569
3. Dixit M, Kundu P, Jariwala HR (2017) Incorporation of distributed generation and shunt capacitor in the radial distribution system for techno-economic benefits. *Int J Eng Sci Technol* 20(2):482–493
4. Injeti SK (2018) A Pareto optimal approach for allocation of distributed generators in radial distribution systems using an improved differential search algorithm. *J Electr Syst Inform Technol* 5(3):908–927
5. Hemeida MG et al (2021) Optimal allocation of distributed generators DG based Manta-Ray Foraging Optimization algorithm (MRFO). *Ain Shams Eng J* 12(1):609–619
6. Sanjay R et al (2017) Optimal allocation of distributed generation using hybrid grey wolf optimizer. *IEEE Access* 5:14807–14818
7. Quadri IA, Bhowmick S, Joshi D (2018) Multi-objective approach to maximize loadability of distribution networks by simultaneous reconfiguration and allocation of distributed energy resources. *IET Gener Trans Distrib* 12(21):5700–5712
8. Yuvaraj T, Ravi K (2018) Multi-objective simultaneous DG and DSTATCOM allocation in radial distribution networks using the cuckoo searching algorithm. *Alex Eng J* 57(4):2729–2742
9. Prakash DB, Lakshminarayana C (2018) Multiple DG placements in the radial distribution system for multi objectives using Whale Optimization Algorithm. *Alex Eng J* 57(4):2797–2806
10. ChithraDevi SA, Lakshminarasimman L, Balamurugan R (2017) Stud Krill herd algorithm for multiple DG placement and sizing in a radial distribution system. *Int J Eng Sci Technol* 20(2):748–759
11. Sambaiah KS, Jayabarathi T (2020) Loss minimization techniques for optimal operation and planning of distribution systems: A review of different methodologies. *Int Trans Electr Energy Syst* 30(2):e12230
12. Adam SP et al (2019) No free lunch theorem: a review. *Approx Optim*, pp 57–82
13. Askari Q, Younas I, Saeed M (2020) Political optimizer: a novel socio-inspired meta-heuristic for global optimization. *Knowl-Based Syst* 195:105709
14. Janamala V (2021) A new meta-heuristic pathfinder algorithm for solving optimal allocation of solar photovoltaic system in the multi-lateral distribution system for improving resilience. *SN Appl Sci* 3(1):1–17

Cyberbullying Detection in Social Media Using Supervised ML and NLP Techniques



Karthiga Shankar, A. M. Abirami, K. Indira, C. V. Nisha Angeline, and K. Shubhavya

Abstract With the increase in the number of social media applications, cyberbullying is growing day by day. Though individuals are expressively attached in out of community Internet platform, these chances stance a major threat, such as cyberattacks, including cyberbullying. In the existing system, we have more machine learning models to overcome the cyberattacks, even though it does not hold all the security metrics. To overcome this problem, we have developed a machine learning model to identify abusive and non-abusive tweets with the source of Twitter data. Consequently, we also established a model that automatically detects bullying on the social media application. There are many ways to detect bullying on social media, but many are text-based and give minimal accuracy on the Twitter dataset. In this system, we have explored about 20,001 unique tweets from Twitter. Finally, in the proposed model, the data is given into two classifiers such as SVM and Naive Bayes which are used to train and test destructive social media content. Experimental results are achieved in a better manner to detect the unwanted tweets from the application. The proposed model is a better model that gives good quality precision, and thus, this model can be used for future works.

Keywords SVM · Naïve Bayes · Accuracy

1 Introduction

Community Internet sites are being extensively utilized at present for numerous functions like entertainment, networking, user interaction, etc. Community Internet platform involves the authority of the contributing people. Connecting sideways individuals is no exemption. As knowledge has changed the approach, individuals relate with a broader manner and have given a new component to transmission. Several individuals are illegitimately utilizing the neighborhoods.

K. Shankar (✉) · A. M. Abirami · K. Indira · C. V. N. Angeline · K. Shubhavya
Department of Information Technology, Thiagarajar College of Engineering, Madurai, India
e-mail: skait@tce.edu

Cyberbullying is the consumption of knowledge to threaten somebody. Community Internet sites provide a creative medium for people who employ these sites to attack or bully other powerless and young users. Tormenters use several essential services like Twitter, Facebook, and email to intimidate other individuals. Cyberbullying is one of the greatest problems come about Internet exploitation and a major public trouble particularly for adolescent people. Consequently, scientists are dedicating on how to uncover and avoid the occurrence of cyberbullying, especially in community mass media. Cyberbullying is not simply restricted to making a pretend personality and advertising/displaying some uncomfortable photo or video, disagreeable stories about someone but also openhanded threats to others. The impacts of cyberbullying on social media are alarming, sometimes leading to the death of some unfortunate victims.

Even though individuals are sensitively linked all as one through social media, these services take down huge pressures with them such as cyberattacks, which involve cyberbullying. Thus, we are in need of a complete machine learning model to analyze the cyberattacks. Presently, social media is an essential component of regular lifetime. Nevertheless, young people's handling of expertise, as well as social media, can show them too many social and emotional consequences. One of these risks is cyberbullying, which is a prominent social attack occurring on social media platforms. The challenge can be faced by identifying and avoiding it by using a machine learning method, and this need to be done using a distinct viewpoint. Cyberbullying occurrences are growing day by day as knowledge moves out. Many cyberbullying instances are described by companies each year. Many machine learning models not actually classify and predict the tweets which are displayed in the social media. Even though there are numerous systems, they use only labeled data which has been previously given in the dataset. However, at times the dataset may well give way to create precise outcome of identified information.

1.1 Objective

The primary purpose is to find out abusive posts that seem to be degrading for society and people. We are competent to effectively alter the offenses that are discriminating in the online platform. The provisions of this article include

1. To build a machine learning model to predict the data trustworthiness in the social media application.
2. To overcome the prevailing model performance issues, classifier is used to evaluate the system effectiveness and is added with the framework for normal and abusive tweets.

2 Related Work

Currently, there has been a growth in the intensity of cyber actions by youngsters, particularly on community Internet sites, which consistently subjected them to cyberbullying. Data Handling is very important and this will influence the attitude of people, discourage the person, and can take the lead them to decline the action. Many machine learning models and cybersecurity team researchers worked on it, and some of the works are cited: The framework is developed that uses the detecting offensive language in social media to guard the protection of elder people. Consumer-point rudeness discovery appears a more viable methodology [1]. We recognize the impact of expletives and obscenities in verifying attack matter and establish a small hand-authoring syntactic rules in identifying name-calling harassments. Cyberbullying system is created with data of the user arguments for tagging is employed with crowdsourcing, catching the true-time situation of intentionally abusive words or sort words [2]. Intelligent system is developed, an experimental study to establish the efficiency and implementation of deep learning algorithms in exposing community analysis. The numerous models were tested with experimental results of tokenization and streaming. Experimental solutions given as deep learning model are ready to provide good accuracy in the proposed system [3].

Security would be aligned with AI automated model, and this is the next generation of attack system with soft computing methods [4]. The work represents a deep neural model for cyberbullying detection with active steering for guessing the verbatim bullying substance and a model for calculating the graphic stalking question. Investigational assessment is made on a mix-modal data which covers 9 K statements along with columns argued with good AU_ROC value [5]. Model determination enhances exercise information and utilizes a uncertain SVM. Experimental results showed that fuzzy-based SVM approach had great accuracy to detect the data. Framework is established with administered model key aimed at cyberbullying discovery and classification in the community Internet. Experimental results are shown that the semi-supervised model gives best f1-measure in the cybersecurity domain [6].

The framework has developed a successful method to identify cyber aggressive and infrastructures through joining data domain. Individual features, such as handbag of key arguments (BoA), are utilized and analyze the machine learning model [7]. Two approaches are created to develop ensemble supporting model to classify the contents as “offensive” or “non-offensive” with most excellent implementation of 96% when TF-IDF (Unigram) feature extraction included with K-fold cross-validation [8]. The new framework is evolved based on the conditions and usage of text and its linguistic natures in the proposed model with improved performance metrics [9].

3 Proposed Model

The proposed model is introduced to eliminate any shortcomings in the existing system. When classifying data, the system improves the accuracy of the monitored classification results. A method for detecting and preventing cyberbullying on Twitter is proposed by the help of sentimental analysis and natural language processing. Corpus bag of words is used as a preprocessing tool to remove punctuation and other unnecessary notations. Our prototypical is assessed on support vector machines and Naive Bayes methods, thereby showing best efficiency among other system.

3.1 Methodology

Toward classify resemblances model created by intimidators and c may grow a cubic centimeter perfect mechanically find community internet users. Biased sentimentality cuts to the words inside sentences enforced supervised algorithms to enhances the presentation of the arrangement results.

Figure 1 describes the system architecture of the proposed work. The data collected is firstly preprocessed, and the sentiment analysis is done on the vectorized preprocessed data. The data is then split into train and test and is fed into models of support vector machine and Naïve Bayes. The trained model is then used for prediction, and several parameters are used for evaluation.

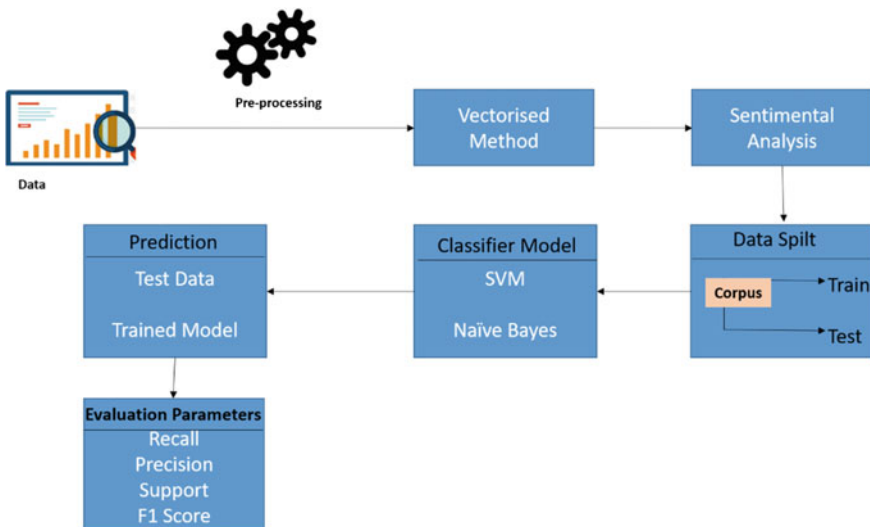


Fig. 1 System architecture

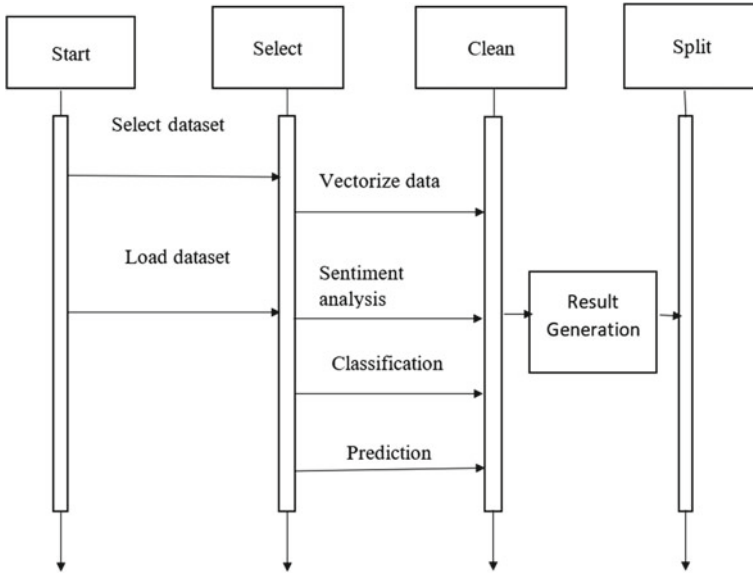


Fig. 2 Sequence diagram

Figure 2 describes the proposed model as a sequence diagram. The dataset is selected and loaded, the preprocessed data is vectorized, then, sentiment analysis is done on the vectorized data, and after classification using SVM and Naïve Bayes, the prediction is made.

4 Implementation

4.1 Proposed Modules

Data Selection and Loading

The data assortment is the development of choosing the data for noticing occurrences. Cyberbullying information is hand-me-down for noticing aggressive and non-attacking information which contains data about the person designation and concern label. Tweets can be of any form, and there is a vital role for the data selection part since the unwanted data may cause the improper results at the end.

Information Preprocessing

Information processing is the development of eliminating the annoying statistics after the concern. They are of two types:

Splitting the dataset

Data splitting is the act of partitioning the data into portions. This is used to expand a prognostic version and to assess the algorithm routine.

Feature extraction

Feature grading is a technique to regulate the variety of self-governing variable quantity or structures of information.

Classification

Gaussian Naïve Bayes cares incessant appreciated landscapes and copies each as complaint to a Gaussian (normal) delivery.

A support vector machine (SVM) model is fundamentally a depiction of dissimilar courses in a hyperactive flat in multi-dimensional space.

Calculation

This is used to achieve the bottom most mistake likely by moreover using “boosting” or “bagging”.

Evaluation Metrics

The result will be produced on overall organization and forecast.

The quality measures of this work are as follows:

The benefits of RS can be determined by passing the attributes into high-quality experiments. The performance of the proposed method is usual. Accuracy, in terms of statistics, will compare the predicted ratings with actual ratings in user-item combination, and it would also utilize mean absolute error (MAE). It can be evaluated as follows:

$$MAE = \sum_{i=1}^U \sum_{j=1}^I \frac{|P_{i,j} - r_{i,j}|}{U * I} \quad (1)$$

Precision is defined as the fraction of recommended items that are relevant to user’s interest.

$$Precision = \frac{|{C} \cap {RI}|}{|{C}|} \quad (2)$$

Here, C defines the recommended items, and R refers to relevant items.

Recall is defined as the fraction of relevant items that are recommended.

$$Recall = \frac{|{D} \cap {RI}|}{|{RI}|} \quad (3)$$

5 Experimental Results

5.1 Outcome of Proposed Work

The proposed system helps us to detect the cyberbullying attacks in Twitter using the supervised binary classification machine learning algorithms. The system will increase the accuracy of the supervised classification results by classifying the data. We have developed an approach toward the detection of cyberbullying behavior. It will detect the abusive and non-abusive posts in the tweets by which we can effectively deal with the crimes that are committed using these platforms. Results will show the accuracy for detecting cyberbullying content has also been greater than the performance of SVM. In future, it is possible to provide extensions or modifications to the proposed clustering and classification algorithms to achieve further increased performance. Apart from the experimented combination of data mining techniques, further combinations and other clustering algorithms can be used to improve the detection accuracy and to reduce the rate offensive tweets. Finally, the cyberbullying detection system can be extended as a prevention system to enhance the performance of the system.

Figure 3 shows the pie chart that shows the distribution of tweets as abusive and non-abusive in the dataset.

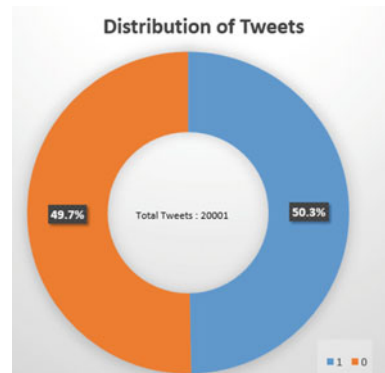
Testing and analysis

Our proposed model is trained and tested and is implemented using two algorithms, namely SVM and Naïve Bayes to check the accuracy of our model, and we find the model that is better in accuracy.

Figure 4 describes the dataset that is extracted using Twitter API. It has 200,001 sentiments.

Various preprocessing is done in the dataset to make the implementation easy by removing stop words, emojis, repeated characters, etc., since they do not have any importance in the output.

Fig. 3 Pie chart showing distribution of tweets



Index	Unnamed: 0	annotation	content	user
0	0	{'notes': '', 'label': ['1'...	Get fucking real dude.	scotthamilton
1	1	{'notes': '', 'label': ['1'...	She is as dirty as they...	mattycus
2	2	{'notes': '', 'label': ['1'...	why did you fuck it up. I...	ElleCTF
3	3	{'notes': '', 'label': ['1'...	Dude they dont finish e...	Karoli
4	4	{'notes': '', 'label': ['1'...	WTF are you talking about...	joy_wolf
5	5	{'notes': '', 'label': ['1'...	Ill save you the trouble s...	mybirch
6	6	{'notes': '', 'label': ['1'...	Im dead serious.Real ...	coZZ
7	7	{'notes': '', 'label': ['1'...	...go absolutely in...	2Hood4Hollywo...
8	8	{'notes': '', 'label': ['1'...	Lmao im watching the ...	mimismo
9	9	{'notes': '', 'label': ['1'...	LOL no he said What do...	erinx3leannexo
10	10	{'notes': '', 'label': ['1'...	truth on both counts that g...	pardonlauren
11	11	{'notes': '', 'label': ['1'...	Shakespeare nerd!	TLeC
12	12	{'notes': '', 'label': ['1'...	you are SUCH a fucking dork	robrobberobe...
13	13	{'notes': '', 'label': ['1'...	Heh. Fuck 'em WHERE?!?	bayofwolves
14	14	{'notes': '', 'label': ['1'...	damn it i totally forgo...	HairByJess
15	15	{'notes': '', 'label': ['1'...	wow damn I would have be...	lovesongwriter

Fig. 4 Dataset

Figure 5 shows the results of the sentiment analysis performed on the data. Here, four columns are added that indicate the polarity of a single tweet. The positive, negative and neutral scores are calculated using the polarity score of each word. If the word has positive value, it is considered as a positive comment, and a negative score describes a negative score.

Figure 6 depicts the compound score column that is consolidated from the positive, negative and neutral scores. This compound score is thus the metric to describe the tweet as either a positive or a negative tweet. The compound score column has two values:

1. If the compound value is a negative value, the compound score is 1 which depicts that the tweet is a negative tweet.
2. If the compound value is zero or a positive value, the compound score is 0 which means the tweet is positive.

Figure 7 is the classification report for showing the variations of values of tweet using SVM and Naïve Bayes algorithms.

Figure 8 shows the histogram chart that describes the variations of different values ranging between zero and one. The chart contains the number of tweets that are positive, negative and neutral between zero and one.

Figure 9 shows the accuracy comparison between SVM and Naïve Bayes algorithms, and we can see that SVM model produces better results.

Index	Unnamed: 0	content	user	compound	neg	neu	pos
0	0	Get fucking real dude.	scotthamilton	0	0	1	0
1	1	She is as dirty as they...	mattycus	0.2023	0.098	0.743	0.159
2	2	why did you fuck it up. I...	ElleCTF	-0.5423	0.108	0.892	0
3	3	Dude they dont finish e...	Karoli	-0.7322	0.245	0.755	0
4	4	WTF are you talking about...	joy_wolf	-0.7739	0.341	0.659	0
5	5	Ill save you the trouble s...	mybirch	-0.8271	0.383	0.514	0.103
6	6	Im dead serious.Real ...	coZZ	-0.7918	0.307	0.613	0.08
7	7	...go absolutely in...	2Hood4Hollywo...	-0.7096	0.237	0.763	0
8	8	Lmao im watching the ...	mimismo	0.7947	0.108	0.608	0.284
9	9	LOL no he said what do...	erinx3leannexo	0.7798	0.091	0.578	0.331
10	10	truth on both counts that g...	pardonlauren	-0.296	0.141	0.766	0.093
11	11	Shakespeare nerd!	TLeC	-0.3595	0.714	0.286	0
12	12	you are SUCH a fucking dork	robobbierobe...	-0.4005	0.35	0.65	0
13	13	Heh. Fuck 'em WHERE!?!?	bayofwolves	-0.6958	0.742	0.258	0
14	14	damn it i totally forgo...	HairByJess	-0.4574	0.333	0.667	0
15	15	wow damn I would have be...	lovesongwriter	-0.4767	0.413	0.359	0.228
16	16	nigga u geigh lmao! fuck yo...	armotley	0.1759	0.256	0.438	0.306
17	17	that sucks :(starkissed	-0.6597	0.844	0.156	0
18	18	read that this morning. mv f...	gi_gi_bee	0.5994	0	0.726	0.274

Fig. 5 Sentiment analysis performance on the data

Index	Unnamed: 0	content	user	compound	neg	neu	pos	comp_score
0	0	Get fucking real dude.	scotthamilton	0	0	1	0	0
1	1	She is as dirty as they...	mattycus	0.2023	0.098	0.743	0.159	0
2	2	why did you fuck it up. I...	ElleCTF	-0.5423	0.108	0.892	0	1
3	3	Dude they dont finish e...	Karoli	-0.7322	0.245	0.755	0	1
4	4	WTF are you talking about...	joy_wolf	-0.7739	0.341	0.659	0	1
5	5	Ill save you the trouble s...	mybirch	-0.8271	0.383	0.514	0.103	1
6	6	Im dead serious.Real ...	coZZ	-0.7918	0.307	0.613	0.08	1
7	7	...go absolutely in...	2Hood4Hollywo...	-0.7096	0.237	0.763	0	1
8	8	Lmao im watching the ...	mimismo	0.7947	0.108	0.608	0.284	0
9	9	LOL no he said what do...	erinx3leannexo	0.7798	0.091	0.578	0.331	0
10	10	truth on both counts that g...	pardonlauren	-0.296	0.141	0.766	0.093	1
11	11	Shakespeare nerd!	TLeC	-0.3595	0.714	0.286	0	1
12	12	you are SUCH a fucking dork	robobbierobe...	-0.4005	0.35	0.65	0	1
13	13	Heh. Fuck 'em WHERE!?!?	bayofwolves	-0.6958	0.742	0.258	0	1
14	14	damn it i totally forgo...	HairByJess	-0.4574	0.333	0.667	0	1
15	15	wow damn I would have be...	lovesongwriter	-0.4767	0.413	0.359	0.228	1

Fig. 6 Compound score of the sentiment analysis

```
IPython console
Console 1/A
Naive Bayes
-----Classification Report-----
              precision    recall  f1-score   support

     0       0.79      0.85      0.82     2350
     1       0.85      0.80      0.83     2651

   micro avg       0.82      0.82      0.82     5001
   macro avg       0.82      0.82      0.82     5001
weighted avg       0.83      0.82      0.82     5001

-----Accuracy-----
The Accuracy Score :82.0

-----
Support vector Machine

-----Classification Report-----
              precision    recall  f1-score   support

     0       0.89      0.88      0.89     2538
     1       0.88      0.89      0.88     2463

   micro avg       0.89      0.89      0.89     5001
   macro avg       0.89      0.89      0.89     5001
weighted avg       0.89      0.89      0.89     5001

-----Accuracy-----
The Accuracy Score :89.0
```

Fig. 7 Classification report of SVM and Naïve Bayes

6 Conclusion and Future Works

Community Internet sites. If we can successfully detect such posts which are not suitable for adolescents or teenagers, we can very effectively deal with the crimes that are committed using these platforms. An approach is proposed for detecting and preventing Twitter cyberbullying using supervised binary classification machine learning algorithms. Our model is evaluated on both support vector machine and Naive Bayes, and also for feature extraction, we used the TFIDF vectorizer. As the results show us that the accuracy for detecting cyberbullying content has also been great for support vector machine which is better than Naive Bayes. Our model will help people from the attacks of social media bullies. In future, it is possible to provide extensions or modifications to the proposed clustering and classification algorithms to achieve further increased performance. Apart from the experimented combination of data mining techniques, further combinations and other clustering algorithms can be used to improve the detection accuracy and to reduce the rate of offensive tweets. Finally, the cyberbullying detection system can be extended as a prevention system to enhance the performance of the system.

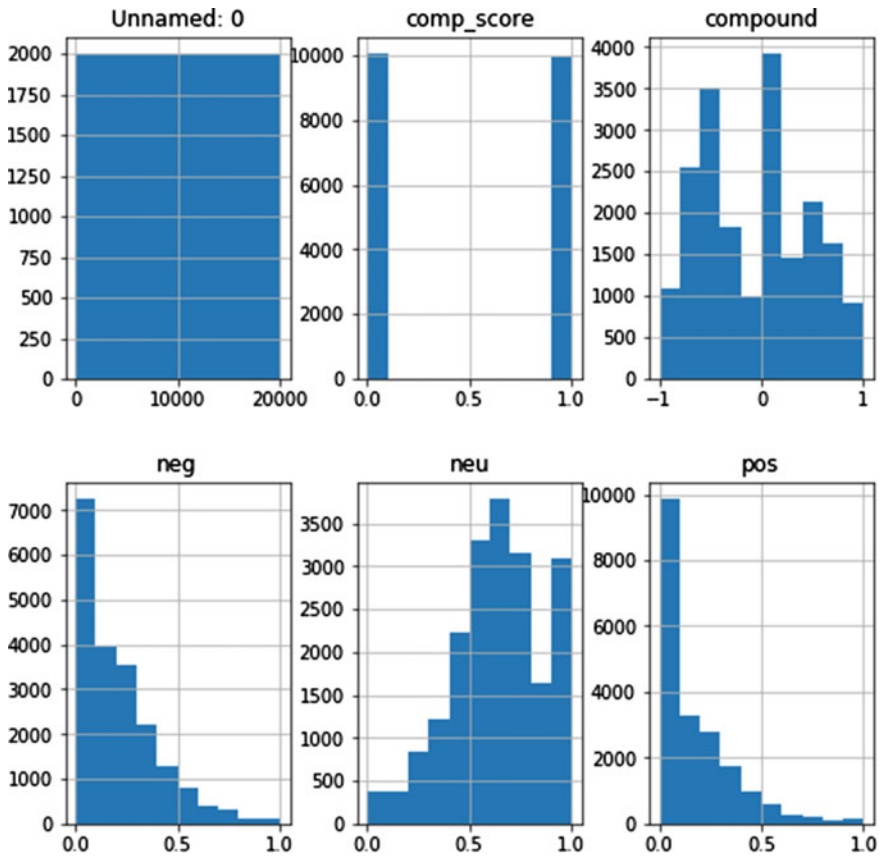
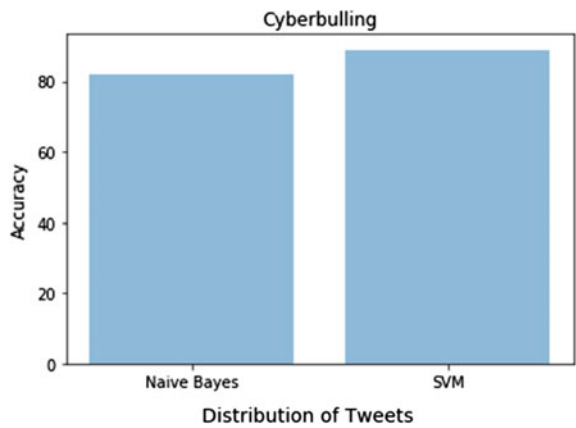


Fig. 8 Histogram showing the variations of values for each tweet

Fig. 9 Comparison of accuracy between SVM and Naive Bayes



References

1. Singh A, Kaur M (2019) Content-based cybercrime detection: a concise review. *Int J Innov Technol Explor Eng (IJITEE)* 8(8):1193–1207. ISSN: 2278-3075
2. Gupta A et al (2012) Characterizing pedophile conversations on the internet using online grooming. *ArXiv abs/1208.4324*: n. page 23
3. Kumar A, Sachdeva N (2021) Multimodal cyberbullying detection using capsule network with dynamic routing and deep convolutional neural network. *Multimedia Syst.* <https://doi.org/10.1007/s00530-020-00747-5>
4. Chen Y, Zhou Y, Zhu S, Xu H (2020) Detecting offensive language in social media to protect adolescent online safety. In 2012 International conference on privacy, security, risk and trust (PASSAT) and 2012 International conference on social computing (SocialCom)
5. Perera A, Fernando P (2021) Accurate cyberbullying detection and prevention on social media. *Proc Comput Sci* 181:605–611. <https://doi.org/10.1016/j.procs.2021.01.207>
6. Iwendi C, Srivastava G, Khan S et al (2020) Cyberbullying detection solutions based on deep learning architectures. *Multimedia Syst.* <https://doi.org/10.1007/s00530-020-00701-5>
7. Talpur BA, O’ Sullivan D (2020) Cyberbullying severity detection: a machine learning approach. *PLoS ONE* 15(10):e0240924. <https://doi.org/10.1371/journal.pone.0240924>
8. Islam MM, Uddin MA, Islam L, Akter A, Sharmin S, Acharjee UK (2020) Cyberbullying detection on social networks using machine learning approaches. In: 2020 IEEE Asia-Pacific conference on computer science and data engineering (CSDE), pp 1–6. <https://doi.org/10.1109/CSD E50874.2020.9411601>
9. Alam KS, Bhowmik S, Prosun PRK (2021) Cyberbullying detection: an ensemble based machine learning approach. In: 2021 Third international conference on intelligent communication technologies and virtual mobile networks (ICICV), pp 710–715. <https://doi.org/10.1109/ICICV5 0876.2021.9388499>

Investigating the Positioning Capability of GPS and Galileo Constellations Over Indian Sub-continent



Devadas Kuna  and Naveen Kumar Perumalla

Abstract The Galileo is a global navigation satellite system (GNSS) available to all users as a civilian system, and it has been developed by the European Union (EU). All the designed Galileo satellites are currently deployed in space, and the constellation is in validation phase with initial positioning assistance. Currently, the 22 Galileo satellites enable stand-alone Galileo positioning, which allows users to evaluate its positioning performance. Not much significant research has been reported regarding comparative performance analysis of GPS and Galileo constellations for the Indian region. In this assessment, GPS positions were determined by online positioning user service-static (OPUS-S) and are used to evaluate the Galileo positioning capability with broadcast ephemeris. This work uses a typical period of one week of continuously acquired data from 28th July to 3rd August 2019. The results show that the average number of visible satellites are 7 for Galileo and 10 for GPS at the Hyderabad Station ($17^{\circ} 24' 28.40137''$ N, $78^{\circ} 31' 4.41613''$ E). Galileo's stand-alone mean position at the user's location is $X = 1,211,941.93$ m, $Y = 5,966,420.66$ m, and $Z = 1,896,089.58$ m. In comparison with the GPS precise position ($X = 1,211,941.49$ m, $Y = 5,966,419.39$ m, and $Z = 1,896,089.21$ m), Galileo's X , Y , and Z values match well. The comparative numerical analysis shows that Galileo provides acceptable navigation solutions similar to GPS solutions determined by OPUS-S. The fully operational Galileo constellation may offer more than currently available satellites to users and improve accuracy and availability in low-latitude regions.

Keywords Accuracy · DRMS · Satellite visibility

1 Introduction

The Global Navigation Satellite System (GNSS) is becoming a vital positioning technology across various services [1]. Currently, only two satellite navigation systems are operational with global coverage for military and civilian applications: The United

D. Kuna (✉) · N. K. Perumalla

Advanced GNSS Research Laboratory, Department of Electronics and Communication Engineering, University College of Engineering, Osmania University, Hyderabad, India
e-mail: devadaskuna@osmania.ac.in

States GPS and the Russian-developed Globalnaja Nawigazionnaja Sputnikowaja Sistema (GLONASS). The Global European Galileo and Chinese BeiDou constellations, the regional Indian Regional Navigation Satellite System (IRNSS), and the Japanese Quasi-Zenith Satellite System (QZSS) are among the emerging navigation satellite constellations [2]. These systems help to compute the position, velocity, and time (PVT) of the end user's location using geodetic receivers [3, 4].

Unlike GPS, the EU states that Galileo guarantees to be available during hostile situations amid public and commercial services [5]. The Walker 24/3/1 design of Galileo comprises 24 satellites and 6 spares which are equally spaced in three medium Earth orbit (MEO) planes. The orbital altitude of Galileo satellites is about 23,222 km which is quite a bit higher than that of the GPS and GLONASS satellites. The 56° of orbital inclination and its altitude ensures improved Galileo satellite signal coverage in the polar regions (higher latitudes), which are poorly served by GPS [6]. Currently, it is not yet fully operational, but the initial services were started in December 2018. Besides GPS and GLONASS, Galileo is expected to introduce new modernization elements around 2021 [7]. At present, the constellation has 22 healthy operational satellites, including 19 fully operational capability (FOC) and 3 in-orbit validation (IOV) satellites [8]. In this paper, 22 MEO Galileo constellation is considered, as it seems to be the most likely choice, and it will be compared with the GPS constellation as it existed with 30 MEOs unequally spaced in 6 orbital planes. These systems provide worldwide coverage using similar frequencies, as L1–E1 and L5–E5a frequencies. Over low-latitude regions, all of the current GPS and the Galileo satellites are visible.

Due to its favorable geographic location, India has an opportunity to receive signals from all GNSS constellations and therefore is well situated to take advantage of this multi-GNSS environment. There has been extensive research on using multiple GNSS components in stand-alone and hybrid GNSS operation studies from other parts of the world. Several Indian researchers have examined GLONASS and GPS [9, 10]. Indian researchers have reported a few results on Galileo with only four Galileo IOV satellites [11] and Galileo satellite visibility [12]. As per the literature review, not much significant research has been done regarding GPS and Galileo comparative performance evaluation over the Indian region. Hence, it is necessary to analyze the GPS and Galileo systems to assess their performance for future navigation applications.

2 Methodology

To assess the stochastic properties of the stand-alone Galileo performance, the data files recorded in receiver independent exchange (RINEX) 3.0× formatted files with an interval of 60 s were collected from 28th July to 3rd August 2019. Data were acquired with the high-sensitivity multi-GNSS triple frequency NovAtel GPStation-6 receiver at 5° of mask angle. The antenna (choke ring, GPS-703-GGG) receives data from L1, L2, and L5 frequencies and is connected to the receiver via a 30 m RF

cable. This device is installed at “Advanced GNSS Research Laboratory (AGRL), Department of Electronics and Communication Engineering, Osmania University, Hyderabad, India.” Here, the receiver installed site has no proneness to multi-path, and it is continuously operational on a 24×7 basis.

In the GPS and Galileo comparative performance evaluation context, the availability and accuracy parameters are considered. In this paper, availability is used to refer to the good number of satellite visibility available to the user. The skyplot is one of the finest tools which can be used for displaying GNSS satellite trajectories over a particular ground location straightforwardly. This plot provides a visual representation of the visible satellite geometry. Accuracy measures the navigation solution’s consistency to the user’s actual location with varying satellite visibility. The distance root mean square (DRMS) and its doubled value (2DRMS) are frequently used to quantify the navigation performance of GNSS [13] statistically. Both have a confidence region of around 65% and 95%, respectively, with the receiver’s reference position being the center. This confidence region’s radius corresponds to the precision of the receiver’s position fix. The DRMS can be determined by taking the square root of the receiver’s latitude and longitude mean squares.

The navigation solution’s accuracy depends upon the satellite visibility and the precise measurements. The broadcast ephemeris (BE) quality has a high impact on navigation solutions. Due to a lack of precise ephemeris (PE) availability, most of the navigation receivers use only their BE measurements to compute navigation solutions. The GPS solutions are estimated with PE measurements provided by the National Oceanic and Atmospheric Administration (NOAA), whereas Galileo uses its own receiver generated BE measurements. Since the OPUS solutions are in the International Terrestrial Reference Frame (ITRF) and the difference between the Galileo Terrestrial Reference Frame (GTRF) and ITRF is less than 3 cm (2σ), it is pretty minor and negligible in comparison [14]. This comparison helps to assess the potentiality of the Galileo solutions, and how well they are near the precise GPS solutions under low-latitude regions. It may be beneficial to further research, how Galileo integrates with GPS and other global and regional constellations across low-latitude regions since the integration will likely be able to improve receiver positioning and increase the number of satellites available for future civilian applications.

2.1 Receiver ECEF Position Estimation Principle

Usually, the receiver’s location is determined only by satellites visible above its horizon. In practice, GNSS receivers rely on satellite elevation masks and the principle of time of arrival (TOA) ranging to determine the time taken by a signal transmitted by a satellite at a known location to reach the receiver. The receiver estimates the position coordinates by combining the raw measurements from the visible satellites and the orbital data from the broadcast satellite. Receivers use satellite navigation messages with orbital data to estimate satellite positions. Here, the satellite Earth-centered Earth-fixed (ECEF) coordinates are computed using the algorithm

Table 1 Satellite ECEF position computation algorithm

$GM_e = 3.986,005 \times 10^{14} \text{ m}^3/\text{s}^2$	Earth's gravitational constant
$\omega_e = 7.2921151467 \times 10^{-5} \text{ rad/sec}$	Earth rotation rate
$a = (\sqrt{a})^2$	Semi-major axis
$T = 2\pi / \sqrt{\frac{GM}{A^3}}$	Satellite orbital period
$n_0 = \sqrt{\frac{GM}{A^3}}$	Computed mean motion (rad/sec) using Kepler's 3rd law
$n = n_0 + \Delta n$	Corrected mean motion
$t_k = t - t_{0e}$	Time according to t_{0e}
$\bar{M}_k = \bar{M}_0 + n t_k$	Mean anomaly
$E_k = \bar{M}_k + e \sin E_k$	Eccentric anomaly
$\cos v_k = \frac{\cos E_k - e}{1 - e \cos E_k}$	True anomaly cosine term
$\sin v_k = \frac{\sqrt{1-e^2} \sin E_k}{1 - e \cos E_k}$	True anomaly sine term
$v = 2 \tan^{-1} \left(\sqrt{\frac{1+e}{1-e}} \tan \frac{E_k}{2} \right)$	True anomaly
$\delta u_k = C_{uc} \cos 2\varphi_k + C_{us} \sin 2\varphi_k$	Argument of latitude
$u_k = \varphi_k + \delta u_k$	The argument of latitude correction
$\delta r_k = C_{rc} \cos 2\varphi_k + C_{rs} \sin 2\varphi_k$	Radius correction
$i_k = i_0 + \delta t_k + \hat{i} t_k$	Inclination correction
$u_k = \varphi_k + \delta u_k$	The corrected argument of latitude
$r = A(1 - e \cos E_k)$	Radius, r
$r_k = A(1 - e \cos E_k) + \delta r_k$	Corrected radius
$i_k = i_0 + \delta t_k + \hat{i} t_k$	Corrected inclination
$\Omega_k = \Omega_0 + (\dot{\Omega} - \omega_e)t_k - \omega_e t_{0e}$	Corrected longitude of ascending node
$X_k = X'_k \cos \Omega_k - Y'_k \sin \Omega_k \cos i_k$	Earth-fixed geocentric satellite Coordinates
$Y_k = X'_k \sin \Omega_k + Y'_k \cos \Omega_k \cos i_k$	
$Z_k = Y'_k \sin i_k$	

stated as in Table 1 [15]. The least square (LS) approach estimated receiver position through RINEX formatted navigation and observation files corresponding to satellite visibility and their positions as depicted in a flowchart (see Fig. 1).

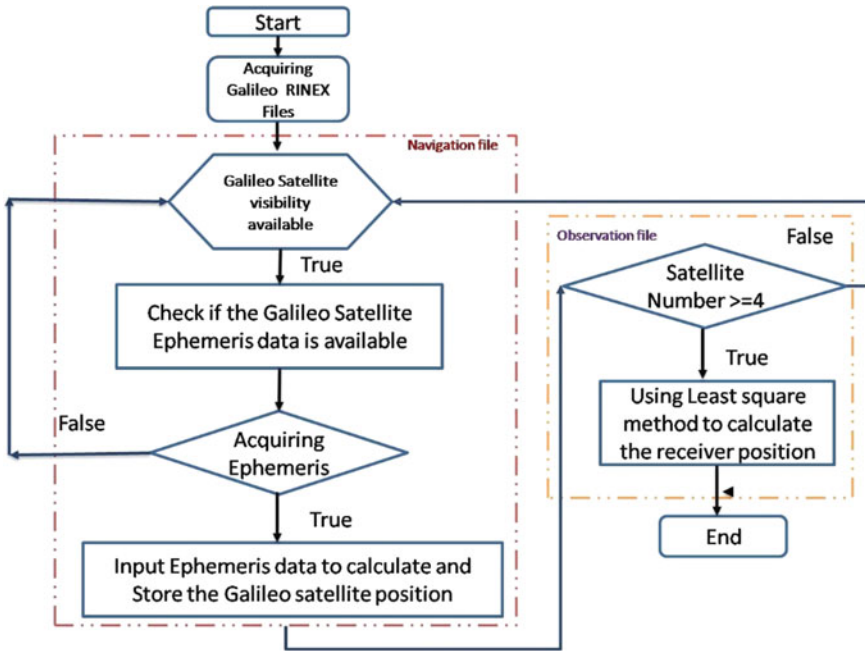


Fig. 1 Flowchart representation for satellite and receiver position calculation from Galileo RINEX files

Let us consider the ECEF satellite position vector for k th satellite is s_k , receiver position vector is u , and the pseudorange ρ_i between receiver and satellite can be represented as

$$\rho_i = \|s_k - u\| + ct_u \tag{1}$$

If (x^k, y^k, z^k) are k th satellite coordinate and (x_u, y_u, z_u) are receiver coordinates, Eq. (1) can be written as

$$\rho_i = \sqrt{(x^k - x_u)^2 + (y^k - y_u)^2 + (z^k - z_u)^2} + ct_u \tag{2}$$

Based on Eq. (2), more satellite visibility causes more equations, but it is possible to estimate receiver position with a minimum of four satellites.

If $(\Delta x_u, \Delta y_u, \Delta z_u)$ is the deviation between the actual position (x_u, y_u, z_u) and the approximate position $(\hat{x}_u, \hat{y}_u, \hat{z}_u)$, then according to the Taylor series expansion, it can be expressed as a function of pseudorange measurements and the satellite known coordinates as

$$\rho_k - \hat{\rho}_k = \frac{x_k - \hat{x}_u}{\hat{r}_k} \Delta x_u + \frac{y_k - \hat{y}_u}{\hat{r}_k} \Delta y_u + \frac{z_k - \hat{z}_u}{\hat{r}_k} \Delta z_u - ct_u \tag{3}$$

where $\hat{r}_k = \sqrt{(x^k - \hat{x}_u)^2 + (y^k - \hat{y}_u)^2 + (z^k - \hat{z}_u)^2}$.

Now Eq. (3) can be written as

$$\Delta\rho_k = a_{xk}\Delta x_u + a_{yk}\Delta y_u + a_{zk}\Delta z_u - c\Delta t_u \tag{4}$$

Here, the $\Delta x_u, \Delta y_u, \Delta z_u$ and Δt_u are four unknown variables if signals are directly acquired from visible satellites for $k \geq 4$. The following equations are enough to solve four unknowns:

$$\begin{aligned} \Delta\rho_1 &= a_{x1}\Delta x_u + a_{y1}\Delta y_u + a_{z1}\Delta z_u - c\Delta t_u \\ \Delta\rho_2 &= a_{x2}\Delta x_u + a_{y2}\Delta y_u + a_{z2}\Delta z_u - c\Delta t_u \\ \Delta\rho_3 &= a_{x3}\Delta x_u + a_{y3}\Delta y_u + a_{z3}\Delta z_u - c\Delta t_u \\ \Delta\rho_4 &= a_{x4}\Delta x_u + a_{y4}\Delta y_u + a_{z4}\Delta z_u - c\Delta t_u \end{aligned} \tag{5}$$

So, the delta pseudorange can be solved by using Eq. (5) as

$$\Delta\rho = H\Delta x \tag{6}$$

where $\Delta\rho = \begin{bmatrix} \Delta\rho_1 \\ \Delta\rho_2 \\ \Delta\rho_3 \\ \Delta\rho_4 \end{bmatrix}$, $H = \begin{bmatrix} a_{x1} & a_{y1} & a_{z1} & 1 \\ a_{x2} & a_{y2} & a_{z2} & 1 \\ a_{x3} & a_{y3} & a_{z3} & 1 \\ a_{x4} & a_{y4} & a_{z4} & 1 \end{bmatrix}$ and $\Delta x = \begin{bmatrix} \Delta x_u \\ \Delta y_u \\ \Delta z_u \\ -c\Delta t_u \end{bmatrix}$.

Finally, Δx is simply resolved as

$$\Delta x = H^{-1}\Delta\rho \tag{7}$$

Here, $(\Delta x_u, \Delta y_u, \Delta z_u)$ is solved, and the approximate position $(\hat{x}_u, \hat{y}_u, \hat{z}_u)$ is known, so the actual receiver coordinates (x_u, y_u, z_u) are attained with a sum of the above two positions. This linearization approach is only viable if the displacement is close to the linearization point; otherwise, it must be repeated until the accurate location is restricted to the user’s precision [16].

3 Results and Discussion

There is no specific time series for GNSS static analysis in the case of satellite navigation systems. It is critical to compute as many latitude and longitude observations from the acquired data to obtain more accurate statistical estimates. As a result, accuracy estimation outcomes will be better averaged over long-term processed samples than short-term processed samples. The Galileo and the GPS measurements are collected on similar days and under similar conditions from 28th July to 3rd August 2019. The Galileo ECEF coordinates are estimated using broadcast measurements

with a well-known LS algorithm in MATLAB. The ECEF coordinates of the receiver position are calculated using the mean values of all the estimated user positions per day and the corresponding standard deviation values (STD) of X , Y , and Z directions as shown in Table 2. The receiver reference position is estimated by the nominal average of all seven days computed positions in ECEF coordinates as 1,211,941.93, 5,966,420.664, and 1,896,089.584 m.

The estimated Galileo ECEF coordinates were compared with the online positioning user service-static (OPUS-S) determined GPS coordinates to assess the stand-alone Galileo position accuracy. Here, keeping the observed session periods (T) longer than 4 h is valuable with OPUS-S. It is noticed that, the outcomes of OPUS-S depend upon the mean of three specified CORS single-baseline solutions [17]. The 24 h duration GPS observation file (20,642,120.19°) of 31st July 2019 is processed on OPUS-S with its simple user interface. As a result, OPUS-S returned an email that included a positioning solution report without any warning messages in the ITRF-2014 (in the text, this solution is referred to as OPUS2021). Due to occasional gaps in available data at requested CORS, it is found that only 53,410 (96%) solutions were utilized over 55,788 submitted RINEX observations. As per OPUS2021, the precise position of the receiver in ECEF coordinates is $X = 1,211,941.49$ m, $Y = 5,966,419.39$ m, and $Z = 1,896,089.21$ m. Similarly, the seven successive days analysis is carried out to determine precise GPS receiver positions as $17^\circ 24' 28.39''$ N and $78^\circ 31' 4.41''$ E in degrees, minutes, and seconds (DMS). The entire analysis is using near CORS base stations like CHUM, IISC, POL2, and BHR4. From Tables 2 and 3, it is observed that the GPS has much higher accuracy characteristics than the Galileo. The Galileo navigation solutions rely on broadcast ephemeris, whereas GPS relies on precise measurements. The GPS PE measurements allow for a greater receiver position precision than the Galileo BE measurements, impacting the 2D horizontal accuracy (i.e., 2-dimensional positioning points). Here, the averaged STD values of receiver ECEF coordinates are 2.294, 5.600, 2.426 (m), and 0.007, 0.016, 0.012 (m) for Galileo and GPS, respectively. The DRMS and 2DRMS of position error for Galileo at Hyderabad station are 6.051 m and 12.103 m, respectively. The typical dual-frequency Galileo horizontal positioning accuracy with 99.5% availability where 24 operational satellites are used is ≤ 10 m [18]. Due to the 22 operational satellites available over the observed period, the averaged 2DRMS (95%) for Galileo is ≤ 12 m, nearly higher than the typical value over the low-latitude region. In the case of PE GPS, the average DRMS and 2DRMS (95%) are 1.7 cm and 3.4 cm, respectively, with 99.5% availability which is much better than the typical value of 31 operational satellites [19]. Hence, as expected, the precise measurements give accurate positions rather than the broadcast measurements.

Figure 2 shows the bar graph of receiver ECEF coordinates with STD variation for GPS PE and Galileo BE. The deviation between PE GPS and BE Galileo's estimated ECEF coordinates and satellite visibility for the observed period is shown in Table 4. It is observed that, from day to day, the deviations of ΔX , ΔY , and ΔZ vary between 0 and 3 m. It indicates that the BE Galileo estimated position determines much nearer positioning solutions compared to PE GPS observables. The maximum, minimum, and average number of Galileo and GPS visible satellites for each observed day

Table 2 Computed receiver ECEF coordinates with standard deviation (STD) values and accuracy measures using Galileo data from 28th July to 3rd August 2019

Sessions	Mean observations				Accuracy measures			
	ECEF-X (m)	ECEF-Y (m)	ECEF-Z (m)	$\sigma_X (m)$	$\sigma_Y (m)$	$\sigma_Z (m)$	DRMS (65%)	2DRMS (95%)
Day 1	12,119 42.265	5,966,422.644	1,896,090.344	2.374	4.475	2.640	5.065	10.131
Day 2	12,119 41.002	5,966,420.609	1,896,088.862	1.975	3.300	2.055	3.845	7.691
Day 3	12,119 42.274	5,966,419.923	1,896,089.834	2.717	6.600	2.534	7.137	14.274
Day 4	12,119 41.241	5,966,419.635	1,896,089.532	2.585	7.640	2.486	8.065	16.130
Day 5	12,119 42.114	5,966,419.980	1,896,089.410	1.685	4.215	1.394	4.539	9.078
Day 6	12,119 42.421	5,966,421.779	1,896,089.614	1.579	5.835	3.068	6.044	12.089
Day 7	12,119 42.186	5,966,420.076	1,896,089.489	3.145	7.136	2.810	7.798	15.596
Avg	12,119 41.93	5,966,420.664	1,896,089.584	2.294	5.600	2.426	6.051	12.103

Table 3 Computed receiver ECEF coordinates with standard deviation (STD) values and accuracy measures using GPS data on OPUS-S service from 28th July to 3rd August 2019

Sessions	Mean observations						$\sigma_X (m)$	$\sigma_Y (m)$	$\sigma_Z (m)$	RMS (95%)	2DRMS S (95%)
	ECEF-X (m)	ECEF-Y (m)	ECEF-Z (m)								
Day 1	1,211,941,489	5,966,419,399	1,896,089,214	0.012	0.010	0.015	0.015	0.031			
Day 2	1,211,941,492	5,966,419,402	1,896,089,215	0.008	0.010	0.013	0.012	0.025			
Day 3	1,211,941,493	5,966,419,397	1,896,089,215	0.004	0.010	0.012	0.010	0.021			
Day 4	1,211,941,489	5,966,419,401	1,896,089,216	0.007	0.025	0.011	0.025	0.051			
Day 5	1,211,941,491	5,966,419,405	1,896,089,219	0.018	0.041	0.020	0.044	0.089			
Day 6	1,211,941,492	5,966,419,397	1,896,089,212	0.005	0.012	0.011	0.013	0.026			
Day 7	1,211,941,494	5,966,419,394	1,896,089,210	0.004	0.003	0.007	0.005	0.010			
Avg	1,211,941,492	5,966,419,399	1,896,089,215	0.007	0.016	0.012	0.017	0.034			

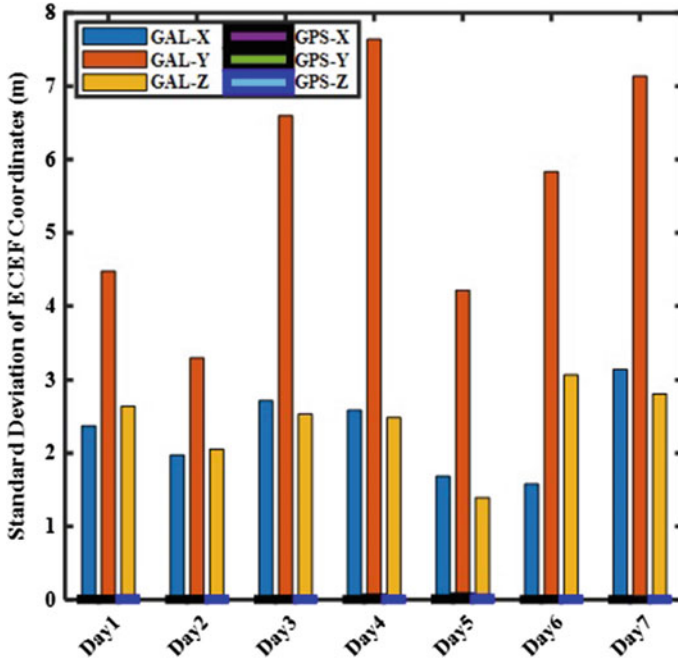


Fig. 2 Relative standard deviation (STD) of receiver ECEF coordinates (m) between precise GPS and broadcasted Galileo measurements at Hyderabad station from 28th July to 3rd August 2019

Table 4 Difference between GPS, Galileo ECEF receiver coordinates, and minimum and maximum satellite visibility over observed station from 28th July to 3rd August 2019

Sessions	ΔX	ΔY	ΔZ	GPS satellite visibility		Galileo satellite visibility	
				Min	Max	Min	Max
Day 1	-0.776	-3.245	-1.13	7	12	4	9
Day 2	0.49	-1.207	0.353	6	12	5	10
Day 3	-0.781	-0.526	-0.619	7	12	5	9
Day 4	0.248	-0.234	-0.316	7	12	5	9
Day 5	-0.623	-0.575	-0.191	7	12	4	10
Day 6	-0.929	-2.382	-0.402	5	12	5	9
Day 7	-0.692	-0.682	-0.279	7	12	5	10
Avg	-0.438	-1.265	-0.369				

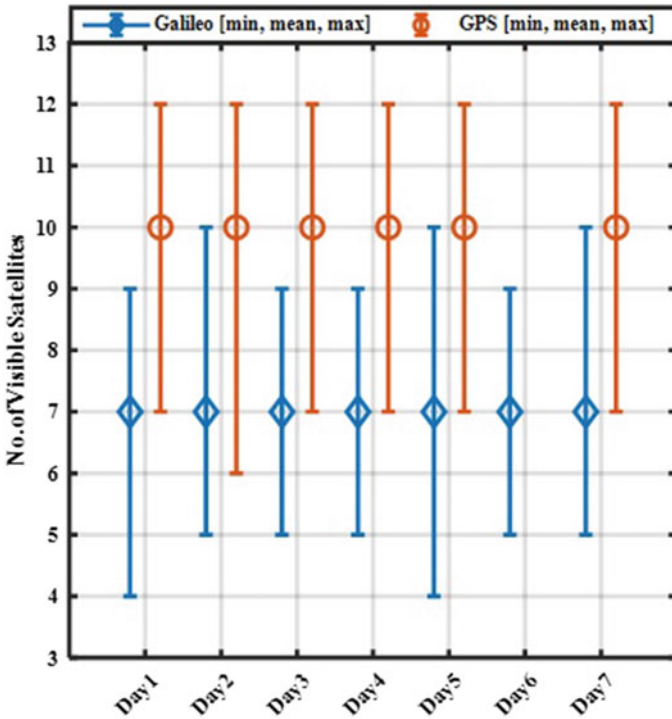


Fig. 3 Minimum, maximum, and average number of Galileo and GPS satellites in view per day are related to Hyderabad station from 28th July to 3rd August 2019

are shown in Fig. 3. The satellite visibility measurements show that a substantial number of satellites for both systems are visible. On an average, 7 Galileo and 10 GPS satellites always seem visible for users at the observed location. It is worth noting that four Galileo satellites are simultaneously accessible for users under an open sky across the observed area. As a result, this may enable a stand-alone Galileo system or complement multi-GNSS operations. Figure 4 shows the distribution of the GPS and Galileo satellites directly above the observed point as a function of elevation and azimuth angles. The skyplot illustrates the wide distribution of 8 Galileo and 12 GPS satellites over Hyderabad station with 5° of mask angle for a selected time interval on 2nd August 2019. According to the entire observations, all the GPS PE observations are nearly centered to zero with better satellite visibility (i.e., more accurate). Galileo BE observations demonstrate the good nature of positioning solutions related to satellite visibility for the observed period.

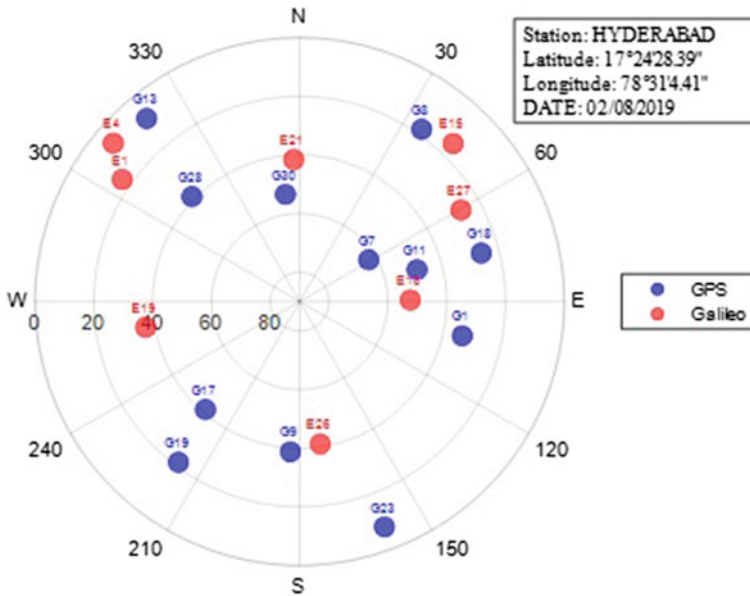


Fig. 4 Skyplot observation of GPS and Galileo satellites over the horizon of the receiver at 5° mask angle on day of the year (DOY) –214 at 00:00:42 h

4 Conclusion

Performance assessment of Galileo’s initial operational capability approach is presented and discussed in this paper. Galileo’s potentiality is compared with that of GPS using satellite visibility, receiver position estimation, and horizontal accuracy based on data from the Hyderabad station. The stand-alone Galileo X, Y, Z coordinates are computed and compared with precise GPS observables. The accuracy of Galileo BE measurements shows discrete behavior rather than the PE GPS accuracy. Here, the availability of PE measurements of GPS determines better accuracy than Galileo. During the seven-day observation period, 95% of navigation solutions are constrained to a radius of 12.10 m and 0.034 m, respectively, for Galileo and GPS. The Galileo satellites vary between 4 and 10, enabling autonomous positioning, and there is no vicinity gap between the Galileo satellites and their observed location.

Regarding the results, both GPS and Galileo constellations offer the best availability of the navigation solution over the experimental period with better satellite visibility. Based on all the comparative numerical results, it can be concluded that the stand-alone Galileo constellation enables autonomous position determination with a good number of visible satellites over the Hyderabad station. After becoming fully operational with a fully deployed designed constellation, the accuracy and availability

of stand-alone Galileo are predicted to improve significantly similar to existing fully functional GPS.

Acknowledgements The authors would like to convey their heartfelt appreciation to the National Geodetic Survey's Online Positioning User Service for making the OPUS Program available for worldwide usage. The authors would like to thank TEQIP-II for assisting them in procuring the GPStation-6 receiver for the Advanced GNSS Research Laboratory (AGRL), Department of Electronics and Communication Engineering, University College of Engineering, Osmania University, Hyderabad, India.

References

1. Parkinson BW, Spilker JJ (eds) (1996) Global positioning system: theory and applications, progress in astronautics and aeronautics. AIAA, Washington DC
2. Hofmann-Wellenhof B, Lichtenegger H, Wasle E (2008) GNSS—global navigation satellite systems: GPS, GLONASS, Galileo & more. Springer, Berlin
3. Naveen Kumar P, Naga Sahitya P (2017) Comparative analysis of accuracy using stand-alone GPS and GAGAN systems for Indian region. *Int J Electron Electr Comput Syst (IJEECS)* 6(7):581–586 ISSN 2348-117X
4. Kuna D, Perumalla NK, Anil Kumar R (2020) Positioning parameters for stand-alone and hybrid modes of the Indian NavIC system: preliminary analysis. In: Satapathy SC et al (eds) International conference on emerging trends in engineering (ICETE). Springer Nature Switzerland AG 2020, ICETE 2019, LAIS 4, pp 1–8
5. Rao GS (2010) Global navigation satellite systems. Tata McGraw-Hill Education
6. Beitz JW (2016) Engineering satellite-based navigation and timing global navigation satellite systems, signals, and receivers. Wiley, Hoboken, NJ
7. [https://en.wikipedia.org/wiki/Galileo\(satellite_navigation\)](https://en.wikipedia.org/wiki/Galileo(satellite_navigation)). Last accessed 21 Aug 2021
8. European GNSS Service Centre, Constellation Information. <https://www.gsc-europa.eu/system-service-status/constellation-information>. Last accessed 22 Sept 2021
9. Banerjee P, Bose A, Das gupta A (2002) The usefulness of GLONASS for positioning in the presence of GPS in Indian subcontinent. *Navigat J Inst Navigat* 55(3):463–475
10. Swathi N, Srilatha Indira Dutt VBS, Sasibhushana Rao G (2020) MCMC particle filter approach for efficient multipath error mitigation in static GNSS positioning applications. In: Saini HS, Singh RK, Tariq Beg M, Sahambi JS (eds) Innovations in electronics and communication engineering. Lecture notes in networks and systems, vol 107. Springer, Singapore
11. Bose A, Das S, Malik R, Dutta D (2013) GALILEO-only position fix from India: first experience, coordinates 9(9):37–41
12. Kundu S, Santra A, Dutta D, Bose A (2017) Recent GNSS observations with GALILEO initial operation. In: National seminar on recent trends in condensed matter physics including LASER applications, NSCMPLA-2017, p 16
13. Misra P, Enge P (2006) Global positioning system: signals, measurements, and performance, 2nd edn. Ganga-Jamuna Press, Lincoln MA
14. Reference Frames in GNSS. https://gssc.esa.int/navipedia/index.php/Reference_Frames_in_GNSS. Last accessed 21 Aug 2021
15. Seeber G (2003) Satellite geodesy: foundations, methods, and applications, 2nd edn. Walter de Gruyter, Berlin, New York
16. Kaplan ED, Hegarty CJ (2006) Understanding GPS principles and applications, 2nd edn. Artech House, Norwood, MA
17. National Geodetic Survey OPUS (Online Positioning User Service). <https://www.ngs.noaa.gov/OPUS/>. Last accessed 09 Aug 2021

18. European GNSS Service Centre, Open Service - Service Definition Document (issue 1.0), December (2016)
19. GPS Interface Control Document (ICD-GPS-200), ARINC Research Corporation, July (1991)

Mapping User-Submitted Short Text Questions to Subjects of Study: A Multinomial Classification Approach



Sanjay Singh and Vikram Singh

Abstract Text classification is the process of assigning labels to text documents for categorizing them. Accurate classification of user-submitted questions is of great use for any e-learning environment where the students are constantly submitting new questions that need to be presented to their appropriate subject experts for resolution. The user-submitted questions are typically short text documents, with a vastly disproportionate number of questions in different subjects. To resolve this, generalized linear model, distributed random forest, extremely randomized trees, gradient boosting model, extreme gradient boosting (XGBoost) and deep learning (ANN) models were trained and compared on a dataset of short text questions, while utilizing TF-IDF feature extraction and grid search for hyperparameter tuning. It was found out that an appropriately tuned XGBoost model gives the best performance out of these. The customized XGBoost model was then combined with a mechanism to handle imbalanced classes to propose a model that demonstrates further improved performance on the heavily imbalanced dataset.

Keywords XGBoost · Text classification · Multinomial

1 Introduction

Ever since the advent of the Internet, people have been trying to bring the process of teaching–learning online. This leads us to the huge research area of e-learning technologies that aim to help learners learn with comparatively greater convenience than a traditional classroom. There have been various e-learning platforms that are attempting to effectively teach students, but the ability of a learner to ask questions to a subject expert for an effective resolution is, at its core, one of the most fundamental activities involved in any learning process. In fact, a good question, combined with its solution, will continue to help future learners who want to learn the subject, who can learn a lot by reading other people’s questions and their solutions. For this, and many

S. Singh (✉) · V. Singh

Department of Computer Science and Engineering, Chaudhary Devi Lal University, Sirsa, India
e-mail: thabhambhu@gmail.com

other reasons, in an e-learning environment that deals with questions, it becomes very helpful to classify the questions according to subject areas for an organized study. Also, many e-learning platforms have various subject experts on the back-end who deal with specific subjects, it becomes crucial to show the appropriate subject's question to the concerned subject expert, and this problem also deals with question classification which is a special case of a domain of research that is called text classification. Text classification deals with utilizing machine learning techniques to assign a class/label to any input text. It has been used widely to classify product reviews by corporations that want to understand the sentiment of their customers, or many specific cases such as to predict the ideological direction of court cases [1], or to classify fake news or hoax, which have become the most prevalent cybercrime in today's day and age that have immeasurable harms [2] or to identify a person's distinctive habits and behaviors through personality classification [3]. Various papers resolve the issue of programmed text classification proposing various strategies and arrangements. Extensive thorough reviews also exist that document text classification in detail [4–7].

2 Materials and Methods

Extensive research has been conducted in the domain of short text classification as documented in the excellent review by [8]. When the dataset consists of user-submitted short questions with a vastly disproportionate number of questions in different subjects, the overall classification becomes a challenge. Researchers in [9] have shed light upon the data imbalance problem in text classification. Over the years, there have been many key enhancements in the overall research domain of text classification giving us various techniques and tools for almost any text classification problem one can think of. These techniques have become industry standard so they deserve a brief review.

2.1 *TF-IDF*

TF-IDF stands for term frequency-inverse document frequency which is a kind of statistical method that is utilized for evaluating the significance of a word in a document set. The significance of a word is proportional to the number of times it appears in the document and inversely proportional to the number of times it appears in the entire document set [10].

It works on the notion that the number of times a word appears in a document, the more important it is, and the more common the word is in the entire document set, the less important it becomes. TF-IDF is determined by multiplying the calculated term frequency value with the inverse document frequency.

$tf(t, d)$ is the prevalence of a term t , known as the term frequency,

$$\text{tf}(t, d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}} \quad (1)$$

where $f_{t,d}$ being the raw count of the occurrences of the term t in a document d .

The measure of the amount of information a word presents is known as the inverse document frequency, that is, how common it is among all documents. It is determined by calculating the logarithm of the value obtained by dividing the count of all documents by the count of those documents that include the term:

$$\text{idf}(t, D) = \log\left(\frac{N}{|d \in D : t \in d|}\right) \quad (2)$$

where the denominator deals with the number of documents wherein the term t appears, and $N = |D|$ (the number of documents).

TF-IDF is calculated as

$$\text{tfidf}(t, d, D) = \text{tf}(t, d) \cdot \text{idf}(t, D) \quad (3)$$

2.2 Grid Hyperparameter Search

In a machine learning model, there are mainly two types of parameters, the parameters that are derived via training and the parameters that control the training process. The latter is called hyperparameters, and one way of finding the best hyperparameters of a model is grid search. It is broadly of two types—traditional (or ‘Cartesian’) grid search and random grid search. In a Cartesian grid search, the values that need to be played are defined explicitly, and the model is trained for every combination of the hyperparameter values. Therefore, if there are three hyperparameters and 2, 4 and 8 values are specified for each of three parameters, there will be $2 * 4 * 8 = 64$ models in the grid.

In random grid search, the hyperparameter space is sampled uniformly from the set of all possible value combinations [11]. In this case, the grid search is terminated when some specified stopping criterion is satisfied. The stopping criterion is often on limiting the number of models to train or the time allowed for the search, or in advanced cases, a performance metric-based stopping criterion, which will terminate the search when the performance stops improving by the specified amount.

2.3 Models

Generalized Linear Model (GLM). It is an adaptable abstraction of linear regression which permits the response variable to embody an error distribution excepting normal

distribution. The GLM relates the linear model with the response variable through a link function and by way of realizing the amount of variance of every measurement as a function of its estimated value. GLMs estimate regression models for results following exponential distributions. These include Poisson, Gaussian, gamma, as well as binomial distributions, each assisting a distinctive purpose. An appropriate one can be utilized either for classification or for prediction depending upon the choice of link function and distribution [12].

Multi-class Classification (Multinomial Family). The current problem required multi-class classification, so a generalization of the binomial model called multinomial family which is used for multi-class classifications was considered. Just like binomial family, it models the conditional probability of occurrence of class ‘c’ given ‘x’ [13]. For each of the output classes, a vector of coefficients exists. The probabilities are defined as

$$\hat{y}_c = \Pr(y = c|x) = \frac{e^{x^T \beta_c + \beta_{c0}}}{\sum_{k=1}^K (e^{x^T \beta_k + \beta_{k0}})} \tag{4}$$

where β_c is a vector of coefficients for class ‘c’.

GLM has its own grid search interface. For the purpose of comparison, a model with λ search enabled was built and provided with a list of α values. It resulted in a single model with the best α - λ combination instead of having one model for each α . Table 1 illustrates the hyperparameter values that were used for grid search.

Distributed Random Forest (DRF). DRF is another robust tool for text classification. DRF works by generating a forest of weak-learning classification trees made from subsets of the data, instead of generating a single tree. Due to a large number of trees, variance is reduced and the average estimation across all the trees is used to make the concluding prediction [14].

Extremely Randomized Trees (XRT). This model further reduces the variance by improving the splitting rule. In XRTs, for each feature, thresholds are generated randomly, and the best of these thresholds is chosen as the splitting rule [15].

Gradient Boosting Machine (GBM). It is a forward-learning ensemble technique. The foundation of GBM is that increasingly refined approximations yield better prediction results [16]. The grid-searched hyperparameters are displayed in Table 2.

Deep Learning Models. There has been extensive work at utilizing deep learning-based models for the purpose of classification of text documents, an excellent survey of which is published in [17]. The model that was trained and evaluated for the purpose of this problem is constructed over a feed-forward multi-layer ANN (artificial neural network), also known as a multi-layer perceptron (MLP), which works

Table 1 GLM hyperparameters used for grid search

Parameter	Values
α	{0.0, 0.2, 0.4, 0.6, 0.8, 1.0}

Table 2 GBM hyperparameters used for grid search

Parameter	Values
Sample rate	{0.50, 0.60, 0.70, 0.80, 0.90, 1.00}
Column sample rate per tree	{0.4, 0.7, 1.0}
Min rows	{1, 5, 10, 15, 30, 100}
Column sample rate	{0.4, 0.7, 1.0}
Min split improvement	{1e-4, 1e-5}
Max depth	{3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17}
Learning rate	{0.1}

well on tabular (transactional) data. Other types of deep neural networks include recurrent neural networks (RNNs) and convolutional neural networks (CNNs) which are a good choice for sequential data (like time-series) and image data, respectively. The stochastic gradient descent was utilized to train the model using back-propagation. The model can comprise a considerable quantity of hidden layers which were compared through grid search for high prediction accuracy. The activation function used was rectifier (with dropout). An image of the global model parameters was trained by each compute node on its local data asynchronously (through multi-threading) and contributed to the global model periodically via model-averaging across the network, and adaptive learning rate ADADELTA [18] was used. Table 3 shows the hyperparameters that were grid-searched.

Extreme Gradient Boosting (XGBoost). It is an enhancement of GBM supervised learning algorithm that works by sequentially combining multiple decision trees. It utilizes the idea of boosting which refers to the notion of creating multiple models, with every model reducing the loss error in the previous model sequentially. XGBoost can also solve many data science problems rapidly by utilizing parallel tree boosting [19]. The grid-searched hyperparameters are stated in Table 4.

Table 3 Deep learning hyperparameters used for grid search

Parameter	Grid-searched values
Input dropout ratio	{0.0, 0.05, 0.1, 0.15, 0.2}
Hidden layers	Set 1: {20}, {50}, {100}
	Set 2: {20, 20}, {50, 50}, {100, 100}
	Set 3: {20, 20, 20}, {50, 50, 50}, {100, 100, 100}
ρ	{0.9, 0.95, 0.99}
Hidden dropout ratios	Set 1: {0.1}, {0.2}, {0.3}, {0.4}, {0.5}
	Set 2: {0.1, 0.1}, {0.2, 0.2}, {0.3, 0.3}, {0.4, 0.4}, {0.5, 0.5}
	Set 3: {0.1, 0.1, 0.1}, {0.2, 0.2, 0.2} {0.3, 0.3, 0.3}, {0.4, 0.4, 0.4}, {0.5, 0.5, 0.5}
ϵ	{1e-6, 1e-7, 1e-8, 1e-9}

Table 4 XGBoost parameters used for grid search

Parameter	Grid-searched values
Booster	gbtree, dart
Column sample rate	{0.6, 0.8, 1.0}
Column sample rate per tree	{0.7, 0.8, 0.9, 1.0}
Max depth	{5, 10, 15, 20}
Min rows	{0.01, 0.1, 1.0, 3.0, 5.0, 10.0, 15.0, 20.0}
Reg alpha	{0.001, 0.01, 0.1, 1, 10, 100}
Reg lambda	{0.001, 0.01, 0.1, 0.5, 1}
Learning rate	{0.3}
Sample rate	{0.6, 0.8, 1.0}

3 Proposed Approach

Unlike most other research on short text classification that utilizes external vocabularies or knowledge base as in [20] or use word embeddings for semantic expansions as performed in [21] the best general-purpose approach that can be reused and generalized without the need for external vocabulary information was attempted. The approach begins with data cleaning, followed by feature extraction using TF-IDF, and then proceeds to apply the aforementioned text classification techniques to get the initial best possible model, chosen based on the performance comparisons utilizing respective grid hyperparameter tuning for each model. The resulting model is then combined with a technique for handling imbalanced data using class weights to get the most optimal results. The flow of the proposed approach is illustrated in Fig. 1.

The dataset consists of 6925 questions from 13 computer science subject areas that were collected and labeled manually by subject experts in respective fields. For the purpose of training and validation, the dataset was shuffled and split across training data and validation data in the proportion of 8:2, respectively. The dataset was then passed through a cleaning process that lowercased the text, removed any punctuations and non-sensical data and tokenized it. Stemming was performed using Snowball stemmer [22], and stop-words were removed. The resulting dataset contained fields for the cleaned question text and an identifier for its appropriate subject. The dataset consists of short text questions in unbalanced proportions, the distribution of which is shown in Fig. 2.

Using grid-based hyperparameter search, multiple iterations of each of the model types were trained and evaluated. The best iteration of each model is presented in Table 5.

As seen in Table 5, an XGBoost iteration had the minimum mean per-class error out of all the training and evaluation iterations of all models. Still, the model suffered from high classification error and log loss due to imbalanced classes as seen in Fig. 3.

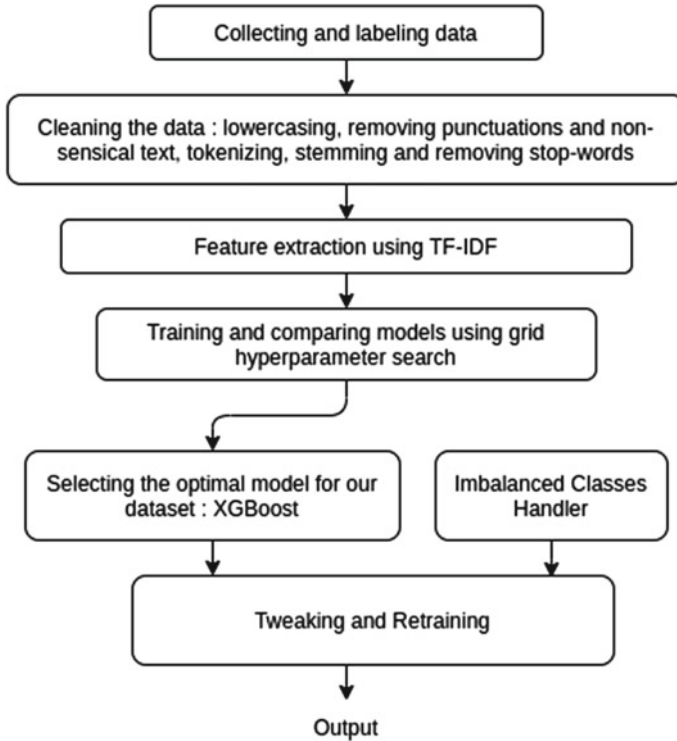


Fig. 1 Flow diagram of the overall approach

This problem of imbalanced classes refers to the situation that all the subjects have not been represented equally. More precisely, there is a huge disproportionality between classes, with certain subjects having more than 800 samples and some not even having 80 samples. The problem of imbalanced classes has been greatly explored in excellent reviews in [23, 24]. Class weights were calculated from the dataset distribution frequency, and the model was retrained with the best hyperparameters found by the previous search. As a result, the $F1$ score as well as precision and recall values improved as seen in Table 6.

4 Results and Conclusions

In the quest for a general-purpose approach for assigning appropriate subject labels to the user-submitted questions, with no availability of any extra vocabulary information related to the subjects, several grid hyperparameter-searched iterations of Generalized Linear Model, Deep Learning (ANN), Gradient Boosting Model, Extreme Gradient Boosting Model (XGBoost), Distributed Random Forests and Extremely

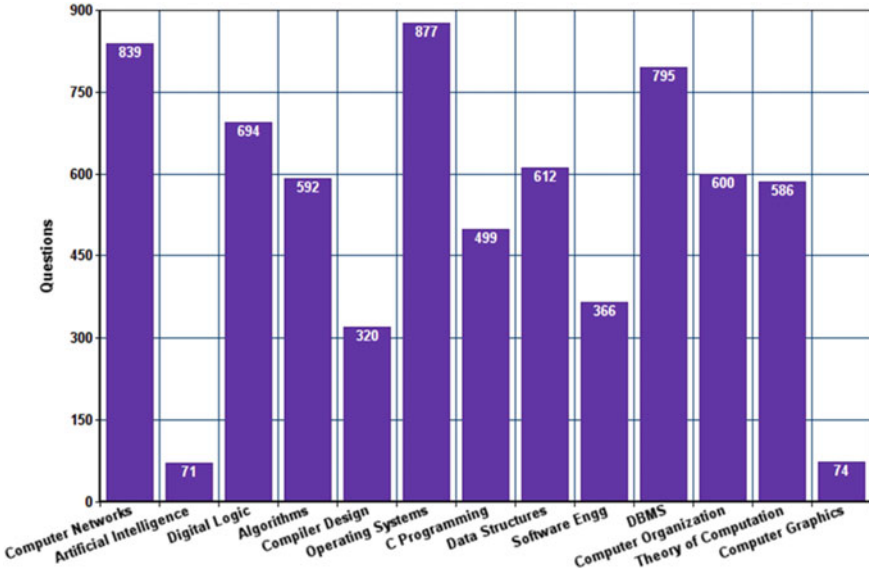


Fig. 2 Dataset distribution

Table 5 Comparison of the trained models

Model	Mean per-class error	Log loss	RMSE	MSE
XGBoost	0.211018	0.571816	0.404856	0.163908
GLM	0.23553	0.527782	0.387519	0.150171
Deep learning	0.249135	0.995679	0.426746	0.182112
GBM	0.260535	0.773564	0.473971	0.224648
DRF	0.2741	0.853088	0.527814	0.278588
XRT	0.294788	0.823955	0.524296	0.274886

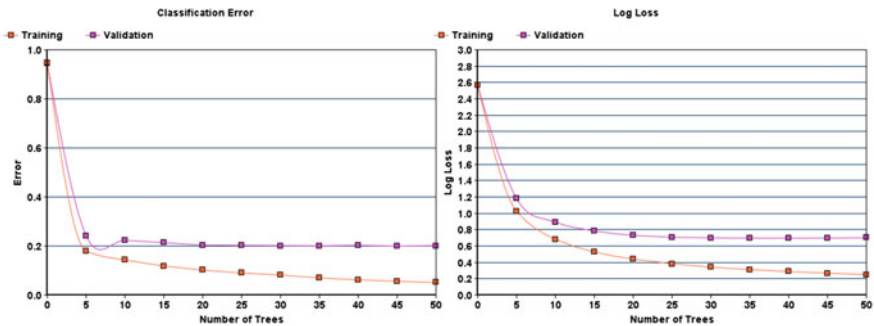


Fig. 3 Classification error and log loss for XGBoost iteration

Table 6 Classification report for XGBoost

Class	Before class weights			After class weights			Support
	Precision	Recall	F1-Score	Precision	Recall	F1-Score	
Artificial intelligence	0	0	0	0.29	0.14	0.19	14
Computer networks	0.84	0.85	0.84	0.87	0.83	0.85	168
Digital logic	0.8	0.89	0.84	0.79	0.86	0.83	143
Algorithms	0.74	0.72	0.73	0.72	0.72	0.72	109
Computer organization	0.75	0.79	0.77	0.73	0.79	0.76	112
Operating systems	0.81	0.81	0.81	0.78	0.75	0.76	167
C programming	0.75	0.78	0.77	0.79	0.71	0.75	105
Data structures	0.72	0.78	0.75	0.69	0.8	0.74	108
Software engineering	0.88	0.78	0.83	0.87	0.79	0.83	78
Compiler design	0.71	0.58	0.64	0.74	0.73	0.73	73
DBMS	0.86	0.89	0.88	0.86	0.87	0.86	166
Theory of computation	0.89	0.9	0.9	0.89	0.88	0.88	131
Computer graphics	0.17	0.09	0.12	0.13	0.18	0.15	11

Table 7 Performance metrics of XGBoost on the classification of validation data

	Before class weights			After class weights			Support
	Precision	Recall	F1-Score	Precision	Recall	F1-Score	
Macro avg	0.69	0.68	0.68	0.7	0.7	0.7	1385
Weighted avg	0.79	0.8	0.79	0.79	0.79	0.79	1385

Randomized Trees were trained and evaluated, and it was found that out of these, XGBoost performs the best with a small and imbalanced dataset of very short text documents.

To further improve the classification performance, a general-purpose approach to handle the unbalanced classes was used utilizing class weights. The overall average performance of XGBoost on the validation data has been shown in Table 7.

The proposals for future work would be to test this approach on multilingual questions and possibly utilizing more advanced feature selection techniques for enhancing this approach.

References

1. Hausladen CI, Schubert MH, Ash E (2020) Text classification of ideological direction in judicial opinions. *Int Rev Law Econ* 62:105903. <https://doi.org/10.1016/J.IRLE.2020.105903>
2. Haumahu JP, Permana SDH, Yaddarabullah Y (2021) Fake news classification for Indonesian news using Extreme Gradient Boosting (XGBoost). *IOP Conf Ser Mater Sci Eng* 1098:052081. <https://doi.org/10.1088/1757-899X/1098/5/052081>

3. Khan AS, Ahmad H, Asghar MZ, Saddozai FK, Arif A, Khalid HA (2020) Personality classification from online text using machine learning approach. *Int J Adv Comput Sci Appl* 11. <https://doi.org/10.14569/IJACSA.2020.0110358>
4. Joachims T (2002) Learning to classify text using support vector machines. Springer Science & Business Media
5. Fabrizio S (2002) Machine learning in automated text categorization. *ACM Comput Surv CSUR* 34:1–47. <https://doi.org/10.1145/505282.505283>
6. Aggarwal CC, Zhai C (2012) Mining text data. Springer Science & Business Media
7. Chaturvedi A, Yadav S, Ansari MAMH, Kanojia M (2021) Comparative multinomial text classification analysis of Naïve Bayes and XGBoost with SMOTE on imbalanced dataset 339–349. https://doi.org/10.1007/978-981-16-2543-5_29
8. Song G, Ye Y, Du X, Huang X, Bie S (2014) Short text classification: a survey. *J Multimed* 9:635
9. Li Y, Sun G, Zhu Y (2010) Data imbalance problem in text classification. In: 2010 Third international symposium on information processing. IEEE, New York, pp 301–305
10. Ramos J et al (2003) Using TF-IDF to determine word relevance in document queries. In: Proceedings of the first instructional conference on machine learning. Citeseer, pp 29–48
11. Bergstra J, Bengio Y (2012) Random search for hyper-parameter optimization. *J Mach Learn Res* 13
12. Zheng B, Agresti A (2000) Summarizing the predictive power of a generalized linear model. *Stat Med* 19:1771–1781
13. Faraway JJ (2016) Extending the linear model with R: generalized linear, mixed effects and nonparametric regression models. CRC Press
14. Biau G, Scornet E (2016) A random forest guided tour. *TEST* 25:197–227
15. Geurts P, Ernst D, Wehenkel L (2006) Extremely randomized trees. *Mach Learn* 63:3–42
16. Ayyadevara VK (2018) Gradient boosting machine. In: Ayyadevara VK (ed) Pro machine learning algorithms: A hands-on approach to implementing algorithms in Python and R, pp 117–134. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-3564-5_6
17. Minaee S, Kalchbrenner N, Cambria E, Nikzad N, Chenaghlu M, Gao J (2021) Deep learning–based text classification: a comprehensive review. *ACM Comput Surv CSUR* 54:1–40
18. Zeiler MD (2012) ADADELTA: an adaptive learning rate method. *ArXiv12125701 Cs*
19. Chen T, He T, Benesty M, Khotilovich V, Tang Y, Cho H et al (2015) Xgboost: extreme gradient boosting. R Package Version 04-2. 1, pp 1–4
20. Wang F, Wang Z, Li Z, Wen J-R (2014) Concept-based short text classification and ranking. In: Proceedings of the 23rd ACM international conference on conference on information and knowledge management. Association for Computing Machinery, pp. 1069–1078. <https://doi.org/10.1145/2661829.2662067>
21. Wang P, Xu B, Xu J, Tian G, Liu C-L, Hao H (2016) Semantic expansion using word embedding clustering and convolutional neural network for improving short text classification. *Neurocomputing* 174:806–814. <https://doi.org/10.1016/j.neucom.2015.09.096>
22. Jivani AG (2011) A comparative study of stemming algorithms. *Int J Comp Tech Appl* 2:1930–1938
23. Kotsiantis S, Kanellopoulos D, Pintelas P (2006) Handling imbalanced datasets: a review. *GESTS Int Trans Comput Sci Eng* 30:25–36
24. Datta S, Arputharaj A (2018) An analysis of several machine learning algorithms for imbalanced classes. In: 2018 5th International conference on soft computing machine intelligence (ISCMI), pp 22–27. <https://doi.org/10.1109/ISCMI.2018.8703244>

Physical Layer Security Aspects of D2D Communications in Future Networks



Chinnam S. V. Maruthi Rao  and Ramakrishna Akella

Abstract Device-to-device communication (D2D) allows direct communication of two devices in the network and initially proposed for cellular networks. D2D finds applications where there is no preexisting/damaged infrastructure network is available. With the advent of technology, there is large possibility of the data to be hacked in various methods. There are a wide range of security protocols that help in preventing these attacks in the upper layers. But securing the data in the physical layer itself will provide smooth communication with low overhead and high spectrum efficiency. There have been a few physical layer security protocols available in the literature. In this paper, the key issues in security of D2D communication at physical layer and the available solutions are presented.

Keywords D2D · Security · Privacy · Encryption · Symmetric · Asymmetric

1 Introduction

Device-to-device (D2D) communication is a technique that allows the devices to communicate directly with one another in the proximity without the need of a base station on as single hop [1]. The D2D communications have advantages of improved spectral and energy efficiency, high throughput, less delay and fairness compared to the traditional cellular communications [2]. In the recent times, D2D is finding a wide range of popularity in intelligent transportation systems (ITSs) [3]. The D2D communication structure is classified as: (i) standalone or peer-to-peer communication (between UE's) that does not need access points, and the connectivity is based on geographic validity or temporal validity as shown in Fig. 1a and (ii) network-assisted communication that needs infrastructure (between UE and BS) as shown in Fig. 1b. In either type, the data transmission is in short bursts of time and in full-duplex [4,

C. S. V. Maruthi Rao (✉) · R. Akella
Department of Electronics & Communication Engineering, KL University, Vaddeswaram, India
e-mail: maruthichinnam@gmail.com

R. Akella
e-mail: ramakrishna.a@kluniversity.in

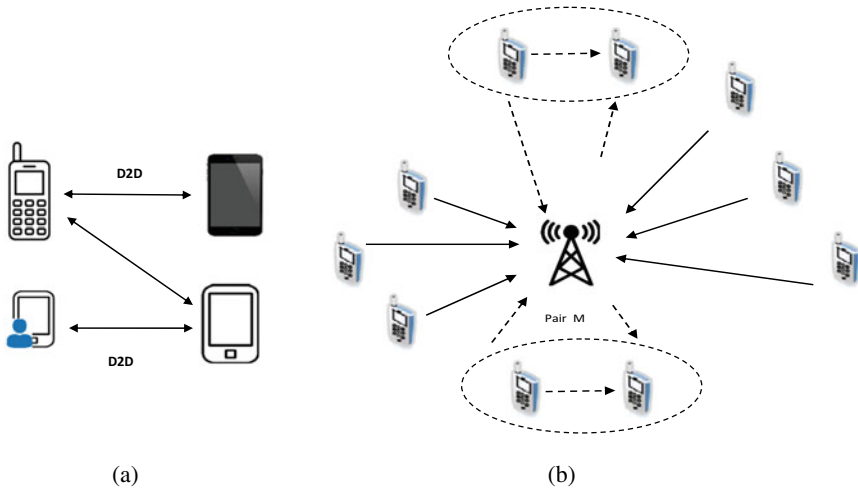


Fig. 1 a Standalone D2D. b Network-assisted D2D

5], thereby increasing the spectrum efficiency. The number of devices is limited by the technology used within a limited range.

D2D is similar to M2M communications except for the fact that the M2M communication is more application-oriented and focuses on the proximity and cannot improve the spectrum efficiency. The D2D also looks similar to MANET's and CRN's architecturally, where the nodes function autonomously and use cognitive sensing in CRNs [6]. D2D was initially proposed in cellular networks for multi-hop relays [2] and later on used for improving the spectral efficiency, multicasting, video dissemination, cellular off-loading, etc.

D2D communication has found a wide range of applications especially when there is a need for ad hoc networks, where the network infrastructure is damaged or not covered. For cellular applications, D2D was standardized in 3GPP with LTE function, but has a lot of limitations. Currently, the D2D is seen as a potential solution for emergency communications, IOT enhancements and local services on the 5G platform. D2D in 5G can be applied in scenarios like multi-user MIMO enhancement, co-operative relaying and virtual MIMO. In 5G, D2D can be used in a large mobile interconnected network where there is a need for diverse services.

1.1 D2D Classification

D2D communication is classified in terms of spectrum as inband and outband D2D communications. The inband D2D uses the licensed spectrum for D2D as well as cellular link, which results in high interference and is classified as underlay and overlay types. In underlay, the same resources are shared between cellular and D2D,

while in overlay they use dedicated resources. Outband D2D uses the unlicensed spectrum. However, the problem with managing the interference is overcome by using an extra wireless interface like Wi-Fi, Bluetooth, etc. [6]. The outband D2D is further classified as autonomous and controlled types. In controlled D2D, the control of the second interface is under cellular network, while in autonomous, the D2D is left to the users wherein the entire communication is controlled by the cellular network [7].

The D2D can also be categorized based on the degree of involvement of the BS involved in the communication like device relaying, direct communication between devices with the assistance of BS and device-assisted control links [6].

1.2 D2D in 4G/5G

The need for faster data rates and longer ranges has led to the introduction of D2D communication in 4G. While the D2D can be enabled through 3GPP, they differ mostly in the number of hops including quick device discovery and higher data rates with low latency. D2D can also be used to implement M2M communication in IOT, exclusively in vehicular communication (V2V) because of the low latency and real-time responses. V2V communication is based on D2D defined as a part of ProSe services in release 12 and release 13 of the specification.

Recent developments show that 5G can have data rates at least 1000 times faster than 4G. The key techniques of D2D in 5G include device and network discovery, mobility, synchronization, resource management, power control, interference, mode switching, etc. In 5G, research shows that the D2D can be applied through HetNet (heterogeneous networks) that include high data rate, improved spectral efficiency with long- and short-range communication technologies like mmwave (short range), CRN (improved coverage and speed) and massive MIMO (concurrency). In 5G networks, D2D when applied with above techniques shows high noise resilience, improved channel capacity and an improved channel efficiency.

1.3 Challenges in D2D Communication

The major challenges in the D2D communication include device discovery, resource allocation and management, interference control, mode selection, energy management and security assurance. Various algorithms had been proposed to overcome these challenges, and to name a few, TDS, SPS and particle swarm optimization for resource allocation, Game theory-based power control scheme and Diffie–Hellman secure key establishment are the most popular.

Despite extensive research going on in the D2D, the security issues are least addressed [8]. With the present technology, the possibility of information theft is very high and can happen on the network or user applications or on the communication

links. Security in the D2D communications can be improved by providing security through the physical layer [7].

1.4 Motivation and Contribution

One of the biggest challenges of D2D communications is the privacy and secrecy of information as data are not stored at a fixed storage location. The standalone D2D operates in a distributed networking environment where a user device broadcasts the information over the wireless channels. During the process of destination device discovery, an attacker can locate and track the users normally known as violation of location privacy, wherein for network-assisted D2D such type of risk is limited.

2 Security Aspects

This work focuses on the security aspects of D2D communication. In D2D communication, security issues can be due to direct wireless connection, relay transmission structure, mobility and privacy issues in social applications [8]. For the wireless communication to be secured, it has to meet the requirements of confidentiality, authentication and non-repudiation, privacy, availability, integrity, traceability, revocability and FGAC (fine-grained access control) [1]. In most of the wireless networks, crypto algorithms are introduced in upper layers to provide security, where two devices have a common private key. Ex. DES [9]. In the absence of this private key for at least one device, a secure channel is needed. Instead of establishing a channel additionally, physical layer methods can be employed to distribute the key [9]. The radio nature of D2D communications results in the following security threats as summarized in Table 1 along with existing algorithms/framework.

All the below security schemes are complemented by the security schemes from higher levels to improve the security of the entire system [10]. There are many types of attacks possible in D2D, which can be avoided using a suitable security protocol [6]. Most experiments have shown that majority of DOS and DDOS attacks occur on UE relay [8].

2.1 Physical Layer Security in D2D

The characteristics of channel used in wireless communication include data coding and modulation techniques, the number of antennas and their jamming capabilities of eavesdroppers, which are considering while defining the physical layer security in D2D communication [7]. Knowledge on physical layer security can be gained only with thorough knowledge in information theory and physical layer architecture.

Table 1 Comparison of nature of threats

S. No.	Nature of threat	Effect on requirement	Existing framework/protocol/algorithm	Detail
1	Eavesdropping attack/session hijacking/IP spoofing	Confidentiality	Secured network coding-based data splitting algorithm [12]; KEEP algorithm	Attackers can read the messages communicated and inject fake message
2	Impersonate attack/Masquerading Attack/MITM Attack	Authentication, revocability	Key agreement management [16]; Identity-based cryptography and legitimacy pattern attributes (ABE)	Attacker uses the identity of other user with a false identity
3	Forge attack	Integrity	Error-correcting codes	Similar to impersonate
4	Free-riding attack	Quality of experience	Secure data sharing protocol [12]; Ex: BarterCast	Lazy/selfish D2D nodes does not share data with other D2D nodes
5	Active attack	Authentication, confidentiality and Integrity	Symmetric key encryption; secure network coding-based data splitting algorithm [12]	One device is attacked and it can spread through the network
6	Privacy sniffing/violation	Privacy, fine-grained access control	Genetic algorithm for maximizing secrecy rate [12]; ES ³ A protocol and algorithms including ElGamal	Attackers forge the trust so that users still relying on the trust causes privacy issues
7	Denial of service attack	Authentication, availability and dependability	Symmetric encryption algorithms, identity-based cryptographic algorithms	To avoid Dos, switch to another channel

2.2 Solutions for Physical Layer Security in D2D

Based on the comparisons done in [8], the physical layer security provides availability and dependability. By using CSI-based key extraction or PHY-based cooperative key generation in the physical layer, we can attain confidentiality and integrity, but the availability and dependability will be reduced/lost [8].

According to Zhang et al., the physical layer security in D2D is established by extensive analysis and application of characteristics of channel between the devices. Most security algorithms developed for physical layer till date are dependent on the Diffie–Hellman key agreement protocol. It is a known fact that the Diffie–Hellman

Key agreement protocol is vulnerable to MITM attacks because of the mutual authentication problem between the devices [11]. Xi et al. proposed a CSI based on key extraction protocol to validate the key between the devices using a hash function that prevents the leakage of key information [8]. Hash function in cryptography converts an arbitrary length input data into a fixed length output data [12]. Hash functions are helpful in creation of digital signatures and verifying the data integrity [13].

Along with hash functions, various crypto algorithms are proposed to meet the security requirements of D2D communication. Basically, the encryption techniques are classified as symmetric and asymmetric based on the nature of the key used. The symmetric cryptography also termed as a private key cryptography uses a single key for both encryption and decryption, while the asymmetric cryptography termed as public key cryptography uses two keys. Examples of symmetric cryptography include DES and AES, while RSA and ECC are examples of asymmetric cryptography [14]. The symmetric cryptography is found to be useful in standalone D2D, and a combination of symmetric and asymmetric (hybrid) crypto algorithms is more beneficial in network-assisted D2D, especially heterogeneous networks. J. Wan et al. have compared these algorithms and concluded that high performance with low overhead can be achieved through generation of symmetric key and exchanging using asymmetric algorithms [15].

However, there are many limitations to the current hybrid algorithms that include lengthy time, high entropy for producing symmetric key using pseudo-random number generation and implementation of asymmetric algorithm with less overhead. J. Wan et al. proposed an efficient solution for these limitations in automotive field by creating a wireless channel-based PSK (Pre-Shared Key) generation technique by optimizing its length and adopting required methods to convert this key into suitable encryption [15].

Weifeng Lu et al. have developed a model to improve the physical layer security and efficiency of communication in D2D underlay type by ensuring channel security to DUEs and improving security rates in CUEs [16]. Daidong Ying et al. proposed a physical layer security scheme for D2D, wherein each D2D transmission link is protected by the relay user to suppress the eavesdroppers [17]. Most recently, MingSheng Cao et al. proposed a very promising and efficient lightweight method securing the D2D communication in the physical layer using multiple sensors and IOT [18]. A comparison of the various parameters considered in the selection of a crypto key is discussed in Table 2 from the existing research. With the analysis of the available literature and various studies, it can be understood that a lightweight hybrid algorithm is preferable for securing the D2D communication in the physical layer for improving the authentication, integrity and confidentiality.

All the above techniques are based only on the physical layer encryption techniques. From the study of research papers available, it is understood that physical layer security compromises other parameters and system requirements.

Few other techniques are also available for securing the physical layer in communication. Beam-forming and artificial noise injection into the channel is one way of improving the physical layer security by minimizing the secrecy outraging to the least possible but needs additional power consumption and finds its use in applications like

Table 2 Crypto keys: a comparison [15]

	Symmetric	Asymmetric	Hybrid
Authentication (Key/Data)	MAC for both key and data	DS for both key and data	Keys: DS Data: MAC
Confidentiality (Data/Key)	Data encryption (Same)	Data encryption (Same)	Asymmetric key encryption; Symmetric data encryption
Code size	MegaBytes	MegaBytes	MegaBytes
Key length	AES: 128–256 bits	ECC: 256–384 bits RSA: 1024–3072 bits	512–3072 bits for Asymmetric; 32–256 bits for Symmetric
Key sharing	Yes (Pre-Shared)	No	Yes (Randomly Generated)
Vulnerability	Brute force, cipher and plain text attacks	Brute force, MITM attacks	Cloning attacks
Performance	High performance and low overhead	Too slow; high computational requirement and memory usage	High performance with less overhead
Tunability	No	Yes	Yes

telemedicine [19]. Bit flipping is a technique based on CSI using clustering method that helps in confusing the eavesdropper while reducing the energy consumption and transmission time [20] and is used for sensor networks. Constellation rotation is an emerging physical layer security technique that can be fully exploited for channel degree of freedom and is an area of extreme interest in 5G and 6G communications.

3 Conclusion and Future Direction

D2D communication is a promising technology that is used with 5G and future networks. V2V is a promising application of D2D that provides a lot of opportunities used in leveraging the traditional cellular networks. The network connectivity is maintained by the ad hoc nature of V2V that address the randomly changing channel conditions and the required QoS parameters in terms of latency, safety and reliability. V2X communication is considered to be a game changer in 5G communications where autonomous vehicles can communicate with one another and with everything like network, infrastructure, pedestrian, etc., using the D2D with full duplex links establishment and communication in proximity. “D2D can also exploit social networks for providing secured connection setup” [21].

In 5G and future networks, an enormously increased data traffic has to be managed effectively while securing the data. In these networks, communication is based mostly on D2D and multicasting. The network resources have to be improved and managed properly while same information may need to be transferred to multiple devices. In such cases, D2D provide point-to-multi-point communication among the carefully selected relay nodes in proximity and thereby reduce the multicasting area. To enhance the performance and security, secured and trustworthy encryption algorithms are used [22].

The various types of threats, attacks and their effect on the basic requirements of security in D2D communication in the physical layer are emphasized here. It is observed that lightweight hybrid encryption techniques that uses a symmetric pre-shared key with an asymmetric exchange provides efficient security in the physical layer. This is mainly because of the limitation in resources and performance in the 5G and future networks. There are still a lot of potential security issues that needs extensive research with a fixed framework for various scenarios especially by combining the resource allocation and interference management mechanisms.

References

1. Haus M, Waqas M, Ding AY, Li Y, Tarkoma S, Ott J (2017) Security and privacy in device-to-device (D2D) communication: a review. *IEEE Commun Surv Tutor* 19(2):1054–1079
2. Asadi A, Wang Q, Mancuso V (2014) A survey on device-to-device communication in cellular networks. *IEEE Commun Surv Tutor* 16(4):1801–1819
3. Cheng X, Yang L, Shen X (2015) D2D for intelligent transportation systems: a feasibility study. *IEEE Trans Intell Transp Syst* 16(4):1784–1793
4. Alkurd R, Shubair RM, Abualhaol I (2014, June) Survey on device-to-device communications: challenges and design issues. In: *IEEE 12th International new circuits and systems conference (NEWCAS)*, pp 361–364
5. Wang L, Tian F, Svensson T, Feng D, Song M, Li S (2015) Exploiting full duplex for device-to-device communications in heterogeneous networks. *IEEE Commun Mag* 53(5):146–152
6. Gandotra P, Jha RK, Jain S (2017) A survey on device-to-device (D2D) communication: architecture and security issues. *J Network Comput Appl* 78:9–29
7. Zhu D, Swindlehurst AL, Fakoorian SAA, Xu W, Zhao C (2014, May) Device-to-device communications: the physical layer security advantage. In: *IEEE International conference on acoustics, speech and signal processing (ICASSP)*, pp 1606–1610
8. Wang M, Yan Z (2017) A survey on security in D2D communications. *Mob Networks Appl* 22(2):195–208
9. Shiu YS, Chang SY, Wu HC, Huang SCH, Chen HH (2011) Physical layer security in wireless networks: a tutorial. *IEEE Wirel Commun* 18(2):66–74
10. Montezuma P, Dinis R (2015) Implementing physical layer security using transmitters with constellation shaping. In: *24th International conference on computer communication and networks (ICCCN)*, pp 1–4
11. Shen W, Hong W, Cao X, Yin B, Shila DM, Cheng Y (2014) Secure key establishment for device-to-device communications. In: *IEEE global communications conference*, pp 336–340
12. Saravanan K, Senthilkumar A (2013) Theoretical survey on secure hash functions and issues. *Int J Eng Res Technol* 2(10):1150–1153
13. Sideris A, Sanida T, Dasygenis M (2019) Hardware acceleration of SHA-256 algorithm using NIOS-II processor. In: *8th International conference on modern circuits and systems technologies (MOCAS)*, pp 1–4

14. Rizk R, Alkady Y (2015) Two-phase hybrid cryptography algorithm for wireless sensor networks. *J Electr Syst Inform Technol* 2(3):296–313
15. Wan J, Lopez A, Faruque MAA (2018) Physical layer key generation: securing wireless communication in automotive cyber-physical systems. *ACM Trans Cyber-Phys Syst* 3(2):1–26
16. Lu W, Zheng X, Xu J, Chen S, Yang L (2019) Improving physical layer security and efficiency in D2D underlay communication. *Wireless Netw* 25(8):4569–4584
17. Ying D, Ye F (2019) D2D-assisted physical-layer security in next-generation mobile network. In: *International conference on computing, networking and communications (ICNC)*, pp 324–328
18. Cao M, Wang L, Xu H, Chen D, Lou C, Zhang N, Zhu Y, Qin Z (2019) Sec-D2D: a secure and lightweight D2D communication system with multiple sensors. *IEEE Access* 7:33759–33770
19. Wang H, Chen L, Wang W (2014) Enhancing physical layer security through beamforming and noise injection. In: *2014 Sixth international conference on wireless communications and signal processing (WCSP)*. IEEE, New York, pp 1–6
20. Yaacoub E, Chehab A, Al-Husseini M, Abualsaud K, Khattab T, Guizani M (2019) Joint security and energy efficiency in IoT networks through clustering and bit flipping. In: *2019 15th International wireless communications & mobile computing conference (IWCMC)*. IEEE, New York, pp 1385–1390
21. Ansari RI, Chrysostomou C, Hassan SA, Guizani M, Mumtaz S, Rodriguez J, Rodrigues JJ (2017) 5G D2D networks: techniques, challenges, and future prospects. *IEEE Syst J* 12(4):3970–3984
22. Suraci C, Pizzi S, Garompolo D, Araniti G, Molinaro A, Iera A (2021) Trusted and secured D2D-aided communications in 5G networks. *Ad Hoc Netw* 114:102403

The Modern Problem of Accessibility and Complexity of Big Data



Rodmonga Potapova , Vsevolod Potapov , and Petr Gorbunov 

Abstract The digital age allowed scientists to discover completely new ways to obtain, store, and process information. The main topic of this work, around which all the research presented below will develop, is big data. Being one of the realities of the modern world, big data is often used as the main material for research conducted in various fields of science. The phenomenon of big data is also interesting in terms of applied linguistics, where research is often based on language material obtained from social networks, Internet forums, news Websites, and other resources of the World Wide Web. The main purpose of this work is to study the problem of the availability and complexity of big data, to learn how to combine and compare its elements and how to use it regarding the material of one of the latest global linguistic studies conducted at the Institute of Applied and Mathematical Linguistics of the Moscow State Linguistic University. In the study, the methods of working with big data are considered that use the authors' own developments based on modern digital technologies. The presented software solves the problems of combining, comparing, and using extensive databases collected within fundamental interdisciplinary study of the impact of the aggressive Internet environment in the conditions of multimodal polycode communication in social networks on the transformation of the psychophysiological and cognitive characteristics of the personality of Internet users (in relation to the youth contingent).

Keywords Big data · Digital technologies · Linguistic databases

R. Potapova · P. Gorbunov (✉)

Institute of Applied and Mathematical Linguistics, Moscow State Linguistic University,
Ostozhenka Street 38, 119034 Moscow, Russia

e-mail: pete.spw@yandex.ru

R. Potapova

e-mail: r.potapova@linguanet.ru

V. Potapov

Centre of New Technologies for Humanities, Lomonosov Moscow State University, Leninskije
Gory 1, 119991 Moscow, Russia

e-mail: kedr@philol.msu.ru

1 Introduction

Big data is represented as structured and unstructured data of huge volumes and considerable variety, efficiently processed by various software tools.

There are lots of ways to use big data. So, with the help of big data, one can learn about the preferences of customers of a particular service sector, learn about the effectiveness of marketing campaigns, as well as organize a number of scientific projects that were previously inaccessible to scientists due to the huge amounts of research material and lack of funds for its processing.

While analyzing the scientific works devoted to big data, one can find that they have already been reflected in various fields of science—in biology [1], economics [2, 3], and philosophy [4]. Many researchers see not only positive, but also negative consequences of their occurrence, which is also evidenced by a number of works [5, 6].

Experts in the field of applied linguistics are also interested in big data. In this sphere, big data is helpful primarily in terms of statistics, which is one of the main research tools in the field of applied linguistics [7–9]. Big data offers a chance for linguistic researchers to collect a sufficient amount of linguistic material for a detailed analysis of all the factors of the experiment and confirmation of a hypothesis. This kind of material is necessary for the implementation of numerous research projects at the Institute of Applied and Mathematical Linguistics, and therefore, the issues of automating its collection and preliminary analysis are becoming more and more relevant every year.

2 Investigation and Method

As part of the multistage research conducted in recent years, the authors of this paper have proposed several software products of their own design, and the use of which in studying the problem of accessibility and complexity of big data could significantly simplify the creation of a software tool that automates the tasks of linguistic data collecting, combining, comparing, and using [10].

2.1 Development of a Distributed Integrated Mobile System for Remote Work with an Integrated Linguistic Database Based on the Use of Cloud Computing and Cloud Data Storage

One of such products was the Microsoft Azure autonomous research system and SQL Azure [11]. At the preparatory stage, the following tasks facing the project were formulated:

1. Collection of the necessary material for the formation of the test part of the cloud speech database (written and spoken texts, information about speakers)
2. Deployment of a cloud database on the Microsoft Azure platform.
3. Development of a client-side program that makes it possible to connect to a cloud database and perform various operations with the information stored in it (updating the database with new written and spoken texts, as well as information about speakers; displaying the speaker's personal card on request; saving the required spoken and written texts to the computer from which the database is connected).
4. Development of auxiliary utilities that facilitate interaction between the programmer and the user with files from the database (a program that performs trimming of an audio way file to the millisecond; a program that transforms any file into an array of bytes—the source format in which the database stores this file).

Next, a cloud database was created and configured using Microsoft Azure and Microsoft SQL Server Management Studio [12].

The program was developed on the basis of the Microsoft Visual Studio IDE. The DataKey project is a full-featured program for working with cloud databases (Fig. 1).

The main window of the program provides the user with the following features:

- viewing the contents of the database based on the set filter (the presence of written and spoken texts on a particular topic performed by a particular speaker);
- record management inside the database: loading a new record (performed through a separate window), deleting an existing record;
- downloading written and spoken texts to the workstation according to the selected record;
- viewing personal cards containing personal information for each speaker (performed through a separate window);

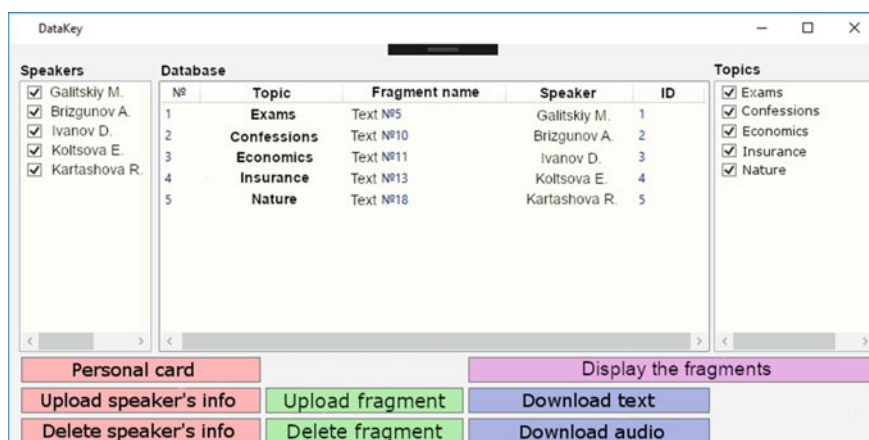


Fig. 1 Interface of the main window of the data key application

- managing personal cards inside the database: uploading a new card (performed through a separate window), deleting an existing card;

2.2 Fundamental Interdisciplinary Study of the Impact Of the Aggressive Environment of the Internet in the Conditions of Multimodal Multicode Communication in Social Networks on the Transformation of Psychophysiological and Cognitive Characteristics of the Personality of Internet Users (in Relation to the Young Users)

To get the most complete picture of the need to develop a system for working with big data, it is important to note another scientific project devoted to multicode communication in social networks and conducted at the Institute of Applied and Experimental Linguistics [13, 14].

One of the tasks set within the framework of the project was the collection of materials necessary for research—multimodal texts stored on the Internet. To describe the materials, a special system of deep annotation was introduced [15].

When the database size reached nearly 1500 multimodal texts, the capacity of cloud data storage in which the database was located began to come to an end, and the question arose of finding a solution to get out of this situation. The solution was found, and it consisted in the following—to create a program with a friendly and intuitive interface for the end user, capable of performing the following actions:

1. Access the Microsoft Azure cloud database, view the materials stored in it and their description (should be displayed on the screen in the form of a table), edit them, provide the user with an easy way to download new materials, and delete old or no longer relevant materials.
2. Similar actions should have been available for another cloud storage—Google Drive, on which, due to its versatility and accessibility to ordinary computer users, the project database was stored [16].
3. In case the free space in one of the repositories came to an end, the program should have allowed the user to export data from one repository to another, while maintaining their integrity in both cases.

The finished DataTravel software product was made using WPF graphics technologies developed by Microsoft (Fig. 2).

It contained all the controls necessary for the implementation of the conceived functionality:

- To implement switching between Microsoft Azure and Google Drive cloud databases, two standard Microsoft Visual Studio RadioButton elements were placed on the main window of the program, each of which was responsible for connecting its own data source.

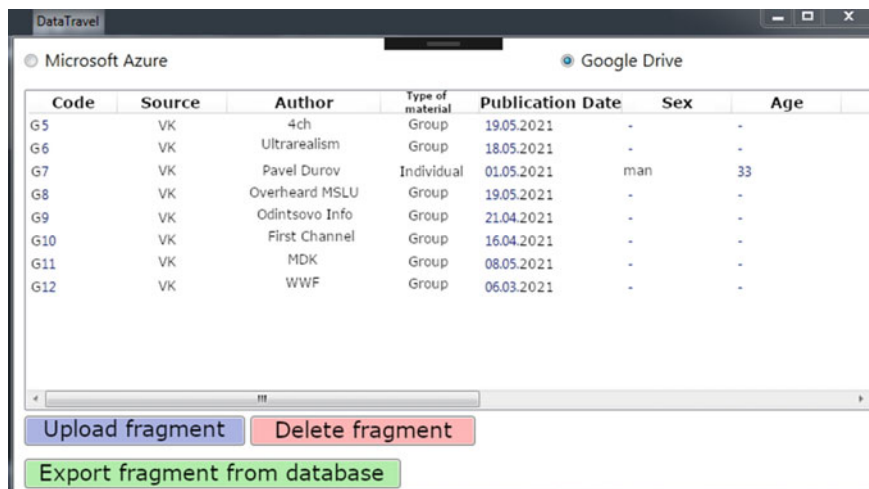


Fig. 2 Interface of the main window of the data travel application

- Immediately below the switches indicated above, there was a field for displaying materials from cloud databases, implemented using such a WPF element as ListView. The advantages of such information output were that, firstly, the information was displayed clearly and in a structured way, and secondly, a minimum of program code was required to implement such functionality.
- Finally, three buttons were placed even lower (“upload fragment,” “delete fragment,” “export fragment from database”), the functionality of which was described by their names. All three buttons were applicable to the currently selected database.

The program described above was another step forward for researchers from the Institute of Applied and Experimental Linguistics. It made it possible to work with two cloud databases at once—Microsoft Azure and Google Drive. The main innovation was the export function, which implied the rapid movement of data from one cloud database to the other, which later allowed storing and processing more materials. In addition, this function would significantly increase the security of the analyzed data and prevent their loss due to a possible technical failure on the side of one of the services used.

2.3 Automated Replenishment and Annotation of Linguistic Databases

To organize full-fledged work with big data, the authors set themselves the task of developing two auxiliary research programs at once.

The first program will be aimed at quickly filling experimental databases with new information from the most popular Internet portals. This will significantly reduce the time spent on the formation of global databases. Thus, its functionality will be as follows:

1. Get access to the main sources of big data that already appeared in the scientific projects of Institute(Instagram Facebook, Twitter, and YouTube video hosting, as well as social networks VK, Facebook, Odnoklassniki, Instagram, and Twitter) [14].
2. Using the algorithms specified by the author, collect information about the publication of interest to the author in accordance with the system of deep annotation of polycode texts [15].
3. Connect to the Microsoft Azure and Google Drive, cloud databases described in the previous studies and upload information about the publication selected in one of the available sources to them.
4. Display, for reference purposes, the scheme of deep annotation at the request of the user of the program.
5. Display, at the user's request, information about the publication selected in one of the available sources for its subsequent download to Microsoft Azure or Google Drive.

The purpose of the second program is the analysis of information already collected and stored in databases, its sorting, grouping by selected parameters, as well as the construction of diagrams for a more visually descriptive representation of the available material.

The software solution common to the two projects listed above will also include the DataKey project and the DataTravel project. Together they will make a basic set of research tools for working with linguistic information contained on the Internet.

The program for primary database analysis is called DataAnalyzer, and the program for filling experimental databases with new information is DataDownloader. To combine these two projects, as well as the previously created projects into a single software package, a DataManager solution was created.

The main window of the DataDownloader project enables to implementation of all the intended functionality for uploading information to the database (Fig. 3).

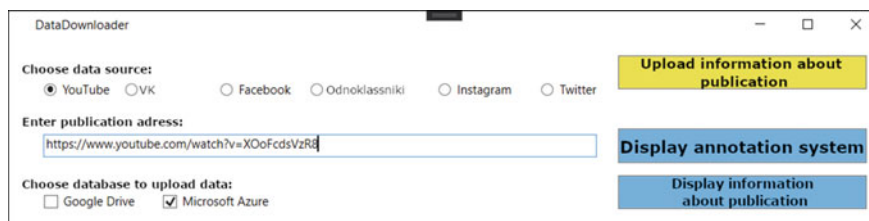


Fig. 3 Interface of the main window of the DataDownloader application

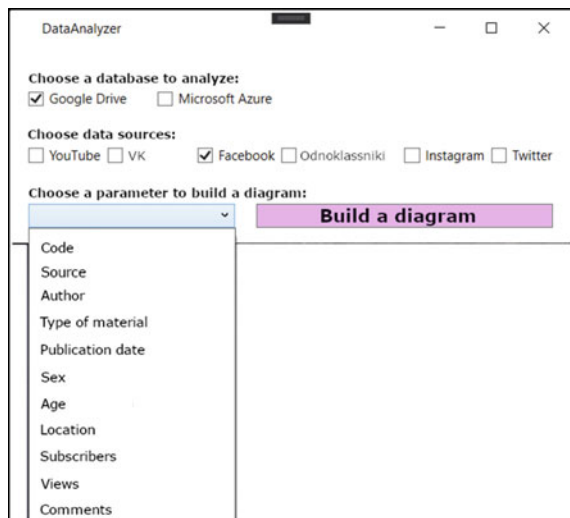
To select the big data source from which information will be uploaded to the cloud database in accordance with the deep annotation system [15], switches were implemented using the RadioButton element standard for Microsoft Visual Studio. In addition, immediately below the data source, there is a text field for entering a link to the publication, information about which needs to be downloaded. Finally, even below the program offers to select a database to download this information. It is worth noting that the program uses the CheckBox element to load it both into one of the databases and both at once. To move further through the databases of the information received, it is proposed to use the DataTravel application.

The functionality of the buttons in DataDownloader is also simple and clear. When one clicks on “upload information about publication,” the information is loaded by the entered link from the selected data source to the specified database. The “display annotation system” button allows the researcher to quickly access the flowchart of the deep annotation system for reference purposes and to visualize the way to store information about the publication in each of the databases. The “display information about publication” button also has a reference function and serves to display already processed information about the selected publication, which when clicked on “upload information about publication” will be recorded in the selected database.

Let us move on to the DataAnalyzer project (Fig. 4).

This project was also implemented in full in accordance with the tasks set. With the help of CheckBox elements, the user is asked to perform initial filtering of database entries by the databases themselves, as well as by the data sources required for specific research purposes. The ComboBox element, which is located below, gives the researcher the capability to select one of the parameters set by the deep annotation system and to analyze the information recorded by this parameter for each entry into

Fig. 4 Interface of the main window of the DataAnalyzer application



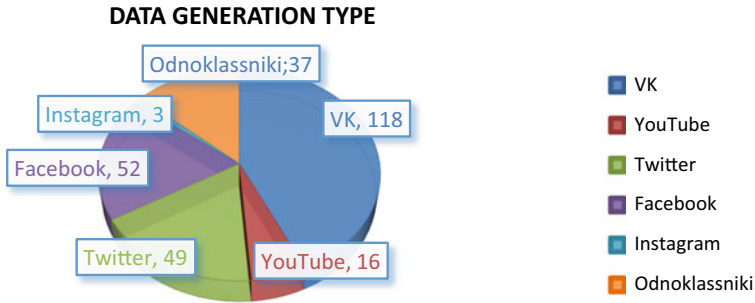


Fig. 5 Diagram based on the “data generation type” parameter

the database. The results of the analysis are reflected in diagrams [17]. The task of starting the analysis process is assigned to the “build a diagram” button.

Let us take a closer look at the examples of constructed diagrams for a better understanding of the principle of their functioning (Fig. 5).

For example, the “data generation type” diagram shows that out of 275 entries, 49 publications were typed into Twitter, 16 publications were uploaded from YouTube, 118 publications from VK (Russia’s largest social network), 37 publications from the slightly less popular but also very well-known Odnoklassniki social network, 3 publications from Instagram and 52 publications from the worldwide social network Facebook.

3 Results

Bringing together the aforesaid, the following set of tools for working with linguistic databases DataManager can be obtained:

- **DataKey**—software to work with cloud databases with Microsoft Azure that allows downloading information (both written and spoken texts) to the cloud to filter it according to the specified parameters, to display additional information about the sources of data, and to remove information from the cloud.
- **DataTravel** is a software package for data exchange between Google Drive and Microsoft Azure cloud databases, which is especially effective when there is a lack of free space in one or another storage.
- **DataDownloader** is a software for automated online downloading of information about a particular publication from big data sources to Google Drive or Microsoft Azure in accordance with the system of deep annotation of multimodal polycode texts [15].
- **DataAnalyzer**—software for the primary analysis of databases and the construction of visual pie charts displaying database statistics on a particular parameter of the deep annotation system.

In the course of the investigation described in the paper, it was possible to create a set of programs for big data collecting, storing, analyzing, and classifying. The software package combines all the latest developments in big data processing within the framework of linguistic research.

4 Conclusion

Despite the fact that big data is gaining the greatest popularity in the commercial sphere of society, they can also be of great assistance in conducting linguistic research. Nevertheless, it is important to understand that data science is very difficult, and to master it requires deep knowledge of both methods of mathematical statistics and programming. Using the example of scientific projects conducted at the Institute of Applied and Experimental Linguistics, one can make sure that in order to obtain the most objective results, and it is important for a scientist to have in their arsenal good tools to collect and analyze impressive arrays of data (for example, materials from social networks).

Such tools have been developed within this study. Together with the already existing programs presented in the previous years, it allowed to solve the problems of big data collection, storage, analysis, and classification.

It is assumed that the created software package will be a good tool for data collection and analysis for future research projects at the Institute of Applied and Mathematical Linguistics.

Today, one of the tasks facing applied linguistics is to improve algorithms for big data acquisition, processing, storage, and analysis. And over time, the importance of this task will only grow. By publishing this study, its authors try to open the door to the world of new and modern capabilities for linguistic science. Therefore, it is very important not to stop there and continue to work in this area.

Acknowledgements The research is supported by the Russian Science Foundation, grant No. 22-28-01050.

References

1. Aronova E, Baker K, Oreskes N (2010) Big science and big data in biology: from the international geophysical year through the international biological program to the long-term ecological research program, 1957-present. *Hist Stud Nat Sci* 40(2):183–224
2. Mainzer K (2014) *Die Berechnung der Welt: Von der Weltformenzu Big Data*. C.H. Beck, Munich
3. Reichert R (2014) *Big data. AnalysenzumdigitalenWandel von Wissen, Macht und Ökonomie*. Transcript Verlag, Bielefeld
4. Floridi L (2012) Big data and their epistemological challenge. *Philos Inform* 25:435–437
5. Bollier D (2012) *The promise and peril of big data*. The Aspen Institute, Washington, DC

6. Boyd D, Crawford K (2011) Six provocations for big data. In: A decade in internet time: symposium on the dynamics of the internet and society. Oxford Internet Institute, Oxford
7. Potapova RK (2017) Speech communication in information space. URSS, Moscow (in Russian)
8. Potapova R, Bobrov N (2015) Versatile linguistic databases annotation: practical issues and a new flexible approach. In: Proceedings of the 17th International conference SPECOM 2015 (Athens, Greece, September 20–24, 2015). University of Patras Press, Patras pp 45–53
9. Potapova R, Potapov V, Komalova L, Dzhunkovskiy A (2019) Some peculiarities of internet multimodal polycode corpora annotation. In: Salah AA, Karpov A, Potapova R (eds) SPECOM 2019. LNCS (LNAI), vol 11658. Springer, Cham, pp 392–400
10. Potapova R, Potapov V (2019) Some elaboration methods for written and spoken multilingual databases. Moscow University Bulletin 3 Series 9 Philology. Moscow University Press, Moscow, pp 71–91
11. Potapova R, Potapov V, Bazhenova I (2015) Development of the research cloud technology stand-alone system (regarding integrated speech databases). In: Proceedings of the 17th International conference SPECOM 2015 (Athens, Greece, September 20–24, 2015). University of Patras Press, Patras, pp 1–7
12. Microsoft Azure, Documentation. <http://azure.microsoft.com/ru-ru/documentation>. Last accessed 26 Aug 2021
13. Potapova R, Potapov V (2020) Some comparative cognitive and neurophysiological reactions to code-modified Internet information. In: Karpov A, Potapova R (eds) SPECOM 2020. LNCS (LNAI), vol 12335. Springer, Cham, pp 399–411
14. Potapova R, Potapov V, Lebedeva N, Karimova E, Bobrov N (2020) The influence of multimodal polycode Internet content on human brain activity. In: Karpov A, Potapova R (eds) SPECOM 2020. LNCS (LNAI), vol 12335. Springer, Cham, pp 412–423
15. Potapova R, Potapov V (2019) On the deep-parametric annotation method for the database of the Russian language polycode social network discourse. In: Historical destinies and modernity: VI International Congress of Researchers of the Russian Language: Proceedings and Materials, vol 1. Moscow University Publishing House, Moscow, p 224 (in Russian)
16. Google Cloud Platform, Documentation. <https://cloud.google.com/docs/#storage>. Last accessed 6 Oct 2021
17. The Data Science Venn Diagram. <http://drewconway.com/zia/2013/3/26/the-data-science-venn-diagram>. Last accessed 30 Sept 2021

Gray Scale Image Enhancement with CPSO Algorithm for Medical Applications



Mani Kumar Jogi and Y. Srinivasa Rao

Abstract In an image processing system, image enhancement plays a crucial role. Improvement of image quality by maximizing the information in the given input image is the main aim of this paper. Adaptive histogram equalization and histogram equalization are most popular non-heuristic or classical techniques for image enhancement by preserving main features of the input image. These techniques failed in achieving good quality enhancement. Optimization techniques have been proposed for enhancement of image problem. The quality is enhanced by selecting the optimal parameters based on objective function formulated during optimization process. The formulation of objective function plays an important role in optimization problem. This paper offers an effective objective approach for image enhancement using constriction factor-based particle swarm optimization (CPSO) algorithm. The proposed algorithm has been tested on medical images like knee cyst, spine MRI, and liver tumor. The proposed algorithm is successful in improving the quality of these images.

Keywords Image enhancement · Constriction factor-based PSO algorithm · Image quality enhancement · PSNR · RMSE

1 Introduction

Image processing has numerous applications in daily life, i.e., medical field and industrial and is a wide and active area of research in computing. Image processing techniques are one which can be treated as transforming one image to another image to develop the perception. For image enhancement, a genetic algorithm was proposed in [1] through contrast enhancement using a multi-objective function. Certain image quality measures have been projected recently. Contrast enhancement of digital images by preserving the mean image intensity using PSO has been proposed in [2, 3].

M. K. Jogi (✉) · Y. Srinivasa Rao
Department of Instrument Technology, Andhra University, Visakhapatnam, A.P., India
e-mail: jogimanikumar@gmail.com

During the optimization process, the algorithm search for optimal parameters based on the objective function. The objective function contains two or three performance measures which return a value during evaluation of objective function. The aim of the optimization is to maximize objective function that should produce an enhanced image. Hence, all performance measures are improved during the optimization process. After image enhancement, quantitative evaluation has conducted with the help of image quality metrics such as PSNR, RMSE, and MSSIM. High-PSNR value, RMSE value close to zero, and MSSIM value close to one indicate the eminence of the image enhanced.

Constriction factor-based PSO algorithm (CPSO) in which objective functions will hunt for the optimal set of gray levels. CPSO algorithm is the novel optimization technique which can be simply understood and easy to implement for various engineering optimization problems. Image enhancement is done using parameterized global intensity transformation function in which the parameters are to be optimized using CPSO algorithm considering number of objectives such as edge intensity, no. of pixels, entropy, and PSNR of the image in a multi-objective function. Three different cases have been conducted using CPSO algorithms in order to assess the effect of objective function on image enhancement. The proposed approach is tested on images like knee cyst, spine MRI, and liver tumor which offered excellent results. In order to validate the outcomes, quantitative, detailed qualitative, and statistical analyzes have also been presented.

2 Gray Level Image Enhancement

The conversion of the test image into the improved quality output image is the main aim of the image enhancement. Without losing original properties of the input image, the quality can be improved by various techniques for better visual judgments and machine understanding. The original value of each and every pixel is replaced by its neighborhood pixel value in spatial operations. This process might suffer from excessive enhancement of the disturbance in the input image over smoothing the image where that needs high-pitched details. Root filtering, linear filtering technique, and homomorphic filtering technique fall under change operations based on inverse transformation operations of the converted image. Gray scale image is artificially colored using a suitable color map in pseudo coloring methods. Due to non-unique features of the color maps, numbers of trails are essential to select an appropriate mapping [4]. Local enhancement transformation function is shown below in Eq. (1).

$$g(x, y) = \frac{G}{\sigma(x, y)}(f(x, y) - m(x, y)) \quad (1)$$

where $m(x, y)$ is mean and $\sigma(x, y)$ is standard deviation computed in a area centered at (x, y) having $M \times N$ pixels. G is the global mean of the input image, $f(x, y)$ intensity of the input and $g(x, y)$ is output image of pixel at (x, y) [5, 6]. In this approach, factors

like locality and compliance of the method are taken into account unlike classical global enhancement techniques. It is derived from Eq. (1) and is applied to every pixel at (x, y) of image is given below.

$$g(x, y) = \frac{k.G}{\sigma(i, j) + b} [f(x, y) - c \times m(x, y)] + m(x, y)^a \tag{2}$$

where $b \neq 0$ permits for zero standard deviation in the area, $c \neq 0$ permits for only fraction of the mean $m(x, y)$ to be deducted from unique. The final term influence has brighten and smooth the belongings on the image [7]:

$$m(x, y) = \frac{1}{n \times n} \sum_{x=0}^{n-1} \sum_{y=0}^{n-1} f(x, y) \tag{3}$$

$$G = \frac{1}{M \times N} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \tag{4}$$

$$\sigma(x, y) = \sqrt{\frac{1}{n \times n} \sum_{x=0}^{n-1} \sum_{y=0}^{n-1} (f(x, y) - m(x, y))^2} \tag{5}$$

Correct tuning of $a, b, c,$ and k parameters in Eq. (2) will yield large variations in the treated output image by protecting its originality. In this paper, CPSO task is to produce better enhanced image for the given input image using global intensity transformation function based on combination of different objectives.

3 Formulation of Objective Function

In order to estimate the superiority of output image without human involvement, an objective function is required that compares vital image performance parameters, i.e., no. of edge pixels, entropy of whole image and its intensity [4]. Certain authors [8] excepted entropy of entire image in their objective function. Entropy is important quality procedures in image enhancement. The ultimate quality of enhanced image has been quantified by using image quality metrics [9]. Hence, PSNR is considered as vital objectives in the objective function.

$$OF_1 = F(I_e) = \log(\log(E(I_S))) \times \frac{n_{edges(I_s)}}{M \times N} \times H(I_e) \tag{6}$$

$$OF_2 = PSNRI(I_e). \tag{7}$$

Equation (2) is $F(I_e)$. The sum of edge pixel intensities of the output image which can be derived by Sobel edge detector is $E(I_s)$ which results in output is $n_{edges} \cdot H(I_e)$ is entropy value. Horizontal pixels are represented by M and vertical pixels by N .

The aim of CIPSO is to search for better solution (a, b, c , and k) that maximizes OF. The purpose of weight factors is to convert individual objectives into a single objective. In this work, two scenarios have been tested for image enhancement problem using CIPSO algorithm in order to assess the importance of objective function. The two cases are as follows:

- Case 1: Single objective, i.e., OF_1 .
- Case 2: Single objective, i.e., $OF_2 = \text{PSNR}$.

4 Particle Swarm Optimization Algorithm

The PSO algorithm offers benefits to the challenging optimization problems [10–12]. PSO can adapt solutions to the changing environments and provide robust response to changing conditions [13]. PSO can solve compound optimization problems [8, 9, 14, 15].

Let the position of i th particle at the t th iteration can be given as

$$x_i^t = (x_{i1}^t, x_{i2}^t, \dots, x_{id}^t) \tag{8}$$

and $pbest$ of the i th particle can be given as

$$pbest_i^t = (pbest_{i1}^t, pbest_{i2}^t, \dots, pbest_{id}^t) \tag{9}$$

the $pbest$ of the current iteration is to be recorded after comparison

$$pbest_i^{t+1} = \begin{cases} pbest_i^t & \text{if } f_i^{t+1} \geq f_i^t \\ x_i^{t+1} & \text{if } f_i^{t+1} \leq f_i^t \end{cases} \tag{10}$$

The $gbest$ which is overall best is given as

$$gbest_i^t = (gbest_{i1}^t, gbest_{i2}^t, \dots, gbest_{id}^t) \tag{11}$$

Determination of $gbest$ which is $pbest$ among all is given as

$$pbest_i^{t+1} = \begin{cases} gbest_i^t & \text{if } f_i^{t+1} \geq f_i^t \\ pbest_i^{t+1} & \text{if } f_i^{t+1} \leq f_i^t \end{cases} \tag{12}$$

The i th particle velocity is given as

$$v_i^t = (v_{i1}^t, v_{i2}^t, \dots, v_{id}^t) \tag{13}$$

Every particle varies its position according to its current velocity as

$$x_{id}^{t+1} = (x_{id}^t + v_{id}^{t+1}) \tag{14}$$

where x_{id}^t is current position of i at iteration t .

To update velocity of particle in, there are some prospects to enhance the velocity is used in this technique. One factor is proposed here which can enhance the velocity.

4.1 Constriction Factor-Based PSO Algorithm (CPSO)

The basic PSO does not ensure convergence of a particle toward its attractors. The main benefit of this practice is that without altering the basic eqn., a constriction factor (χ) constructed on acceleration constants C_1, C_2 is merged in basic PSO algorithm to update the velocity.

$$\text{constriction factor}(\chi) = \frac{2}{\left|2 - \emptyset - \sqrt{\emptyset^2 - 4\emptyset}\right|}, \quad \text{where } \emptyset = c_1 + c_2, \emptyset > 4 \tag{15}$$

The velocity equation of the i th particle with χ in basic PSO algorithm can be expressed as

$$v_{id}^{t+1} = \chi[v_{id}^t + c_1 \text{rand}_1(pbest_{id} - x_{id}^t) + c_2 \text{rand}_2(gbest_d - x_{id}^t)] \tag{16}$$

4.2 Steps Implemented

1. Observe data, initialization of algorithm parameters and rapid generation of preliminary solutions by sustaining the constraints.
2. Determination of fitness value by Eqs. (6) and (7).
3. Determination of $pbest$, comparing with past iteration and choose lower value as $pbest$.
4. Determine $gbest$ which is $pbest$ among all in the current iteration, comparing with past iterations and select lesser value as overall $gbest$.
5. Update the velocity using Eq. (16) and position using Eq. (14) for every iteration.

Check constraints from following conditions

If $V_{id} > V_{max}$ then $V_{id} = V_{max}$, If $V_{id} < V_{min}$ then $V_{id} = V_{min}$.

1. When extreme iterations reached stop, otherwise back to second step.
2. $gbest$ is optimal $a, b, c,$ and k parameters.

Fig. 1 Implementation flowchart of CPSO algorithm

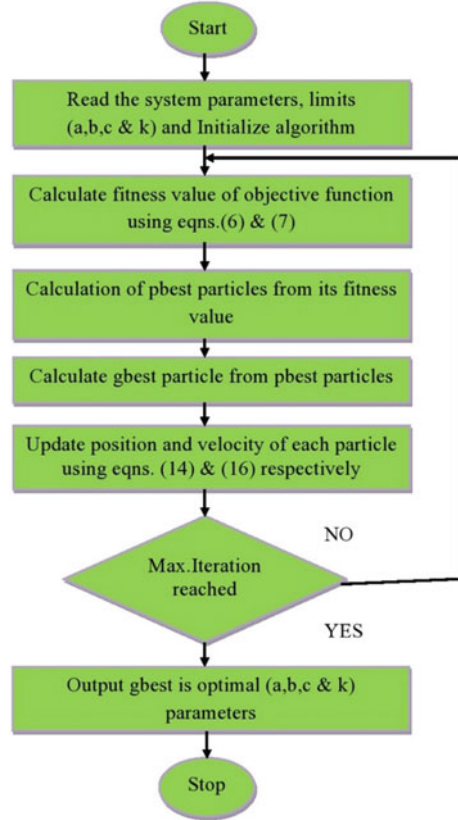


Figure 1 shows flowchart of CPSO algorithm, and Fig. 2 shows the input images and enhanced gray scale images of knee cyst, spine MRI, and liver tumor.

5 Results and Analysis

Figure 3 shows convergence characteristics of medical images with CPSO considering case 1 and case 2. The image quality metrics such as of fitness values, no. of edge pixels, entropy, PSNR, RMSE, MSSIM, and time taken for convergence of case 1 and case 2 are compared in Table 1.

Case 1: It is observed from Table 1, and no. of edge pixels and entropy are increased. The MSSIM value is near to unity. The convergence characteristics of CIPSO for knee cyst, spine MRI, and liver tumor images for case 1 are shown in Fig. 3a. The PSNR values for enhanced images range from 58 to 73. RMSE value and time taken for convergence are more in this case.







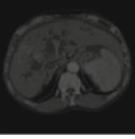
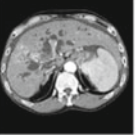
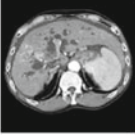
Name of the image	Input image	Output image	
		Case I	Case II
Knee cyst			
Spine MRI			
Liver Tumor			

Fig. 2 Enhancement of medical images with CPSO algorithm with case 1 and case 2

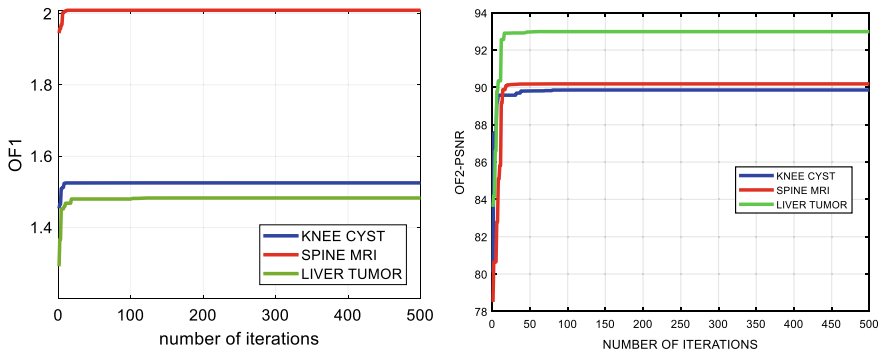


Fig. 3 Convergence characteristics of CPSO algorithm with case 1 and case 2

Case 2: It is observed from Table 1, and number of edge pixels and entropy are decreased compared to case 1. The MSSIM value is exactly unity. The convergence characteristics of CIPSO algorithm for knee cyst, spine MRI, and liver tumor images for case 2 are shown in Fig. 3b. The PSNR values for enhanced images range from 89 to 93. RMSE value is less, and convergence time is more in this case.

Table 1 Comparison of fitness values, no. of edge pixels, entropy, PSNR, RMSE, MSSIM, and time of case 1 and case 2

Image name	Image details	Method	Fit. value		Edge pixels		Entropy		PSNR	RMSE	MSSIM	Time (s)
			Init.	Final	Init.	Final	Init.	Final				
Knee cyst	Format: Tif Dimension: 512 × 512	Case I	1.24	1.52	4702	5798	7.17	7.21	62.074	0.2016	0.9942	241.20
		Case II	1.24	89.8	4702	4635	7.17	7.32	89.861	0.0107	1	10.40
Spine MRI	Format: Tif Dimension: 512 × 512	Case I	1.73	2.00	4644	5311	7.55	7.70	72.656	0.0596	0.9971	233.50
		Case II	1.73	90.1	4644	4883	7.55	7.59	90.181	0.0127	1	10.05
Liver tumor	Format: Tif Dimension: 512 × 512	Case I	0.07	0.72	3894	3785	5.92	5.99	58.147	0.3169	0.9904	242.29
		Case II	0.07	92.9	3894	3936	5.92	5.94	92.989	0.0079	1	10.31

6 Conclusion

In this paper, constriction factor-based PSO algorithm is proposed for the gray level image enhancement of medical images like knee cyst, Spine MRI, and liver tumor. Simulations are carried out for two objective functions formulated to test the image quality. One objective function is completely based on the PSNR value. Several simulation tests were performed for different medical images to investigate the convergence and quality images. The image quality metrics validate the effectiveness of CPSO algorithm in enhancing the image quality. Simulation results confirm that CPSO with case converges faster with enhancement of quality of medical images.

Appendix

CPSO algorithm parameters:

P : 20, n_{par} : 3, $Iter_{max}$: 50, C_1 , C_2 : 2, 2, Cfactor: 0.729, W_{max} : 0.9, W_{min} : 0.4

References

1. Merugumalla MK, Prema Kumar N (2017) Sensorless control of BLDC motor using bio-inspired optimization algorithm and classical methods of tuning PID controller. *J Instrum Control Eng* 5(1):16–23
2. Chen H, Tian (2011) Using particle swarm optimization algorithm for image enhancement. In: *International conference on uncertainty reasoning and knowledge engineering*, vol 1, pp 154–157
3. Coelho LDS, Sauer JG, Rudek M (2009) Differential evolution optimization combined with chaotic sequences for image contrast enhancement. *Chaos, Solitons Fractals* 42(1):522–529
4. Hanmadlu M, Arora S, Gupta G, Singh L (2013) A novel optimal fuzzy color image enhancement using particle swarm optimization. In: *International conference on contemporary computing*, vol 2, pp 41–46
5. Eberhart RC, Shi Y (2000) Comparing inertia weight and constriction factor in particle swarm optimization. *Int J Power Electron* 7(1/2):16–35
6. Huynh DC, Dunnigan MW (2012) Advanced particle swarm optimisation algorithms for parameter estimation of a single-phase induction machine. *Int J Model Ident Control* 15(4):227–240
7. Ilir J, Petraq M (2015) PID design with bio-inspired intelligent algorithms for high order systems. *Int J Math Comput Simul* 9:44–52
8. Adewumi AO, Arasomwan MA (2016) On the performance of particle swarm optimisation with(out) some control parameters for global optimisation. *Int J Bio-Inspired Comput* 8(1):14–32
9. Mauro SI, Johann R (2011) Particle swarm optimization with Inertia weigh and Constriction factor. In: *International conference on swarm intelligence*, Cergy, France, pp 1–11
10. Merugumalla MK, Prema Kumar N (2017) Optimized PID controller for BLDC motor using nature-inspired algorithms. *Int J Appl Eng Res* 12(1):415–422

11. Kwok NM, Ha QP, Liu D, Fang G (2009) Contrast enhancement and intensity preservation for gray-level images using multi objective particle swarm optimization. *IEEE Trans Autom Sci Eng* 6(1):145–155
12. Lei X, Hu Q, Kong X, Xiong T (2014) Image enhancement using hybrid intelligent optimization. *Opt Optoelectron Technol* 4:341–344
13. Draa A, Bouaziz A (2014) An artificial bee colony algorithm for image contrast enhancement. *Swarm Evol Comput* 16:69–84
14. Alberto AP, Michael F, Chunjiang Q (2009) Particle swarm optimization for PID tuning of a BLDC Motor. In: International conference on systems, man and cybernetics, October 2009, San Antonio, TX, USA, pp 3917–3922
15. Altinoz OT, Erdem H (2015) Particle swarm optimisation-based PID controller tuning for static power converters. *Int J Power Electron* 7(1/2):16–35

Domain-Specific Chatbot Development Using the Deep Learning-Based RASA Framework



Vijay Kumari, Chinmay Gosavi, Yashvardhan Sharma, and Lavika Goel

Abstract Conversational agents are actively gaining popularity in research because of their ability to imitate human responses in almost every domain. As there are many research enhancements in deep learning models, it becomes challenging to incorporate all these enhancements while developing a conversational agent. One of the main advantages of conversational agents is their ability to answer frequently asked queries without any human involvement and automatically generate the conversation's story flow. In any educational institution, it becomes difficult for the teaching and non-teaching staff to answer all the students' queries regarding the course, exam, and other information regarding their daily activities in the institute. Using the deep learning framework, we developed a chatbot to answer various questions related to the education domain, such as exam(timetable, venue) and course-related queries(course handout). The questions are answered by querying databases which can be updated via an administrator's web browser. The system will first create intents for the use cases and entity recognition mechanisms after connecting the deep learning framework to the database using custom actions. We had created a user interface to allow updates to the database for exam timetable and course information via either file upload or a web page.

Keywords Natural language understanding (NLU) · RASA-NLU · RASA-Core · LSTM

V. Kumari (✉) · C. Gosavi · Y. Sharma
Birla Institute of Technology and Science, Pilani 333031, India
e-mail: p20190065@pilani.bits-pilani.ac.in

C. Gosavi
e-mail: h20200262@pilani.bits-pilani.ac.in

Y. Sharma
e-mail: yash@pilani.bits-pilani.ac.in

L. Goel
Malaviya National Institute of Technology, Jaipur, India
e-mail: lavika.cse@mnit.ac.in

1 Introduction

Chatbots have become popular in recent years and deployed to solve various problems such as solving customer queries and educating customers regarding the products. In many cases, they act as the first point of contact, avoiding the need for actual human communication based on the use case. They collect valuable data before human-to-human interaction happens, thus saving time and being considered helpful to customers as it shows they prefer self-service as the first option rather than engaging with agents. While the chatbots were first introduced to the broader audience through social media websites, they were used in many domains such as e-commerce sites, banks, schools, or anywhere we can expect customer interaction.

Chatbots are the main application of natural language processing and artificial intelligence, which understands the human language and responds in the same language [13]. Students of any university or college need to access information related to their course work like class timing, classrooms for a particular course, syllabus, exam date, exam time, and a course handout. Particularly during this pandemic, when it is hard to have a physical meeting, a chatbot is expected to respond to each query rapidly, accurately and is accessible whenever it is needed. Functionally, this chatbot is helping students and instructors in many things, such as scheduled time of a course, course handout, students' grade information, and scheduled makeup class.

This chatbot is built using the open-source RASA framework [2], a deep learning model trained over the required data to replicate the required behavior. RASA is a tool to develop custom artificial intelligence-powered chatbots using Python and natural language understanding (NLU). The developed chatbot uses the long short-term memory(LSTM) network [7], which was already integrated into the RASA. The reason for using the LSTM is its effectiveness in using memory. Building the chatbots using the deep learning models requires a framework for unifying and pipelining the deep learning model.

We have already seen how chatbots are becoming common and the various use cases are being applied in. One such use case can be a university chatbot. The university setting has much potential for chatbot use—prospective and current student queries related to various topics such as admissions, timetables, and courses. To create the chatbot, we rely on the RASA, an open-source deep learning-based framework for creating conversation chatbots and linking them to third-party services. The use cases which will be demonstrated will be providing exam timetable information and course information. The created chatbot can then be used on the university website or the app as RASA supports both mediums. This can facilitate in reducing the load on universities while responding to queries, collecting data, etc. Similarly, over time, we can add more queries and train the chatbot based on user responses, thus increasing the usefulness of the chatbot.

2 Related Work

While the first chatbot was being developed in 1966, we can point to the chatbots deployed on the Chinese messaging/social media app to introduce the general public. Nowadays, however, they have become very ubiquitous and can be found on most websites, chat-based games, can be embedded into popular messaging apps, mobile platforms such as Android and iOS, provide native chatbot support (Google Now, Siri, Bixby, etc.) or via third-party applications (Soundhound, etc.).

The primary motivation of commercial chatbots was to reduce the human workload and provide services to customers 24×7 without incurring high costs [5]. On the research side, the motivation of chatbots has been the creation of a conversational user agent that responds naturally to queries. Over time many more aspects such as speech recognition, context recognition (responding not just based on the current question but also the previous conversation), etc., can be considered research interests.

The conversational question answering system can be built using either the rule-based or the deep learning-based approach to train a model on the specific data. The user speech intent is considered while generating the response with the trained model [4]. One category can be using the knowledge base, which can be given in any structured or unstructured form. The chatbots try to answer the query using these knowledge bases [10]. Eaglebot chatbot utilizes the latest transformer-based BERT model for retrieving the answer, which has a three-route base selection method using the dialog flow [12]. The model ‘ranking-based question answering system with web and mobile application’ utilizes the effectiveness of BERT model for answering the user query in a specific domain. The question can be asked through the web interface or using the android-based application [1].

The paper [11] describes an application for a chatting system for use cases such as attendance, marks, and notices. However, rather than relying on natural language understanding (NLU) techniques to allow a variety of user inputs, it uses pattern matching and then queries the database based on the user input. However, there is no mention of context; hence, a natural conversation will not be possible. Similarly, fuzzy pattern matching is not mentioned, so the chatbot becomes more restricted. Paper [8] improves on the above and introduces RASA-NLU and thus can be used for intent identification and entity recognition despite pattern not precisely matching. The entities recognized were used to call third-party APIs. However, no database was introduced in work, which is necessary for the university setting.

Erasmus chatbot, which uses cloud-based services, is an AI-powered chatbot developed to answer the query of students of any institute [14]. The paper [15] describes the development of a chatbot deployed on Facebook for an Indonesian university. It covers specific use cases such as greetings, schedules, grades, information requests, and weather forecasts in the area. The chatbot uses RASA version 1.1.4 and MySQL as a database for storing schedules and other similar information. However, RASA mechanisms to introduce context (by using slots mechanism) are not used here.

Chatbot development framework stays a provoking research problem because of the absence of data for training the model, so to solve this issue, a conversational question generation is used to generate questions for training and assessment purposes [6].

3 Proposed Model

Generally, chatbots use strict rule-based methods like pattern matching or word matching to make a conversation that becomes quite restrictive if we want to have a natural flow of conversation or a large set of rules. In our work, we have used natural language understanding (NLU) approach, which tries to understand the sentence rather than just matching based on some rules. It gives flexibility if the user makes small mistakes in the sentence or the sentence structure is different from training data. We can still classify the input to a relevant intent based on the calculated confidence. The chatbot's knowledge base can also be updated in the back end, and all the updations are reflected without any retraining of the bot. Unlike some other chatbots, it is designed to improve over time as we can tag the new inputs from the users to the relevant intents.

3.1 Model Development Overview

Initially, the use cases that will be supported based on them, the intent, and entity for each use case are decided and created. We have decided that the chatbot will answer topics related to the exams and courses offered by the university. The exam-related question can be asking about exam dates, timings, timetable, etc. Similarly, course-related questions can ask about course instructor name, evaluation components, course objective, and assignment information.

The training data has to be created for all intents with the entities manually tagged in them. We are using Conditional Random Fields(CRF) [9] for entity extraction, and they require manual tagging of entities. The training data has to be created individually for each intent. Suppose we are trying to create data for the intent where the user's question is regarding the scope of a particular course. We create training phrases such as—'What is the scope of the course CS G525?'. Similarly, we can add similar phrases with the same meaning to the above phrase, such as—'What is the objective of the course CS G525?'. We also include the phrase by not including the entity(class code in the above scenario) as the user might be asking the subject about a subject stored in the context, i.e., the user has previously mentioned the course code in the conversation and the bot remembers it from that conversation.

The data is stored in the YML file called nlu.yml as per the standard deep learning-based RASA structure. The entities are tagged as course-id and course name, allowing the user to enter either of those when asking about the instructor. For training the bot,

we include some examples of course IDs and course names. Based on the examples we provide, we have provided the names when the entity gets trained during the training process. Then we have to create rules and stories to create mappings between the intents and the actions. The actions are not written; we create unique names for each action later in the mapping process. A RASA rule is a simple mapping from an intent to action, i.e., when the NLU detects an intent, the action described in the rule takes place. These are included in the rules.yml as per the RASA structure.

However, action will be decided based on the current state and the previous inputs and output in real-life interaction. It can be achieved via RASA stories, where we design a flow of the conversation and decide which action to take based on our previous intents. For example, the initial intent can be to ask for ‘What is the objective, of course, CS G523?’. After sending a response, we can also ask, ‘Do you want to enroll in the course?’ and carry out the registration process based on the user response. So a story can be considered as a collection of rules happening in a particular order.

Based on the type of entity identified, we frame a SQL query and execute it on the correct database and table. Both the entity type and value are used to frame query; if we take the ongoing example of the course scope and the user has asked the question using course name, the entity type will be course name and whatever name of the course user provided in the query. SQLite database is selected, which contains the exam and course-related information populated via web user interface (Fig. 1) designed using Flask framework, using which the admins of the system can upload and update the exam timetable course-related information. Figure 1 shows a portal for updating the exam timetable, and similar portals are present for updating course information and regular timetable.

The image shows a web form titled "Admin Portal". It contains several input fields and a button:

- Input field: "Enter Exam Type - Mid Sem or Compre"
- Input field: "Enter Semester Type - Odd or Even"
- Input field: "Enter Year"
- Input field: "Enter Link For Time Table"
- Section: "Upload CSV File" with a "Browse..." button and the text "No file selected."
- Button: "Update Timetable"

Fig. 1 Portal for updating exam-related information

While the user asks the question, setting and erasing slots are also done in action. Continuing with the earlier example, if the user was asking about the scope of a particular course, a slot would be set holding the course information. Depending on whether the entity value or slot value is non-empty, we formulate the query. The priority is given to entity values and then to the slot values. Based on the query type, we update the corresponding slot value and erase the remaining slot values. Based on the query type and query value, the query of the following format is executed on the database.

SELECT * FROM table_name WHERE query_type = query_value

Here the table name is selected based on the intent while the query type are entities such as course-id and course name. The final term is the query value which is the actual value user provides in the current or the previous queries such as computer graphics and CS G515 etc.

After the query result is received from the database, the same is formatted and provided as a response via the custom action. The RASA bot is deployed on a website created using the Flask framework. RASA creates an endpoint for responding to the front end with which we communicate using REST. Thus, the data flow of the developed model is shown in Fig. 2.

How words are understood by the model

The input is first passed to the RASA-NLU to be converted into a vector format, for entities and intent is classification. The intent is classified based on which class received the highest confidence after the Dual Intent And Entity Transformer (DIET) classifier applied to the vectorized inputs. The score is calculated based on the phrases and sentence structure provided as the training data for each intent. Thus, we consider the entire sentence rather than just words and check which intent is the best choice for the input. Similarly, the entities are extracted based on what entity extractor is used. We have used the condition random fields in this work as we have domain-specific entities like course names and course ID. After having both the intent and entity, we map the functionality to be executed for each intent. The entity is passed as arguments, and the database is queried based on these arguments to get the result relevant to the intent of the question asked by the user. Thus, the combination of intent and entity identification and a mapping to the action after identifying the intent allows the system to understand and respond to words.

3.2 RASA Architecture Overview

RASA includes natural language understanding (NLU) and RASA-Core as a part of the deep learning framework, whereby NLU tries to understand the user's response in the chat, and RASA-Core is responsible for deciding the actions based on the user input. The final part of the framework is natural language generation (NLG), where the response is generated [2]. The overview of the architecture is given below.

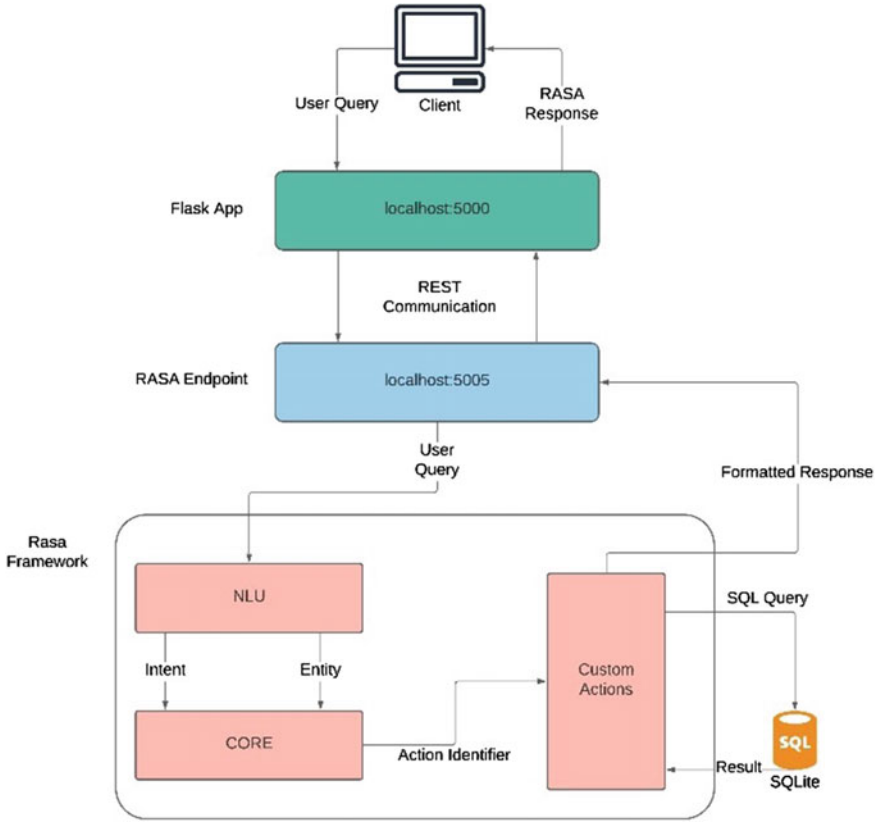


Fig. 2 Workflow of the developed model

Natural Language Understanding (NLU) The NLU is the part where the intent classification and entity extraction part take place. Intent can be considered as the objective of the input from the user. An entity can be considered as a term that is related to the intent. For example, in a university setting, we might expect the following question—‘Who is the instructor of the CSG523 course?’. The intent here is ‘request_user_instructor_name’ which indicates that user is requesting the name of the course instructor, and the entity is the course code which in this case was CS G523. We can have multiple entities in the same user query.

The process starts with vectorizing the user response, where the words are converted into embeddings. Word embeddings are the representation of user response in vector format. There are two types of embedding available in the intent classification process—supervised and pre-trained. We do not have any available domain-specific acronyms and pre-trained embedding, so we selected the supervised embeddings for the process of intent classification.

The next step is entity recognition, for that model has three options: spaCy, Duckling, and Conditional Random Fields (CRF). The spaCy option is suitable when the entity is of standard types like organization, city, or similar entities. Duckling is suitable for numerical information. In both of these, we need not annotate the training data for the process of entity recognition. However, the university domain has a variety of entities like course code, instructor name, and time. In these cases, it is better to annotate the training data to recognize such entities by using the Conditional Random Fields, where we manually annotate the entities in the training data itself. Continuing with the example of the course instructor, when we are inserting the line in the training data, it will be annotated as follows—Who is the course instructor of [CS G523] (course_code). A variety of such examples will be added to the training dataset so that entity can be extracted successfully when an unseen course code number arrives.

RASA-Core The core is the part of the chatbot development framework which decides the next action in the dialog after receiving the input and entity from the RASA-NLU. Similarly, the context is also considered as input. The output of the RASA-core part is a probability model which predicts the next action and sends the response based on the rules. In our chatbot model, the RASA-Core utilizes long short-term memory (LSTM) network for these predictions. As mentioned before, RASA allows the context to be considered input for the probability model, so storing this context for some steps in the dialog is enabled by LSTM or recurrent neural networks (RNN).

Natural language generation (NLG) The natural language generation module generates the responses, which are then sent to the user. These responses have to be structured as a template in which various components can be included. The main component is the 'text,' which is compulsory for every response; others include buttons as a choice, image, and attachments. The response can be in JSON format or something based on the front-end framework being used where it is going to be displayed.

RASA Actions In the core part, it was discussed that actions would be taken based on the input received from the NLU. Here we have decided that a database will be acting as a data source for any user query. Thus, the actions will include checking the recognized entities and framing a query based on those. The action will also include the process of connecting to the DB and retrieving results. Once the results are generated, they can be sent as a custom response described in the NLG section.

Dual Intent And Entity Transformer (DIET) The input sentences broken into the individual tokens by the pipelines are fed to the DIET architecture. The function of the DIET classifier is to identify the intent and entities from the input tokens [3]. It is the advantage of DIET since it is a multi-task transformer that can predict intent and entity together. DIET has flexibility in that it can be used with pre-trained embeddings such as BERT, GLoVe, and convert. So without writing any code lines, we can experiment with various types of NLU pipelines. Generally, many pre-trained language models already exist, but they are heavy to use and require

enormous computing power. They are not optimized to use with conversational AI software. The DIET architecture requires the dataset to have the input text, labels of intents, and labels of entities. The total loss is computed by the addition of entity loss, intent loss, and mask loss.

3.3 Model Training

Training has been carried out for each of the intent present in the chatbot. For each intent identifier, we include some examples of the type of sentences we want to get matched to that particular intent. For example, let us consider the `course_instructor_query` intent. The following is included as training data for the `course_instructor_query`

- **who is the course instructor?**
- who is the course instructor of [CS F111](course_id)
- who is the course instructor of [CS G513](course_id)
- who is the course instructor of [Computer Programming](course_name)
- who is the course instructor of [General Mathematics II](course_name)
- **name of the course instructor**
- name of the course instructor of [CS F111](course_id)
- name of the course instructor of [CS G513](course_id)
- name of the course instructor of [Computer Programming](course_name)
- name of the course instructor of [General Mathematics II](course_name).

Thus, if the input from the user contains the above phrases, the input will be identified as a query regarding the course instructor. We also include some entities (CS F111, network security, etc.) and their respective tags (`course_id` and `course_name`). Different entities are included so that the RASA-NLU can generalize for the inputs it might receive rather than just matching based on the training data. Thus, inputs outside of the training data are handled. Similarly, we check what queries the chatbot could not classify, and we can later tag them to particular intents. Thus, as the number of users increases, the chatbot becomes more accurate as of the number of phrases per intent increases.

4 Experiments and Results

The deep learning-based chatbots are relied upon to give the best results for chatbot improvement. Their performance evaluation of intent and entity classification is measured in precision, recall, and F1-score. The model is divided into various parts, and each part's accuracy can be tested separately. We have designed about 30 scenarios in daily conversation and the expected actions based on the input for testing purposes. A sample is also given in Fig. 3. Here we can see the query from the user

```

- story: Story from Conversation ID d8606be0664046e39297fc9bab9880e8
  steps:
  - intent: course_assignments_query
  user: |-
  What are the course assignments of [CS G523](course_id)?
  - slot_was_set:
  - course_id: CS G523
  - action: action_assignments
  - intent: course_chamber_query
  user: |-
  What are the chamber hours?
  - action: action_chamber
  - intent: course_lecture_plan_query
  user: |-
  What is the lecture plan in this course?
  - action: action_lecture_plan
  - intent: course_reference_query
  user: |-
  What are the reference books?
  - action: action_reference
  - intent: course_textbook_query
  user: |-

```

Fig. 3 Sample of story generation for testing purpose

and the expected action identifier from the set of identifiers defined while training the model.

A file of test stories is created for the evaluation of the model. These stories are different from those in the 'stories.md' file in that they also contain an example user message for each intent. The model is then evaluated using the command 'rasa test.' This command returns a confusion matrix each for intent classification and response selection. It also returns an intent report that logs each intent's precision, F1-score, and recall. The results of the action and intent classification are shown in Table 1. The true label and the predicted label analysis can be done from the above accuracy tables, and we can see that all intents were classified correctly. The tables are confusion matrices for the process of intent classification and entity recognition carried out by the NLU. In this case, we have a separate action for each entity. Thus, the accuracy for intent and actions will be the same due to such one-to-one mapping. They are both represented in Table 1. The use cases where minor spelling mistakes or sentence structure are slightly different from the training data are tested whether the NLU part can handle the variety of conversations we expect in real life. A similar analysis of the entities is shown in Table 2. Here analysis for only two entities is done only for course_id and course_name. Current testing was focused on the accuracy of identifying course ID; more test cases for course names can be added to check the accuracy. We have some queries in the testing data which do not contain any

Table 2 Entity_Recog_DIET_Classifier

Entity	Course_id	Course Name	No_entity
Course_id	26	0	0
Course Name	0	2	0
No_entity	0	0	174

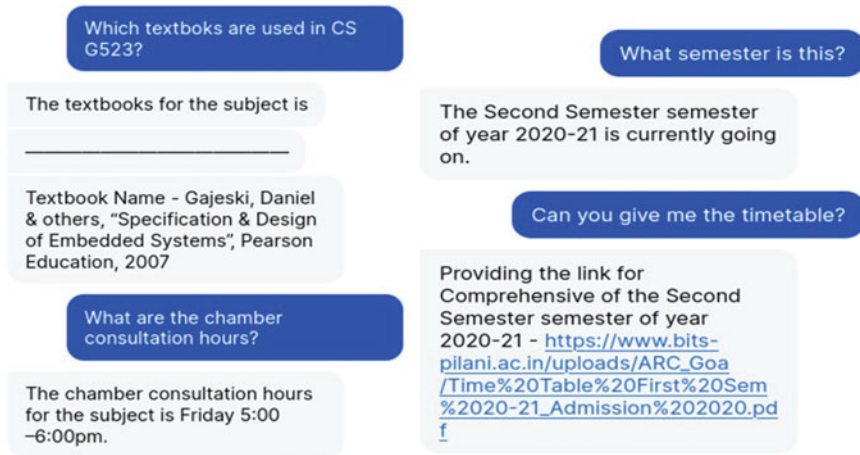


Fig. 4 Spelling mistake in the word ‘textbooks’ handled by NLU + intent classification by identifying semester information request intent

entities, for example—‘Which semester is going on?’. Such queries are annotated with ‘No_Entity’ label in Table 2 which is Entity Recognition table .

Figure 5 shows the setting of slots where the context of previous messages affects the new reply. Moreover, how small spelling mistakes do not cause any problems to the chatbot (the first message has a mistake) shown in Fig.4. Similarly, all the messages show DB connectivity. Figure 4 shows how intents are defined and their identification via the NLU and entity extraction process via the NLU.

5 Conclusion

The university setting provides many opportunities to use a chatbot, reducing loads on the working staff. The RASA framework provides an ideal option as it natively supports natural language understanding based on deep learning libraries. Similarly, it allows various responses like simple hard-coded responses, database retrievals, API calls, etc., thus increasing the degree of interoperability with existing systems. The created chatbot can respond naturally to questions based on covered domains, like

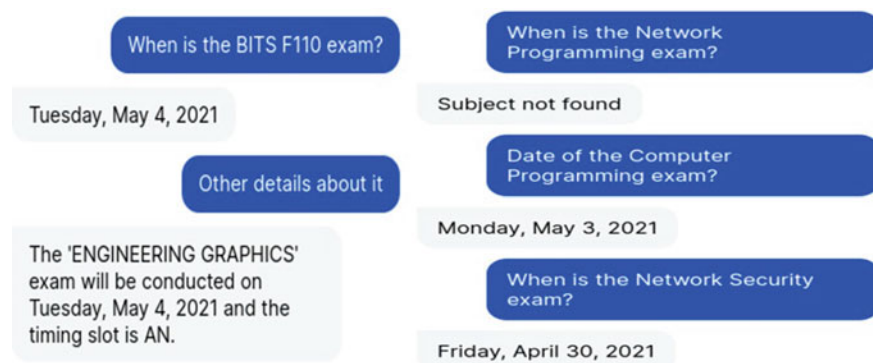


Fig. 5 Entity recognition as course name is extracted + slot mechanism where context of course ID is maintained over messages

exam timetables and course-related queries, so we can conclude that deep learning-based chatbots can give excellent performance in terms of intent and entity representation and give appropriate information reply based on each evaluation metrics.

This exploration is in the stage of improvement as we can explore the possibility of introducing a UI-based solution of adding more use cases into the chatbot and subsequent connection with the database rather than writing the code and making it more customizable. We can explore the possibility of fuzzy matching and discuss thresholds that allow some small user error—the user types in a slightly incorrect course name (e.g., user types in ‘Embedded’ instead of ‘Embedded’).

Acknowledgements The authors would like to convey their sincere thanks to the Department of Science and Technology (ICPS Division), New Delhi, India, for providing financial assistance under the Data Science (DS) Research of Interdisciplinary Cyber-Physical Systems (ICPS) Program [DST/ICPS/CLUSTER/Data Science/2018/ Proposal-16:(T-856)] at the department of computer science, Birla Institute of Technology and Science, Pilani, India. The authors are also thankful to the authorities of Birla Institute of Technology and Science, Pilani, to provide basic infrastructure facilities during the preparation of the paper.

References

1. Agarwal A, Kumari V, Sharma Y, Goel L (2021) Ranking based question answering system with a web and mobile application. In: 2021 11th International conference on cloud computing, data science & engineering (Confluence). IEEE, New York, pp 52–58
2. Bocklisch T, Faulkner J, Pawlowski N, Nichol A (2017) Rasa: open source language understanding and dialogue management. arXiv preprint [arXiv:1712.05181](https://arxiv.org/abs/1712.05181)
3. Bunk T, Varshneya D, Vlasov V, Nichol A (2020) DIET: lightweight language understanding for dialogue systems. arXiv preprint [arXiv:2004.09936](https://arxiv.org/abs/2004.09936), 2020
4. Csaky R (2019) Deep learning based chatbot models. arXiv preprint [arXiv:1908.08835](https://arxiv.org/abs/1908.08835)
5. Desai M (2019) Paving the way to a chatbot revolution

6. Gu J, Mirshekari M, Yu Z, Sisto A (2021) ChainCQG: flow-aware conversational question generation. arXiv preprint [arXiv:2102.02864](https://arxiv.org/abs/2102.02864)
7. Huang Z, Xu W, Yu K (2015) Bidirectional LSTM-CRF models for sequence tagging. arXiv preprint [arXiv:1508.01991](https://arxiv.org/abs/1508.01991)
8. Jiao A (2020) An intelligent chatbot system based on entity extraction using RASA NLU and neural network. In: Journal of physics: conference series, vol 1487. IOP Publishing, p 012014
9. Lafferty J, McCallum A, Pereira FCN (2001) Conditional random fields: probabilistic models for segmenting and labeling sequence data
10. Nagarhalli TP, Vaze V, Rana NK (2020) A review of current trends in the development of chatbot systems. In: 2020 6th International conference on advanced computing and communication systems (ICACCS). IEEE, New York, pp 706–710
11. Prashant BP, Anil MS, Dilip KM (2017) Online chatting system for college enquiry using knowledgeable database
12. Rana M (2019) Eaglebot: a chatbot based multi-tier question answering system for retrieving answers from heterogeneous sources using BERT
13. Teller V (2000) Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition
14. Thakkar J, Raut P, Doshi Y, Parekh K (2018) Erasmus-AI chatbot. Int J Comput Sci Eng 6(10):498–502
15. Windiatmoko Y, Rahmadi R, Hidayatullah AF (2021) Developing facebook chatbot based on deep learning using rasa framework for university enquiries. In: IOP conference series: materials science and engineering, vol 1077. IOP Publishing, p 012060

Pulse Shaper Design for UWB-Based Medical Imaging Applications



M. K. Devika Menon and Joseph Rodrigues

Abstract In this paper, a pulse shaping filter is designed to shape the higher-order derivatives of the basic UWB Gaussian pulse for efficient pulse transmission through human tissues for medical imaging applications. The shaped pulse for the desired center frequency fits the FCC mask and power spectral density (PSD) specifications with higher spectral efficiency being achieved. It is observed that the ringing effect of Gaussian pulse is reduced by using the proposed bandpass FIR shaping filter. The low ringing effect observed in the shaped pulse ensures better antenna power distribution and improved location accuracy which is critical factor for medical imaging applications. The pulses synthesized are highly orthogonal which aids in multi-access communication, improved bit error rate (BER) performance and short duration UWB pulses leading to higher data rate transmission. The drooping frequency response characteristics of the synthesized pulse have reduced clutter hence tightly focused image obtained for imaging applications.

Keywords Gaussian pulse · UWB pulse shaping filter · Ringing effect · Clutter

1 Introduction

Ultra wideband (UWB) is a radio technology that support a variety of medical applications like medical radar, microwave imaging, wireless communications with medical implants due to superior transmission in various medium and better image resolution. Ultra wideband is extensively employed in biomedical imaging for non-invasive tumor detection applications like early breast cancer detection due to its superior features of very narrow sub-nanoseconds pulses to transmit data with very low power level, no tissue ionization due to low power, high data transmission rate, and good localization properties [1–5]. The Federal Communication Commission (FCC)

M. K. Devika Menon (✉) · J. Rodrigues
CHRIST (Deemed to Be University), Bengaluru, India
e-mail: devika.menon@christuniversity.in

J. Rodrigues
e-mail: joseph.rodrigues@christuniversity.in

introduced spectral mask regulations due to the large bandwidth of UWB signals and to limit transmitting capacity below -41.3 dBm/MHz in the 3.1–10.6 GHz frequency range of UWB systems in order to reduce interference with adjacent wireless systems [2]. UWB technology implementations are divided into three parts, namely communications, sensors, position location, tracking, radar, and biomedical applications [6–10]. For all these applications, a suitably designed UWB pulse is required to improve the target response while canceling the background interference. Time varying channel bandwidth, transmission rate, delay, jitter, and loss in accuracy are the major challenges in video streaming while employing UWB signals [11–15]. UWB pulse shapes play an important role in efficient UWB system applications since signal transmission in majority impulse radio (IR) UWB radio systems employ direct pulse emission without a radio frequency (RF) carrier modulation [16–20]. UWB pulse transmission employs basic Gaussian, Hermitian, and discrete prolate spheroidal pulses for wave propagation inside the body. The various pulse shapes which are used for IR UWB systems like Gaussian and Hermite monocycle, and discrete prolate spheroidal pulses does not have optimum spectrum utilization, therefore, modified using higher-order derivatives of the pulses and suitably bandpass filtered to eliminate undesired frequency components to satisfy the FCC requirements [21–24]. Also, using scaling techniques, the spectrum of standard UWB pulses can conform to FCC mask specifications. Gaussian and Hermite waveforms can be matched to the FCC mask for UWB radios. In [13], various UWB pulse generation methods using combination of Gaussian derivative pulses are proposed, but it does not conform to FCC mask and requires a complex implementation to determine optimum weighting coefficients based on the minimum mean-square error algorithm. In [14], modified Hermite polynomial-based pulses are processed using frequency shifting and bandpass filtering to fit the FCC mask. All standard UWB higher-order derivatives UWB pulses will meet FCC mask, but pulse length will increase resulting in lower data transmission rate. But these schemes have less flexibility in exactly fitting power spectral density FCC mask. In [15], the pulses generated satisfy FCC mask specifications, but spectrum efficiency is low and operated at high sampling rate, hence hardware required is more. In multiple access UWB communications, the principal requirement is that the pulses emitted are strongly orthogonal. Prolate spheroidal wave functions are used in the most common FCC-compliant orthogonal pulses [16]. Prolate spheroidal wave functions (PSWFs) were investigated by the authors in [25–30]. Prolate spheroidal pulses has superior features like FCC mask compliance and orthogonality. But very high computational complexity is involved when pulse shapers are used to produce response and were prolate spheroidal pulses can be approximated in the least square sense [8]. Among different ordered pulses, the modified Hermite pulses (MHP) have orthogonal property aiding simultaneous transmission of UWB pulses without collision and hence improved bit error rate. The enhanced data rate techniques would include high-volume data tracking, measurement, and control of wireless devices without overloading the network bandwidth and hence new applications of high-speed wireless sensors [5].

2 UWB Pulse Shaping Techniques

The UWB pulse shaping filter to shape the transmission pulse for a UWB impulse radio system is designed to satisfy the FCC transmission spectral mask, to optimize the spectrum bandwidth efficiently, to suppress the ringing effect, to improve the BER performance and to lower the sidelobe level, and thereby improves the localization properties necessary for employing UWB pulse effectively for medical imaging applications. To comply with the FCC indoor emission, generally fourth, seventh or higher derivative of Gaussian, Hermitian, and DPSS pulses are used as an input pulse. This pulse is spectrally altered by windowing with Gaussian and other windows to achieve the desired quality. Baseband pulse shaping filters are employed to sustain a signal at an allocated bandwidth, optimize the rate of transmission of data and reduce errors in transmission. They also produce high stop band attenuation, to eliminate inter-channel interference and reduced inter-symbol interference to achieve as low bit error rate and to achieve the maximum power transmitted within FCC mask. The transmitting signal is processed employing a finite impulse response pre filter before being modulated, and the impulse response coefficients are designed to yield the required-shaped pulse. Digital FIR filters, multistage half-band filter [27] raised cosine filter, and optimal designs are employed to obtain the FIR filter coefficients. In [28], pulse shaping using frequency domain characteristics is employed to utilize frequency characteristics for decomposition and recombination of the signal received. In [29], a pulse shaper design is proposed to decrease the interference and reduce the ringing oscillation. The modified pulse can track the transmitted power spectral density and provides enhanced resolution of antenna power. To decide the antenna size, the signal bandwidth is a vital factor [30, 31]. The bandwidth of a transmitted signal is varied with a difference in the standard deviation of the Gaussian pulse-shaped window. Modified pulse generators as per the bandwidth requirement are useful for various UWB transmitters, which provide flexibility of antenna arrangements for various applications. The analysis shows that with the reduction in standard deviation values for the Gaussian window, the bandwidth of the UWB pulse increases. The analysis of pulse shape factor (PSF), accuracy estimation, and signal-to-noise ratio (SNR) is carried out in [29] to examine the proposed pulse shaping for position accuracy.

Employing the proposed pulse shaping filter, resulting UWB pulse has superior features like higher spectral utilization efficiency (SUE), negligible ringing effect, orthogonality, improved BER performance, and lower sidelobe level (SLL) that aids in clutter reduction and target image improvement for medical imaging applications is synthesized.

3 UWB Pulse Shaping Filter

Pulse transmission in human body commonly employs Gaussian pulse with suitable center frequency and bandwidth based on particular application.

The n th derivative of commonly employed Gaussian pulse is [29]

$$g^n(t) = \frac{-n-1}{\sigma^2} g^{n-2}(t) - \frac{t}{\sigma^2} g^{n-1}(t) \tag{1}$$

σ is the standard deviation of n th derivative of Gaussian pulse. In this paper, the seventh derivative UWB Gaussian pulse is shaped employing a proposed FIR bandpass filter. The pulse shaper filter output is the shaped UWB pulse to be transmitted in the body given by

$$p(nT) = \sum_{n=0}^{L-1} h(n)g(t - nT) \tag{2}$$

where $h(n)$ is the L filter impulse coefficients to shape the basic pulse, $g(t)$ is the basic seventh derivative Gaussian UWB pulse, and T is the sampling interval. Thus, superior UWB pulses are numerically generated for transmission in the body.

In the proposed filter design, the desired frequency response of the proposed bandpass FIR filter is modeled using trigonometric functions in the pass band and stop band to satisfy the FCC spectral mask specifications and required center frequency and bandwidth for a particular application. Thus, a continuous function of the desired frequency response $H_d(f)$ of the filter is obtained [32, 33]. Employing the frequency sampling FIR filter design technique, FIR filter coefficients $h(n)$ are obtained for a filter length to meet the desired specifications. This method minimizes the effects of Gibb’s phenomena and allowing us to eliminate ripples at the filter transition region, and hence a lower dominant sidelobe is obtained for a lower filter length to synthesize the UWB transmission pulse.

The impulse response coefficients of FIR pulse shaping bandpass filter are obtained for the desired specifications as

$$\begin{aligned} h(n) = & \frac{A\delta_p \cos k\omega_b}{\pi(k_p^2 - k^2)} [k_p \sin k_p\omega_c \cos k\omega_c - k \cos k_p\omega_c \sin k\omega_c] \\ & + \frac{A\delta_s k_p}{2\pi(k_p^2 - k^2)} [\cos k_p(\omega_b - \omega_z) - \cos k(\omega_b - \omega_z)] \\ & + \frac{A}{k^2\pi(\omega_z - \omega_c)} [\cos k(\omega_b - \omega_c) - \cos k(\omega_b - \omega_z)] \\ & + k(\omega_z - \omega_c) \sin k(\omega_b - \omega_c)] \\ & + \frac{A}{k^2\pi(\omega_z - \omega_c)} [\cos k(\omega_b + \omega_c) - \cos k(\omega_b + \omega_z)] \end{aligned}$$

$$\begin{aligned}
& -k(\omega_z - \omega_c) \sin k(\omega_b + \omega_c)] \\
& + \frac{A\delta_s}{2\pi(k_p^2 - k^2)} [[k_p \cos k_p(\pi - (\omega_b + \omega_z)) \cos k\pi - k_p \cos k(\omega_b + \omega_z)] \\
& + k \sin k_p(\pi - (\omega_b + \omega_z)) \sin k\pi + \frac{2A \cos k\omega_b \sin k\omega_c}{k\pi}
\end{aligned} \tag{3}$$

where k_p is the parameter that shapes the response in the pass band region based on the intended specifications, ω_c is the center frequency, ω_b is the 10 dB bandwidth, δ_p is 10 dB, δ_s is the dominant sidelobe level, ω_z is the frequency at which the magnitude of the filter response is zero, and A is the filter gain parameter.

$k = \frac{L-1}{2} - n, n = 0, 1, \dots, \frac{L-1}{2}$ for L odd and $n = 0, 1, \dots, \frac{L}{2} - 1$, for L even. Employing the shaping filter coefficients $h(n)$, the frequency response $H(f)$ of the shaping filter is obtained. Also employing suitable transformation techniques, our proposed time domain filter window function $h(t)$ is obtained. Further, to obtain the shaped Gaussian pulse, we consider the UWB seventh order Gaussian pulse

$$g(t) = K \left(\frac{105t}{\sigma_1^8} - \frac{105t^3}{\sigma_1^{10}} + \frac{21t^5}{\sigma_1^{12}} - \frac{t^7}{\sigma_1^{14}} \right) \exp\left(-\frac{t^2}{2\sigma_1^2}\right) \tag{4}$$

where the standard deviation or PSF of Gaussian pulse is

$$\sigma_1 = \frac{1}{2\pi f_b \sqrt{\log_{10}(e)}} \tag{5}$$

The Gaussian seventh order time domain pulse $g(t)$ is multiplied with our proposed time domain filter window function $h(t)$ to obtain the shaped Gaussian time domain pulse $p(t)$ to be transmitted in the body mathematically given by

$$p(t) = g(t)h(t) \tag{6}$$

4 Results and Discussions

To comply with FCC limitations, the seventh derivative UWB Gaussian pulse is shaped using the proposed FIR filter with a standard deviation of 35 ps and a center frequency of $f_c = 4$ GHz. Figure 1 shows the PSD plot of the basic UWB pulse and shaped pulse. It is found that the shaped pulse fits the FCC mask specifications with good spectral utilization efficiency of 84%, which is better than widely used Gaussian pulse and hence provide better antenna resolution and low ringing effect in time domain. The effective isotropic radiated power (EIRP) of the shaped pulse is -41.3 dBm/MHz, and hence the synthesized pulse has superior features like robustness against jamming, does not cause significant interference to other systems operating

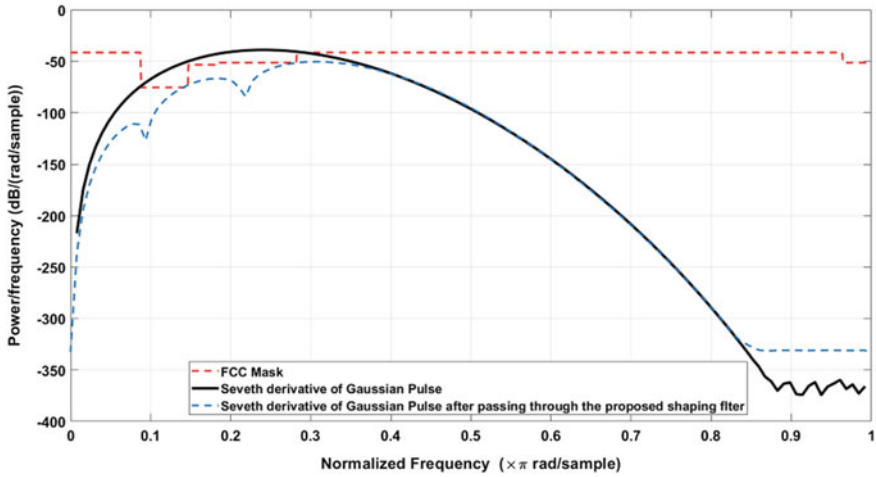


Fig. 1 FCC spectral mask and PSD of seventh derivative Gaussian pulse and the shaped pulse with $f_c = 4$ GHz employing the proposed pulse shaper

in the vicinity and will not cause a threat to patient’s safety in medical imaging applications.

Figure 2 displays the frequency characteristics of synthesized UWB pulses using the suggested pulse shaping filter with $f_c = 4$ GHz in the medical band of 3.4–4.8 GHz. It can be noted that the exact center frequency is obtained with low sidelobe level. For the medical imaging applications like tumor detection, due to the important contrast in the electromagnetic properties of malignant and normal human tissue, the presence of significant SLL will create strong clutter. Therefore, frequency response characteristics of the synthesized pulse with drooping sidelobe characteristics will result in tightly focused image which is required for UWB imaging applications. The time domain basic pulse and shaped pulse of duration 0.5 ns is shown in Fig. 3. It is observed that the ringing effect of Gaussian pulse is reduced by using the proposed bandpass FIR shaping filter. This allows impulse period T_p of the pulse to be reduced and hence minimizes adjacent interference for pulse transmission, thus supporting

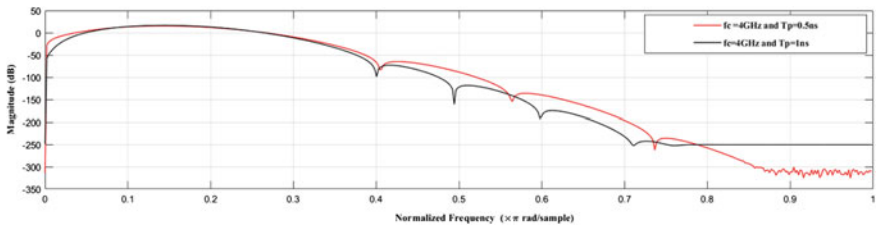


Fig. 2 Frequency response of synthesized UWB pulse employing proposed pulse shaping filter with $f_c = 4$ GHz in the medical band of 3.4–4.8 GHz for $T_p = 1$ ns and $T_p = 0.5$ ns

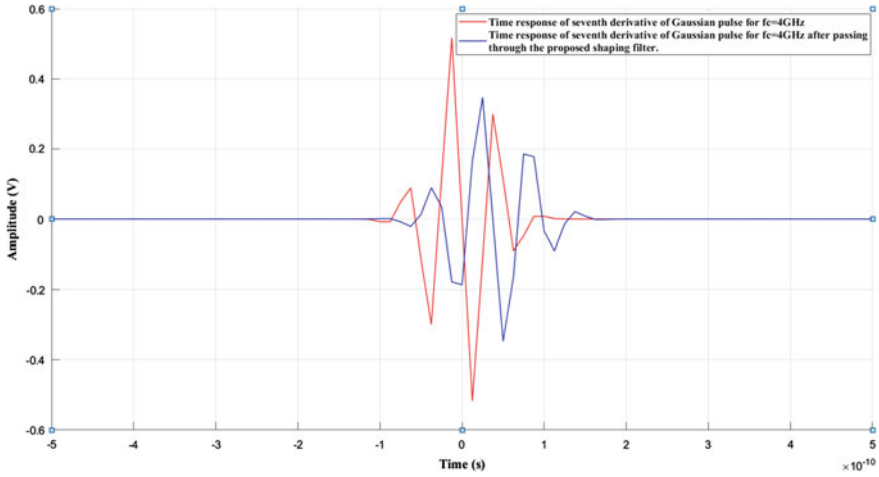


Fig. 3 UWB time domain seventh derivative Gaussian pulse with center frequency $f_c = 4$ GHz and its shaped response employing proposed pulse shaping filter of duration 0.5 ns

higher data transmission rate in UWB pulse communication. Better antenna power distribution and improved location accuracy are achieved due to low ringing effect observed in the shaped pulse for precise location of tumor for medical imaging. Figure 4 shows the autocorrelation of shaped pulses $\Psi_1(t)$ and cross-correlation of shaped pulses $\Psi_1(t)$ and $\Psi_2(t)$. It is observed that at the sampling instant, the autocorrelation of $\Psi_1(t)$ is more than a cross-correlation between $\Psi_1(t)$ and $\Psi_2(t)$, where $\Psi_1(t)$ and $\Psi_2(t)$ are the eigenvectors corresponding to highest eigenvalues of the

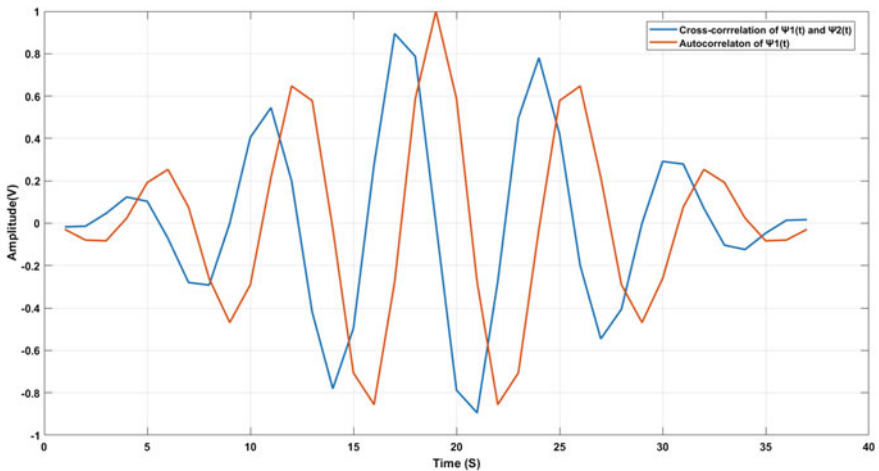


Fig. 4 Correlation functions of pulse shapes obtained from the proposed pulse shaper. **a** Cross-correlation of $\Psi_1(t)$ and $\Psi_2(t)$ and **b** autocorrelation of $\Psi_1(t)$

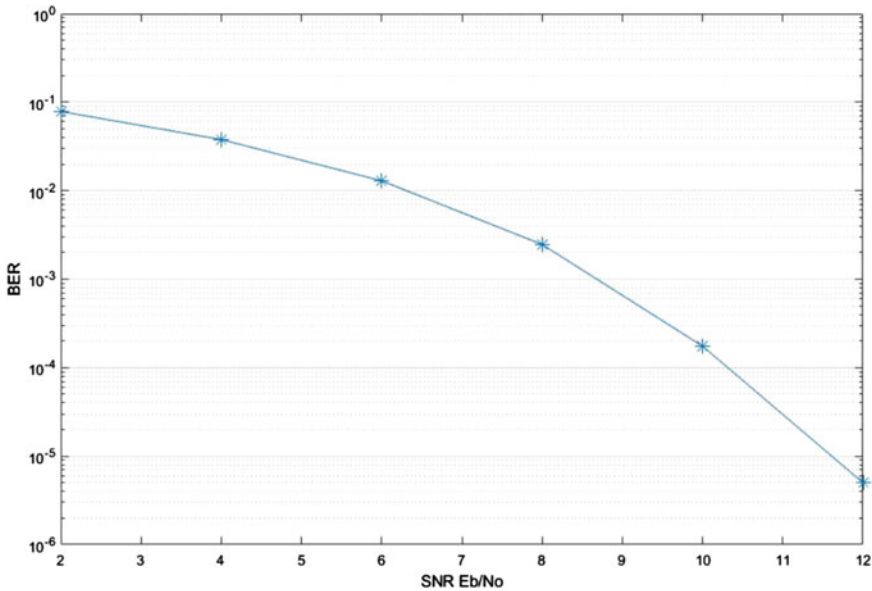


Fig. 5 BER of shaped UWB pulse using the proposed pulse shaper

Toeplitz matrix formed by the filter coefficients of the proposed band pass FIR filter. Having large autocorrelation is advantageous in determining multipath energy where as having zero cross-correlation at sampling instant results in the removal of multiuser interference assuming perfect timing synchronization. The strong eigenvector pulses generated using the proposed FIR filter has the pulse duration less than that generated using Hermite polynomials of order three or higher discussed in [12] resulting in higher data rates which supports medical imaging applications. Figure 5 shows BER performance of the shaped UWB pulse which shows reduced BER and hence better performance can be achieved in multiuser environment with additive white Gaussian noise (AWGN). The synthesized pulse has BER value less than 10⁻⁵ which is better than averaged BER performance using different Gaussian templates discussed in [34]. It ensures lesser distortion and more signal energy collection of transmitted pulse while propagating through body tissue. A low BER of the synthesized pulse makes it perfect for location tracking applications like tumor detection.

5 Conclusion

A bandpass FIR filter is designed to shape widely used seventh derivative Gaussian pulse to be used for medical imaging applications. The synthesized pulse shaped

using the proposed band pass filter has superior features, i.e., low BER and negligible ringing effect for better antenna resolution, low SLL for interference avoidance, the low impulse time period for high data transmission rate, and good SUE and orthogonality. These features help to achieve a tightly focused image with less clutter, high precision for location accuracy of time of arrival (ToA) measurement that makes it a perfect choice for medical imaging applications like development of UWB-based cancer detection systems with good detection and localization properties. The shaped UWB pulse spectra fit the desired FCC frequency mask. Any required center frequency and bandwidth can be accommodated by the design. To boost data rate, the impulse time period T_p is reduced, but the dominating sidelobe level is increased, resulting in adjacent channel interference. T_p is chosen to obtain the desired frequency specification of higher spectral utilization efficiency and lower sidelobe while minimizing the duration of the UWB pulse width. Employing this technique any desired response like single band and multiband frequency response can be achieved for interference avoidance for the given center frequencies easily.

References

1. Bo Hu N, Beaulieu C (2005) Pulse shapes for ultrawideband communication systems. *IEEE Trans Wireless Commun* 4:1789–1797
2. Federal Communications Commission.: Revision of part 15 of commission's rules regarding Ultra-Wideband Transmission systems, ET Docket, pp 98–153 (2002)
3. Rahayu Y, Rahman TA, Ngah R, Hallv PS (2008) Ultra wideband technology and its applications. In: 5th IFIP International conference on wireless and optical communications networks (WOCN '08) on Proceedings, pp 1–5. Indonesia
4. Yong X, Yinghua L, Hongxin Z, Yeqiu W (2007) An overview of ultra-wideband technique application for medial engineering. In: *IEEE/ICME international conference on complex medical engineering on proceedings*, pp 408–411. Beijing
5. Pradhan KP, Li Y-G, Arifuzzman AKM, Haider MR (2017) Modified hermite pulse-based wideband communication for high-speed data transfer in wireless sensor applications. *J Power Electron Appl* 7(4):30
6. Sablatash M (2006) Pulse shaping, modulation and spectrum shaping for UWB wireless communications and the effects on interference for single and multiband transmission of UWB signals. In: *Canadian conference on electrical and computer engineering on proceedings*, pp 1640–1645, Ottawa
7. Immoreev I, Fedotov PGSDV (2002) Ultra-wideband radar systems: advantages and disadvantages. In: *IEEE Conference on ultra-wideband systems and technologies on proceedings*, pp 201–205. Baltimore, MD, USA
8. Werfelli H, Chaoui M, Ghariani H, Lahiani M (2013) Design of a pulse generator for UWB communications. In: *10th International multi-conferences on systems, signals & devices on proceedings*, pp 1–6. Tunisia
9. Wang Y, Zhou L (2008) Performance evaluation in the presence of timing jitter using a novel pulse design based on Hermite functions for UWB communications. In: *Asia-Pacific microwave conference on proceedings*, pp 1–4. Macau
10. Ketata M, Dhieb M, Ben Hmida G, Ghariani H, Lahiani M (2015) UWB pulse propagation in human tissue: comparison between Gaussian and square waves shape. In: *16th International conference on sciences and techniques of automatic control and computer engineering on proceedings*, pp 158–162. Tunisia

11. Hu B, Beaulieu NC (2003) Exact bit error rate analysis of TH-PPM UWB systems in the presence of multiple access interference. *IEEE Commun Lett* 7(12):572–574
12. Michael LB, Ghavami M, Kohno R (2002) Multiple pulse generator for ultra-wideband communication using Hermite polynomial based orthogonal pulses. In: *IEEE conference on ultra wideband systems and technologies on proceedings*, pp 47–51. Baltimore
13. Parr B, Cho B, Wallace K, Ding Z (2003) A novel ultra-wideband pulse design algorithm. *IEEE Commun Lett* 7(5):219–221
14. Win M (1999) Spectrum density of random time-hopping spread-spectrum UWB signals with uniform timing jitter. In: *Military communications conference (MILCOM'99) on proceedings*, pp 1196–1200. Atlantic City, NJ
15. Welborn ML (2001) System consideration for ultra-wideband wireless networks. In: *IEEE radio and wireless conference on proceedings*, pp 5–8. Boston, MA
16. Qiu RC, Liu H, Shen X (2005) Ultra-wideband for multiple access communications. *IEEE Commun Maga* 43(2):80–87
17. Zhang J, Abhayapala TD, Kennedy RA (2003) Performance of ultrawideband correlator receiver using Gaussian monocycles. In: *IEEE international conference on communications (ICC 2003) on proceedings*, pp 2192–2196. Anchorage, AK
18. Kendall SM, Stuart A (1977) *The advanced theory of statistics distribution theory*, vol 1. London
19. Hu B, Beaulieu NC (2004) Accurate evaluation of multiple access performance in TH-PPM and TH-BPSK UWB systems. *IEEE Trans Commun* 52(10):1758–1766
20. Scholtz RA (1993) Multiple access with time hopping impulse modulation. In: *Military communications conference (MILCOM'93) on proceedings*, pp 447–450. Boston, MA
21. de Ávila LA, Kunst R, Pignaton E, Rochol J, Bampi S (2016) Energy efficiency evaluation of the pulse shapes and modulation techniques for IR-UWB in WBANs. In: *IEEE international conference on electronics, circuits and systems (ICECS) on proceedings*, pp 173–176. Monte Carlo
22. Ghavami M, Amini A, Marvasti F (2008) Unified structure of basic UWB waveforms. *IEEE Trans Circuits Syst II: Express Briefs* 55(12):1304–1308
23. Joe J (2003) Cellonics UWB pulse generators. In: *International workshop on ultra-wideband systems*, Oulu, Finland
24. Foerster JR (2002, May) The performance of a direct-sequence spread spectrum ultra-wideband system in the presence of multipath, narrowband interference and multiuser interference. In: *IEEE conference on UWB systems and technologies on proceedings*, pp 87–91
25. Win MZ, Scholtz R (2000) Ultra-wide bandwidth time-hopping spread spectrum impulse radio for wireless multiple-access communications. *IEEE Trans Commun* 48(4):676–91
26. Guvenc I, Arslan H (2003) On the modulation options for UWB systems. In: *Military communications conference MILCOM '03 on proceedings*, pp 892–97
27. Cramer RJ, Scholtz RA, Win MZ (2002) Evaluation of an ultra-wide-band propagation channel. *IEEE Trans Antennas Propag* 50(5):561–570
28. Luo X, Yang L, Giannakis GB (2003) Designing optimal pulse-shapers for ultra-wideband radios. In: *IEEE conference on ultra-wideband systems and technologies on proceedings*, 2003, pp 349–353. Reston, VA, USA
29. Sharma A, Sharma SK (2019) Spectral efficient pulse shape design for UWB communication with reduced ringing effect and performance evaluation for IEEE 802.15.4a channel. *Wirel Networks* 25(5):2723–2740
30. Harrington RF (1960) Effect of antenna size on gain, bandwidth, and efficiency. *J Natl Bureau Standards* 64D(1):1–12
31. Devika Menon MK, Rodrigues J, Gudino LJ (2020) Synthesis of UWB pulse shaper for efficient pulse propagation in human tissue. In: *12th international symposium on communication systems, networks and digital signal processing (CSNDSP) on proceedings*, pp 1–5, Portugal
32. Rodrigues J, Pai KR (2005) New approach to the synthesis of sharp transition FIR digital filter. In: *IEEE international symposium on industrial electronics on proceedings*, pp 1171–1173. Coratia

33. Gudino LJ, Rodrigues JX (2008) Linear phase FIR filter for narrow-band filtering. In: International conference on communications, circuits and systems on proceedings, pp 776–779. Fujian
34. Chávez Santiago R, Balasingham I, Bergsland J (2012) Ultrawideband technology in medicine: a survey. *J Electr Comput Eng* (9):1–9

Quantitative Analysis of Transfer Learning in Plant Disease Classification



Pawan Dubey, Vineeta Kumari, Ajay K. Sharma, and Gyanendra Sheoran

Abstract The non-invasive schemes offer an opportunity for farmers in early plant disease symptom detection and remedial step identification. Recently, deep learning (DL) techniques provide a non-invasive way for early plant disease classification because of ease in high computational hardware availability and estimated disease detection. Therefore, many of the research works have investigated transfer learning of various popular DL models for plant disease detection. Though several studies in the literature considered leaf images of multiple plant species for model training, model may display improper classification performance because of such training process. The proposed work intends to provide quantitative study of performance analysis of popular DL models which takes images database of plant species, such as tomato, grapes, corn, and potato, into its consideration. The proposed work evaluates the transfer learning performance of following DL models, i.e., AlexNet, Inception V3, ResNet-50, VGG 16, and MobileNet, on image databases available in open-source Plant Village database. This evaluation reveals that Inception v3 model is a primary efficient algorithm which managed accurate classification for the most of diseases of considered plant species except grapes plant, for which MobileNet model displays outperforming results.

Keywords Plant disease classification · Transfer learning · Deep learning

1 Introduction

Many of the countries across the world are dependent on agricultural produce for economy and food. Plants are the vital entity for producing food, medicine, and other needs of the living species. As like humans, these plants are also suffered with diseases

P. Dubey · V. Kumari · A. K. Sharma · G. Sheoran (✉)
National Institute of Technology Delhi, New Delhi, Delhi 110040, India
e-mail: gsheoran@nitdelhi.ac.in

P. Dubey
e-mail: pawand@mitsgwalior.in

A. K. Sharma
e-mail: director@nitdelhi.ac.in

that may cause death to these plants and result in substantial economic, social, and environmental losses for any country. Thus, plants should be properly monitored for early disease symptom identification. Such monitoring had been performed in ancient times as well. But, it was performed through tedious and time-taking process like manual crop inspection by persons having some training and expertise about plant disorders. However, most of the time, it was very hard to find such trained experts in small, poor, and isolated areas. Also, human inspection may be biased and optical illusions may result an error while inspection [22]. Since the characteristics of plant diseases are observed in terms of deformation in plant elements, such as root, leaf, blossom, flowers, and stems [8], the visual inspection through computer vision-based techniques, i.e., non-invasive, could, provide a better solution and eradicate unsuitability and unreliability of human inspection in early disease detection.

The computer-based visual inspection of plant diseases can be subdivided into two subcategories: the conventional and the machine learning- or deep learning-based approaches. The traditional approaches used to obtain disease attributes (features) from the plant leaf image using specific image processing algorithms such as color, shape, and texture or the combination of both [2, 4, 6, 14]. These attributes were either employed for direct classification or subjected with classifiers. However, these approaches were required appropriate imaging system in terms of light source and image capturing angle which may increase the system cost. Additionally, these approaches demonstrate low performance with tedious and difficult real-time feature collection at the cost of plant life. Moreover, these are often suffered with the impact of scene changes on recognition performance [13]. On contrary, DL-based approaches uses convolution neural network (CNN) which have self-learning capability and have proved their utility in various fields of computer vision (CV), e.g., traffic detection, medical image recognition, scenario text detection, expression recognition and face recognition, etc. [18]. Since the employability of CNN-based methods is not only limited to above-said domains, they are also being employed for precision agriculture and related industries. In view of re-utilization of deep learning (DL) models from one field of research to another field of research, many of the researchers have exploited transfer learning for agriculture and plant health monitoring and classification.

The disease recognition in plant leaves is important topic of research [5] in recent time and deep learning exploits models composed of various layers. These layers process the signal and extract vital characteristics of leaf image data for model learning [16]. The involved CNN layers in DL model extract vital information for image classification which helps in diagnosing the plant diseases for precision farming. Mohanty et al. employed end-to-end deep transfer learning of CNN models on a very large database and shown the proficiency of various approaches over hand-crafted feature-based representations [21]. However, considered images were captured in constrained environment. Therefore, drop in performance was observed while training and test image sets were of different capturing environments.

Sladojevic et al. [25] designed and applied a new CNN model on new diseased plant image database which had more than 3000 original images taken from internet sources. The database was further extended to more than 30,000 through data

augmentation to attain a high accuracy. Amara et al. employed DL approaches for disease detection in banana leaf [3]. LeNet was the model that employed on dataset derived from the Plant Village database [17]. However, the method was not appropriate for the real-time scene capturing conditions. Authors revealed that the real scene capturing disease localization is an important step in the plant monitoring system. Ferentinos applied deep learning models, such as AlexNet, AlexNet, OWTBn, GoogLeNet, OverFeat, and VGG, on laboratory and real time field, captured plant leave database [9]. The authors investigated performance of the CNN approaches on un-segmented images. Liu et al. applied traditional machine learning as well as CNN approaches on synthetic apple images generated from 1153 samples [19]. The study aimed to simulate real-time conditions and devise new computationally efficient CNN approaches. Moreover, authors pointed out the need of single shot detector algorithms and database expansion.

Among several exhaustive surveys based on CNN for plant health monitoring [1, 5, 20, 23], most of the schemes were trained on database which was formed by considering the images of various plant species. Since each plant species may be suffered with different kinds of diseases and sometimes it is observed that some of the plants may have multiple number of diseases as compared to other plants, this may limit one to prefer appropriate model for a particular plant disease. Therefore, the proposed work intends to provide quantitative analysis of the powerful *DL* approaches to opt a better transfer learning model for particular plant disease identification. Therefore, to provide optimum model selection for specific plant diseases, the performance of the five popular DL methods, i.e., AlexNet [15], Inception v3 [26], Visual Geometry Group (VGG) [24], ResNet [10], and MobileNet [11], has analyzed quantitatively. The remainder of the paper is organized in the following manner: Section 2 presents the employed materials and methods in study; Section 3 provides obtained results and discussion; and finally, Section 4 will conclude the study along with future directions.

2 Materials and Methods

2.1 Database

This work utilizes the Plant Village dataset [12] for quantitative analysis of transfer learning of the considered models. The database contains 54,309 plant leaf images where plant images from well-known 14 plant species were captured. Also, in this dataset, plant leaves were suffered with 17 fungal, 4 bacterial, 2 mold (oomycete), 2 viral and 1 mite diseases. The healthy images which have no visible defects from 12 crops were also present. For transfer learning of our considered models, 2058, 4066, 12,994, and 3852 number of image samples of healthy and sick leaves from potato, grapes, tomato, and maize are employed, respectively.

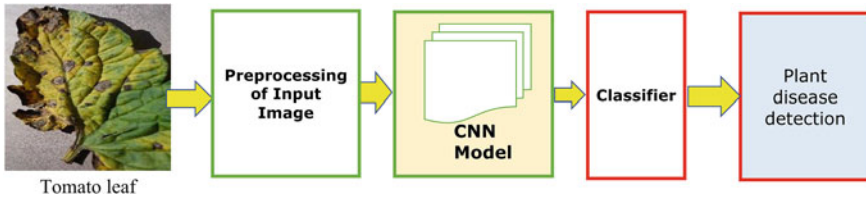


Fig. 1 General steps for deep learning classification

2.2 Methods

Recently, due to significant performance of DL methods, these methods are being used in various fields of research such as recognizing the speech, visual objects, and detecting the objects. Though earlier the investigations of DL approaches were very difficult to implement due to scarcity in data creation, storage, and processing, the generation of fast computers with large memory mobilized the use of these approaches for different research areas [18].

Summarized steps of Transfer Learning through deep learning models are shown in Fig. 1 and discussed briefly below:

- Step 1: Input plant image.
- Step 2: Resize the image samples as per the DL model requirement using interpolation methods.
- Step 3: Train the pre-trained deep (CNN) models with specific number of convolutional layers, fully connected layers, softmax layers, and classification output layers, respectively.
- Obtain classification performance for plant diseases.

In this work, performance of five different powerful deep neural network architectures has been investigated to resolve the disease classification task in various plant diseases. These architectures are trained with available different plant images in Plant Village database. The AlexNet is a basic model composed of 25 layers where the weights are trained with eight different layers, only [15]. The GoogLeNet exploits network in network approach and employs multiple convolution at a time to find fine to coarse feature points [26]. The VGG employs homogeneous architecture for better plant disease representation [24]. The VGG composed of 16/19 layers deep as compared to AlexNet and obtains the depth relation along with representational caliber of the network. The ResNet developed by He et al. trains the model in more deeper way using residual ensembles [10]. The ResNet performs much better recognition and localization tasks for images and displays. The MobileNets employs convolutions which are depth-wise separable to construct a lightweight deep neural networks. It has claimed that MobileNet is a fast architecture that provides accuracy equivalent to VGG 16 with less number of computations [11]. The proposed work utilizes the weights of the each model architecture obtained through training the models using

the above-mentioned database images. Thereafter, the classification is performed on test image databases using these training weights.

3 Results and Discussion

3.1 *Experimental Setup and Hyper-parameters*

In the experimental setup, tomato, grapes, corn, and potato subset databases, contains 12,994, 4066, 3852, and 2058 number of image samples, respectively. Exemplary image samples of diseased leaves within datasets are shown in Fig. 2. The 70% image samples of each species database is employed for model training, whereas remaining 30% of image samples are kept for validation and testing purpose. Thus, validation and test datasets are derived from dataset which is not a part of training data and share half number of images of remaining database. The performance for each architecture is optimized using Adam optimization algorithm which is an extension of the stochastic gradient decent method. The learning rate parameter for the Adam optimizer is tuned to 0.001, whereas β is set to 0.7. Each architecture is run up to 100 epochs for learning rate convergence. The Softmax classifier is used in order to get final classification output. All the experiments are performed with GPU hardware supported by “Google Colaboratory” [7].

3.2 *Evaluation on Tomato Plant*

There are seven type of diseases, i.e., bacterial spot, septoria, spider-mites, target spot, yellow leaf curl, early blight, leaf mold, samples were collected in Plant Village tomato database. The considered models are trained with 70% images of total images present in different classes of disease, and the remaining were kept for validation and testing. The experimental outcomes in terms of classification accuracy are presented in Table 1. The experimental outcomes reveal that MobileNet [11] is the best performing algorithm for bacterial spot disease classification with accuracy of 100%. For classifying the Septoria disease, the Inception v3 outperforms all approaches by showing 100 percent classification accuracy. The disease, spider-mites, is classified efficiently not only by Inception v3 [26] but also with VGG 16 [24] and MobileNet [11] with accuracy of 99.17%. Target spot disease is classified best by the Inception v3[26] with accuracy of 99.06%. However, there is a marginal improvement of 0.47%. The VGG 16 [24] proved to be the best in classifying the yellow curl virus disease with 100% accuracy. Again, Inception v3 [26] outperforms all the approaches for early blight disease detection displays the accuracy of 99.37%, whereas for the leaf mold disease identification, VGG 16 [24] proves to be the best approach among all approaches. If we compare the overall performance of approaches, the Inception

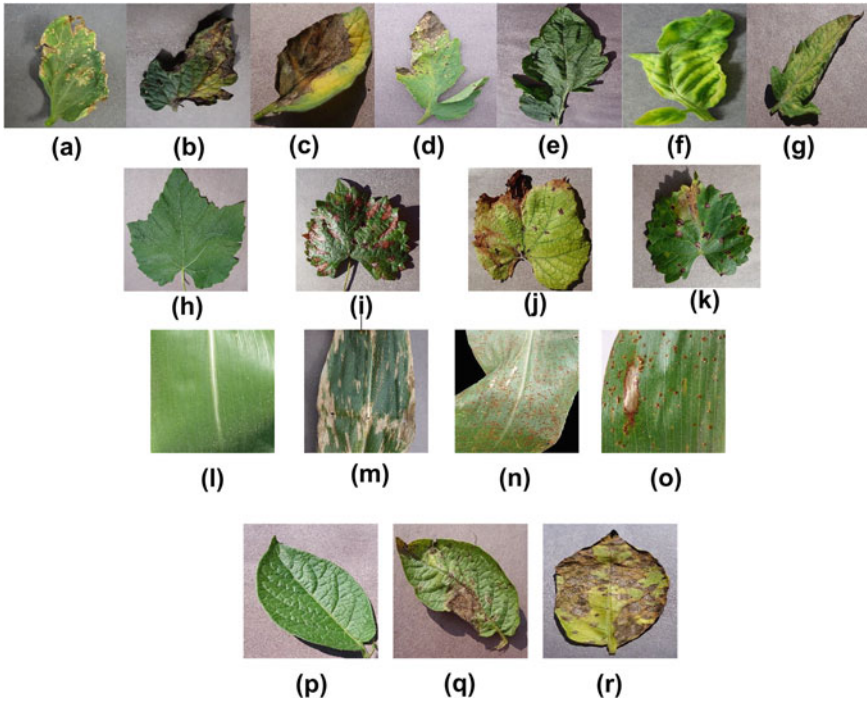


Fig. 2 Disease affected images for different plants: tomato diseases (first row): **a** bacterial spot, **b** early blight, **c** late blight, **d** septorial leaf, **e** target spot, **f** yellow curl virus and **g** leaf mold, grapes plant diseases (second row): **h** healthy, **i** black measles, **j** black rot, and **k** leaf blight; corn plant diseases (third row): **l** healthy, **m** cercospora, **n** common rust and **o** northern leaf blight; potato plant diseases (fourth row): **p** healthy, **q** late blight, **r** early blight

v3 [26] can classify most of tomato leaf diseases, except bacterial spot, yellow curl virus, and leaf mold. The reason for such a good classification is present in model architecture of Inception v3 [26] where traditional convolutional layers are modified to a small blocks that possess different size filters to capture more versatile features.

3.3 Evaluation on Grapes Plant

The grape leaves dataset contains healthy leaves and diseased leaf both with name late blight, black rot, and black measles. Table 2 displays the experimental outcomes of considered algorithms with this dataset. It can be observed that all approaches except AlexNet are able to identify healthy leaf with the accuracy of 100%. The late blight disease can simultaneously be classified by Inception v3 [26], VGG 16 [24] and MobileNet [11] with accuracy of 100%. Thus, one can choose MobileNet [11] as best algorithm because of its less computations. For black rot, VGG 16

Table 1 Comparative disease test accuracies on tomato image dataset

Diseases/ Models	ResNet 50 [10]	Inception v3 [26]	Alexnet [15]	VGG 16 [24]	MobileNet[11]
Bacterial spot	99.73	99.73	99.47	99.47	100
Septoria	98.56	100	93.18	98.20	99.28
Spider-mites	97.93	99.17	94.62	99.17	99.17
Target spot	93.47	99.06	93.42	98.13	98.59
Yellow leaf curl	99.06	99.68	96.26	100	99.68
Early blight	91.19	99.37	83.01	96.26	96.85
Leaf mold	97.76	98.50	92.53	99.25	97.76

Table 2 Comparative disease test classification accuracies on grapes database

Diseases/ Models	ResNet 50 [10]	Inception v3 [26]	Alexnet [15]	VGG 16 [24]	Mobile net [11]
Healthy	100	100	98.70	100	100
Late blight	99.41	100	96.49	100	100
Black rot	98.79	99.69	97.87	100	100
Black measles	99.52	100	96.17	9.76	99.52

[24] and MobileNet [11] emerged out to be the best algorithms. For black measles disease detection, the Inception v3 is proved to be the best algorithm among all. For the overall disease classification performance, MobileNet can be a better option in grapes' disease classification. The MobileNet [11] outperforms all approaches for the most of grapes plant diseases except black measles. However, by showing 99.52% classification accuracy, it emerged out to be the next best approach for this disease as well.

3.4 Evaluation on Corn Plant

In Plant village database, the corn plant images are affected with three diseases, such as cercospora, northern leaf blight, and common rust. The experimental outcomes of considered models are demonstrated in Table 3. It is apparent from the table that all the model architectures are able to detect healthy leaves with cent percent accuracy. However, the disease cercospora is efficiently classified with MobileNet [11], for which the obtained accuracy is 90.46%. For this disease, the MobileNet [11] outperforms the AlexNet, VGG 16, Inception v3 and ResNet-50 models with the margin of 7.4, 7.3, and 2.4%, respectively. The northern leaf blight is classified well by Inception v3 [26] with accuracy of 99.41%. The common rust disease is classified

Table 3 Comparative disease classification accuracies on corn leaves

Diseases/ Models	ResNet 50 [10]	Inception v3 [26]	Alexnet [15]	VGG 16 [24]	Mobile net [11]
Healthy	100	100	100	100	100
Cercospora	88.09	88.09	83.69	83.80	90.46
Northern leaf blight	96.64	99.41	95.97	94.63	96.55
Common rust	100	99.76	100	100	100

Table 4 Comparative disease classification accuracies on potato leaves

Diseases/ Models	ResNet 50 [10]	Inception v3 [26]	Alexnet [15]	VGG 16 [24]	Mobile net [11]
Healthy	91.30	100	95.65	100	91.30
Early blight	99.32	100	97.30	98.64	98.64
Late blight	99.32	100	97.30	98.64	100

accurately by all the models, except Inception v3 [26]. Finally, it can be observed that MobileNet [11] and Inception v3 [26] better classify most of the corn leaf diseases.

3.5 Evaluation on Potato Plant

The Plant Village dataset contains both healthy and diseased images. The diseased images of the dataset contain early blight and late blight diseases. The experimental outcomes of different models are presented in Table 4. The observations in the table reveal that Inception v3 [26] dominates all models and able to classify both healthy and diseased images with 100% accuracy. Apart from this method, VGG 16 [10] and MobileNet [11] can also be the other options for accurate classification of healthy and late blight affected leaves, respectively. As far as early blight disease classification is concerned, ResNet-50 demonstrates better performance after Inception v3 [26] with the accuracy of 99.62%. Thus, for overall classification of potato plant leaf diseases, the Inception v3 [26] is an outperforming classification model.

4 Conclusion

This paper presented an investigation for selecting an appropriate deep transfer learning model among the most simple and powerful DL models for a particular plant species' disease classification. Although the considered algorithms demonstrated near to accurate disease classification performance for all considered databases, it

was observed that Inception v3 model was found to be the most favorable approach which could classify various diseases for all considered plants such as tomato, grapes, corn, and potato, individually. However, still there is a scope of improvements for this model, so that it can be applicable to classification of various plants simultaneously. Moreover, it was observed that the MobileNet model could be deployed as an alternative model which outperformed all models for grapes diseases and demonstrated second best performance for remaining databases. In the future, this study can also be further carried out on database of diseased and healthy plant images captured in spectral domain.

Funding Acknowledgments This work is funded by the department of science and Technology (DST), India, under the grant number DST/TDT/DDP-09/2018(C)-G.

References

1. Recent advances in sensing plant diseases for precision crop protection. *Eur J Plant Pathol* 133(1):197–209 (2012)
2. Al-Hiary H, Bani-Ahmad S, Reyalat M, Braik M, Alrahamneh Z (2011) Fast and accurate detection and classification of plant diseases. *Int J Comput Appl* 17(1):31–38
3. Amara J, Bouaziz B, Algergawy A (2017) A deep learning-based approach for banana leaf diseases classification. *Datenbanksysteme für Business, Technologie und Web (BTW 2017)-Workshopband*
4. Arivazhagan S, Shebiah RN, Ananthi S, Varthini SV (2013) Detection of unhealthy region of plant leaves and classification of plant leaf diseases using texture features. *Agric Eng Int CIGR J* 15(1):211–217
5. Barbedo JGA (2016) A review on the main challenges in automatic plant disease identification based on visible range images. *Biosyst Eng* 144:52–60
6. Bashir S, Sharma N (2012) Remote area plant disease detection using image processing. *IOSR J Electron Commun Eng* 2(6):31–34
7. Bisong E (2019) Google colab, pp 59–64. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-4470-8_7
8. Bock C, Poole G, Parker P, Gottwald T (2010) Plant disease severity estimated visually, by digital photography and image analysis, and by hyperspectral imaging. *Crit Rev Plant Sci* 29(2):59–107
9. Ferentinos KP (2018) Deep learning models for plant disease detection and diagnosis. *Comput Electron Agric* 145:311–318
10. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 770–778
11. Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, Andreetto M, Adam H (2017) Mobilenets: efficient convolutional neural networks for mobile vision applications. [arXiv:1704.04861](https://arxiv.org/abs/1704.04861)
12. Hughes D, Salathé M et al (2015) An open access repository of images on plant health to enable the development of mobile disease diagnostics. [arXiv:1511.08060](https://arxiv.org/abs/1511.08060)
13. Johannes A, Picon A, Alvarez-Gila A, Echazarra J, Rodriguez-Vaamonde S, Navajas AD, Ortiz-Barredo A (2017) Automatic plant disease diagnosis using mobile capture devices, applied on a wheat use case. *Comput Electron Agric* 138:200–209
14. Khirade SD, Patil A (2015) Plant disease detection using image processing. In: *2015 International conference on computing communication control and automation*, pp 768–771. IEEE

15. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. *Adv Neural Inf Process Syst* 25:1097–1105
16. LeCun Y, Bengio Y, Hinton G (2015) Deep Learn *Nat* 521(7553):436–444
17. LeCun Y et al (2015) Lenet-5, convolutional neural networks 20(5):14. <http://yann.lecun.com/exdb/lenet>
18. Li Z, Liu F, Yang W, Peng S, Zhou J (2021) A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE Trans Neural Netw Learn Syst*
19. Liu B, Zhang Y, He D, Li Y (2018) Identification of apple leaf diseases based on deep convolutional neural networks. *Symmetry* 10(1):11
20. Liu J, Wang X (2021) Plant diseases and pests detection based on deep learning: a review. *Plant Methods* 17(1):1–18
21. Mohanty SP, Hughes DP, Salathé M (2016) Using deep learning for image-based plant disease detection. *Frontiers Plant Sci* 7:1419
22. Patil SB, Bodhe SK (2011) Leaf disease severity measurement using image processing. *Int J Eng Technol* 3(5):297–301
23. Sankaran S, Mishra A, Ehsani R, Davis C (2010) A review of advanced techniques for detecting plant diseases. *Comput Electron Agric* 72(1):1–13
24. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. [arXiv:1409.1556](https://arxiv.org/abs/1409.1556)
25. Sladojevic S, Arsenovic M, Anderla A, Culibrk D, Stefanovic D (2016) Deep neural networks based recognition of plant diseases by leaf image classification. *Comput Intell Neurosci*
26. Szegedy C, Liu W, Jia Y, Sermane P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 1–9

Absolute Moment Block Truncation Coding and Singular Value Decomposition-Based Image Compression Scheme Using Wavelet



Rajiv Ranjan  and Prabhat Kumar 

Abstract The present paper proposes a new hybrid image compression method via the incorporation of the methods, namely Singular Values Decomposition (SVD), Discrete Wavelet Transform (DWT), and Absolute Moment Block Truncation Coding (AMBTC). In order to maneuver these unified methods, DWT compression gives high compression, whereas SVD gives a low compression ratio but gives high image quality. In our new method, the image has firstly compressed with SVD and then further compressed by DWT and AMBTC. The image compressed by SVD is the input image for DWT, and it breaks up into its approximate coefficients and detail coefficients. Further, only approximate image is coded through AMBTC and details images are discarded because of containing less significant data. The newly proposed image compression algorithm has been checked on different natural images, and the results have been compared with other standard methods like BTC, AMBTC, and DWT-AMBTC. The new technique gets a high compression ratio of 20:1 and bitrates to reach 0.4 bits per pixel with good image quality. The objective and subjective results of our method are better than the other compression techniques. The goal of this idea is to attain high compression, and the high compression ratio is received comparatively higher than all other methods with comparable PSNR values for all images.

Keywords Absolute moment block truncation coding (AMBTC) · Discrete wavelet transform (DWT) · Singular values decomposition (SVD) · Image compression

R. Ranjan (✉)
BIT Sindri, Dhanbad, India
e-mail: rajivranjan.it@bitsindri.ac.in

P. Kumar
NIT Patna, Patna, India

1 Introduction

These days, in computer applications, digital images are used broadly. Big storage capacity and high communication bandwidth are needed for those images which are uncompressed. The most appropriate answer to surpass these constraints is image compression [1], which is becoming more important in recent time due to growth of multimedia figures.

The primary aim of data compression is to minimize the volume of necessary data by discarding duplicate data to denote a piece of significant information [2]. For image compression, ‘transform coding’ is the most impressive tactics to reduce redundancy and to provide decorrelated coefficients which are the main objective of this transformation [3]. Wavelet transform (WT) has a famous multi-resolution analysis tool due to its time–frequency features for a long time. The DWT is the discrete version of WT and is too popular for compression of digital images [4] so that the data could be compressed and decompressed in an effective way. The properties of DWT are adopted by several wavelet-based image coding systems to compress the images [5]. On that point, there are many image compression algorithms available nowadays like Transform Coding, Vector Quantization, Block Truncation Coding (BTC), Run Length Coding (RLE), Huffman coding, and Arithmetic Coding, etc. Out of these compression techniques, the paper focuses on AMBTC as it provides quantized reconstructed image in very little time. It possesses the advantages of processing enhanced images when combined with ‘Discrete Wavelet Transform’ (DWT). The proposed methodology of AMBTC with DWT explains taking the existing studies of a few researchers. BTC requires less computational complicity, and it is an easy and rapid lossy compression method. For blocks of image components, the fundamental goal of BTC [6] is to execute moment preserving quantization. With the high quality of the approximate image, this method provides improved compression ratio. Near the edges, it shows only some artifacts or raggedness. BTC has obtained a wide interest in its further expansion, and it is used in image compression due to its little complication and easy execution.

The arrangement of this paper is as taken. Section 2 depicts the published work survey on BTC. The proposed algorithm is discussed in Sect. 3. The proposed technique’s comparison is presented in Sect. 4. Finally, Sect. 5 describes the main conclusion.

2 Related Work

Various changes in BTC have been performed throughout the last many years for enhancing the perfection of the approximate image for the betterment of compression. AMBTC [7] preserves the higher mean and the lower mean of all the blocks. This amount is used for the quantized output. AMBTC is superior to BTC which gives good image quality at the uniform compression ratio. Another feature of AMBTC

is that it is comparatively faster than BTC. Cheng and Tsai [8] advises an adapted image compression method using multilevel BTC. One, two, or four levels are done to appropriately quantize the gray values of each block. This method is better than BTC and AMBTC in computation time, and the tone of the reconstructed image is too good with a higher compression ratio.

To compress the images, Cheng and Tsai [9] have given an idea to keep which depends onto features of the moment preserving edge detection. In an image block, to calculate the parameters of the edge feature, a simple calculation formula is available, so it is faster in computation. The approximate images are proficient in conformity with human visibility. An edge and mean-based method to compress the images are given by Desai et al. [10], which builds competent quality pictures at too little bit rates. On both sides of the edges, the technique depicts the image in forms of its binary edge chart, mean details, and the intensity info. To reach a lower bit rate, Wu and Tai [11] has given a method based on a moment preserving. It is better than BTC in compression ratio with good quality. By using a Fuzzy Complement Edge Operator (YIFCEO) [12], an improved BTC image compression method was invented by Amarunnishad et al. [13] which gives better image quality of an approximate image than BTC. By using the BTC, this technique is depended on the substitution of bit blocks with the fuzzy logical byte block (LBB) so that the sample mean and standard deviation in every image block are sustained. Venkateswaran et al. [14] propose an algorithm which uses AMBTC in the DWT domain and combines the simple numeration and edge-preserving goal of BTC, the high fidelity, and compression ratio of DWT. It gives a higher compression ratio with good image perfection than BTC and AMBTC. For reducing spatial redundancy and the correlation among pixels of an image, Enhanced Block Truncation Coding (EBTC) [15] is a developed compression technique for grayscale images. Compression efficiency and image quality are also maintained by this algorithm. Without degradation of bit rate, Futuristic Algorithm for Gray Scale Image depended on Enhanced Block Truncation Coding (FEBTC) [16] gives more PSNR value than AMBTC and EBTC. An Improved Block Truncation Coding using k-means Quad Clustering (IBTC-KQ) algorithm is invented by Mathews et al. [17]. Bi-clustering in BTC on the place of k-means quad clustering is used by this method.

Identical pixels come under the identic cluster, and it bears on the first-order moment of every cluster in decoding because of this quad clustering. The MSE is less because of the minor distinction between the input image and the approximate image. Thus, it gives a better visual quality of the approximate image than BTC and other modified BTCs. Rooted onto AMBTC and Clifford Algebra, a method is given by Sau et al. [18]. Incorporation of Clifford algebra in the AMBTC algorithm is to enhance the PSNR value of the approximate image. Ghrare and Khobaiz [19] have given a method which incorporates Walsh Hadamard Transform (WHT) in the Block Truncation Coding (BTC). With maintaining good visible standard of the approximate image, it also enhances the compression ratio. Hence, the development of BTC has reduced the bit rate and also has grown the visible standard of the approximate image.

3 Proposed Approach

First, by the procedure of the SVD [20], SVD-based compression technique is lossy. After discarding some singular values, up to some point, the qualitative loss is not visible. Incorporation of SVD-based compression with DWT [1, 2, 21, 22] and AMBTC [6] is benefiting our proposed method. The fact is that the SVD compression method gives better PSNR values than DWT but DWT gives higher compression ratios than SVD. The compression ratio of SVD-based compression method is acquired by calculating the demanded number of bits to represent S , D , and T over the required number of bits to represent $\overline{S}_{x \times p}$, $\overline{D}_{p \times q}$, and $\overline{T}_{y \times q}$. Firstly, using SVD, the image is compressed. The approximate image is then again compressed by using DWT and AMBTC.

Using SVD, the image is firstly decomposed and then few singular values are discarded. After that, the approximate image is computed. The approximate image is then applied as the input image for the DWT part of our advised technique. At the SVD part of our advised technique, gets the higher compression ratio, if a large number of singular values have discarded. By the multiplication of the SVD-based compression with that of DWT and AMBTC, the overall compression ratio is calculated.

Using a Haar wavelet, into its approximation, horizontal, vertical, and diagonal detail coefficients, the original image is decomposed. The approximation coefficients are taken for the next processing, and all detail coefficients are ignored. With AMBTC, the approximate coefficients are coded. Into a 4×4 bit plane and two quantization levels, the answer gets which is known as a low mean and a high mean. The compression process is known as encoding, and just reverse of encoding steps is called decoding, we get the approximate image here. With the quantization levels, into an image block, the 4×4 bit plane is reconstructed. The image rebuilds, after quantization, through the Inverse DWT of the quantized block. With the following block diagram in Fig. 3, all algorithm steps have been demonstrated. The beauty of this method is to combine SVD, DWT, and the AMBTC together to get its benefit. Incorporation of DWT and AMBTC gives high compression ratio with fine image quality.

The structural outline of the proposed method for image compression is demonstrated in Fig. 1.

4 Proposed Algorithm

Our propose method has the following steps:

Step 1: Input a grayscale image $C(x, y)$ of size $x \times y$ pixels. Apply SVD decomposition to the $C(x, y)$ to get three matrices S , D , and $T \forall$

$$C_{x \times y} = S_{x \times x} \times D_{x \times y} \times (T_{y \times y})^r \quad (1)$$

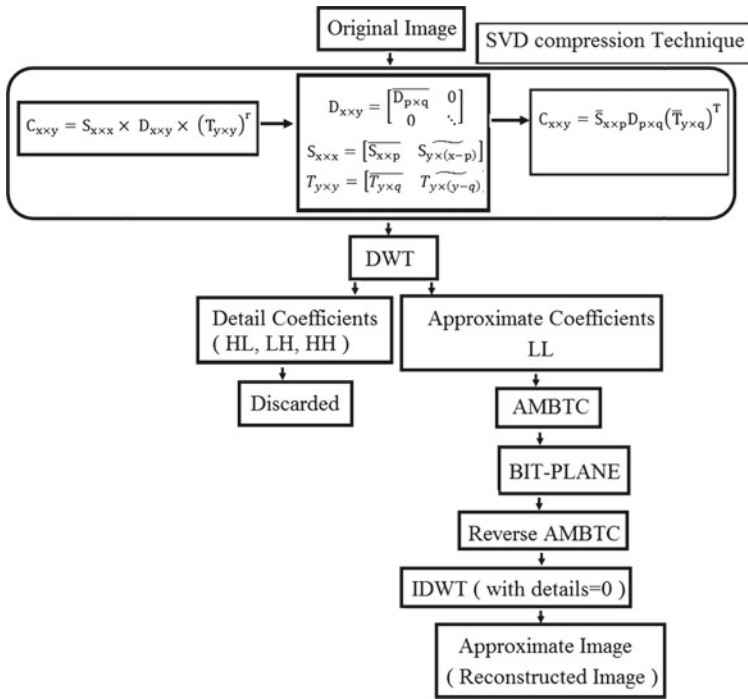


Fig. 1 Schema chart of the novel technique for image compression

Step 2: Considering only the low-rank approximation (Eliminating Singular Values (σ_i)) for compression and now the approximate image is

$$C_{x \times y} = \bar{S}_{x \times p} \times \bar{D}_{p \times q} \times (\bar{T}_{y \times q})^T \tag{2}$$

Step 3: Set number of decomposition level 1.

Step 4: Apply discrete wavelet transforms (DWT) using Haar wavelet to get four output matrices (LL, LH, HL, and HH; among these matrices, LL is known as approximate image and other three are known as detail images). Detail images have three parts, which are vertical details, horizontal details, and diagonal details, take the LL (approximate coefficient) as our image, and set the other detail images to zero.

Step 5: The approximate image (output of Step 4) of size $m \times n$ is split into non-overlapping blocks (B) of the size $m \times m$ (Suppose = 4) and one by one each block is computed. Let $f_o(x_i)$, $x_i \in B$ is the original intensity of the pixel where B represents the set of coordinates of the pixel of the image plane in that block. Also let

$$(m)^2 = k. \tag{3}$$

Step 6: Rooted on pixel intensities, the block B is divided into two sets of pixels B_0 and B_1 as

$$B = B_0 \cup B_1, \text{ and } B_0 \cap B_1 = \phi \tag{4}$$

Here,

$$B_0 = \{t'_1, t'_2, t'_3, \dots, t'_k\}, \text{ and } B_1 = \{t''_1, t''_2, t''_3, \dots, t''_{k-k'}\} \tag{5}$$

This partitioning is presented by allocating one of two levels; say 0 and 1 to the pixel. Therefore, the partitioning might be presented as the bit pattern of 0's and 1's based on the below concept:

$$B_0 = \{0 : f_0(z_i < \overline{M})\} \tag{6}$$

$$B_1 = \{1 : f_0(z_i \geq \overline{M})\} \tag{7}$$

where \overline{M} is the mean intensity of 't' pixels, i.e.,

$$\overline{M} = \frac{1}{t} \sum_{i=1}^t M_i \tag{8}$$

Step 7: From the B_0 and B_1 are the blocks matrix (C) can be computed, mentioned as below for each pixel of that block.

$$C = \begin{Bmatrix} 0, z < x_{td} \\ 1, z > x_{td} \end{Bmatrix} \tag{9}$$

Here, 'td' is considered as \overline{M} and it can be defined through the Eq. (3).

Step 8: The block matrix (C), mean (\overline{M}), and \overline{y} have to be sent for every block to the receiver for reconstructing the image. Here, $\overline{y} = \frac{t\overline{\alpha}}{2}$ and $\overline{\alpha}$ is the sample first absolute central moment.

Step 9: The value 0 of the C matrix will be replaced by the value 'x' and the value 1 of the C matrix will be replaced by the value 'y', for reconstructing the image. The values 'x' and 'y' satisfy the following relation.

$$x = \overline{M} - \frac{\overline{y}}{t - o} \text{ and } y = \overline{M} - \frac{\overline{y}}{o} \tag{10}$$

where

$$\bar{\gamma} = \frac{t\bar{\alpha}}{2} \text{ and } \bar{\alpha} = \frac{1}{t} \sum_{i=1}^t |M_i - \bar{M}|; \tag{11}$$

‘o’ is the counting of ‘1’s in the bit plane (C).

Step 10: Apply Inverse DWT (IDWT with details = 0) to the output of Reverse AMBTC (Step 9) to get approximate image.

5 Results and Discussions

We have likened the demonstration of the proposed algorithms with the Standard BTC [5], AMBTC [6], and DWT-AMBTC [13]. A standard Lena (lena.bmp) (512 × 512, 8 bits per pixel) image is adopted for the assessment of outcomes. BTC, AMBTC, DWT-AMBTC, and proposed technique are implemented in MATLAB code to verify the outcomes. The entire functioning of discrete techniques has evaluated using the Peak signal to noise ratio (PSNR), bit rate (BPP), and Compression ratio (CR). The performance of several techniques in the standard Lena image is given in Table 1. From Table 1, it is authentic that our proposed method outperforms standard BTC, AMBTC, and DWT-AMBTC with respect to CR and PSNR. The value of the compression ratio is high, and our proposed method is compressing the image more than other existing methods. This is the target of the newly proposed method. The resulting images for AMBTC, DWT-AMBTC, and Proposed technique are presented in Fig. 2. From shown Fig. 2, it is clear that the quality of the compressed image of the proposed technique is comparable to other compressed images, compressed by the existing method.

The demonstration of the proposed method SVD-DWT-AMBTC has been assessed for a set of test images of size 512 × 512 with a resolution of the 8-bit per pixel, viz., ‘Barbara’, ‘Boat’, ‘Goldhill’, ‘Lena’, and ‘Peppers’. SVD-DWT-AMBTC is compared with conventional BTC, AMBTC, and DWT-AMBTC. Table 2 depicts the comparative demonstration results of BTC, AMBTC, DWT-AMBTC, and SVD-DWT-AMBTC. The performance is evaluated based on five parameters CR, RMSE,

Table 1 Comparing several techniques for different Parameter on the Famous Lena 512 × 512 image

Algorithm	PSNR (dB)	Bit rate (BPP)	Compression ratio (CR)
BTC	21.45	2.0	4:1
AMBTC	35.37	2.0	4:1
DWT-AMBTC	27.66	0.5	16:1
SVD-DWT-AMBTC	29.35	0.4	20:1

Bold values indicate the best performances obtained in forms of PSNR (dB), BPP, and CR

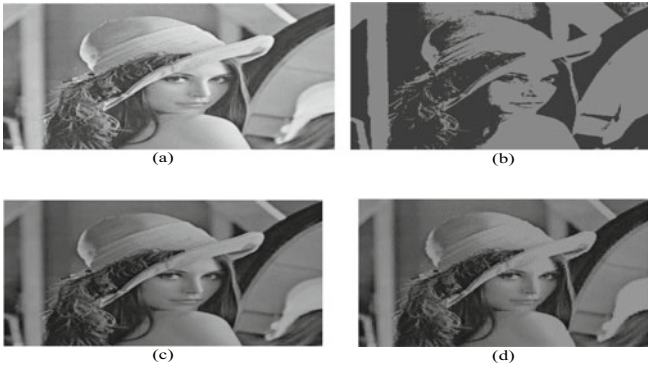


Fig. 2 a Input Lena Image b AMBTC approximate image 2 bits/pixel c DWT-AMBTC approximate image 0.5 bits/pixel d SVD-DWT-AMBTC approximate image 0.4 bits/pixel

Table 2 Lossy performance comparison of SVD-DWT-AMBTC (Proposed), DWT-AMBTC, AMBTC, and BTC image codecs for several Grayscale test images

Image	Method	CR	RMSE	PSNR	SSIM	SC
Barbara	BTC	4	27.17	19.45	0.6894	1.0020
	AMBTC	4	08.19	29.87	0.9747	1.0040
	DWT-AMBTC	16	27.45	19.36	0.7875	1.3406
	Proposed	20	27.45	19.36	0.7875	1.3406
Boats	BTC	4	28.49	19.04	0.6640	0.9965
	AMBTC	4	07.05	31.16	0.9834	1.0021
	DWT-AMBTC	16	29.80	18.65	0.7006	1.4223
	Proposed	20	29.79	18.65	0.7006	1.4223
Goldhill	BTC	4	31.84	18.07	0.6252	0.9969
	AMBTC	4	05.80	32.86	0.9825	1.0017
	DWT-AMBTC	16	31.88	18.06	0.8094	1.4048
	Proposed	20	31.88	18.06	0.8095	1.4047
Lena	BTC	4	21.57	21.45	0.7088	1.0009
	AMBTC	4	04.35	35.37	0.9905	1.0018
	DWT-AMBTC	16	10.35	27.66	0.9265	1.0036
	Proposed	20	08.76	29.28	0.9357	1.0471
Peppers	BTC	4	27.30	19.41	0.6651	1.0044
	AMBTC	4	05.58	33.20	0.9888	1.0017
	DWT-AMBTC	16	27.51	19.34	0.8156	1.4371
	Proposed	20	27.51	19.34	0.8156	1.4371

Bold values indicate the best performances got in terms of CR, RMSE, PSNR (dB), SSIM, and SC

PSNR, SSIM, and SC [23–25]. The standard of the output images is computed using the parameter PSNR. Test images are shown in Figs. 3a, 4, 5, 6, and 7a, and the reconstructed images using 4×4 block by the BTC, AMBTC, DWT-AMBTC, and SVD-DWT-AMBTC (Proposed algorithms) are shown in Fig. 3b–7b, 3c–7c, 3d–7d, and 3e–7e, respectively. The outcomes are reported in Figs. 3–7, which exhibits that the proposed method outperforms significantly over the existing method. From the graph, it is clear that the compression ratio of our suggested method is better than all

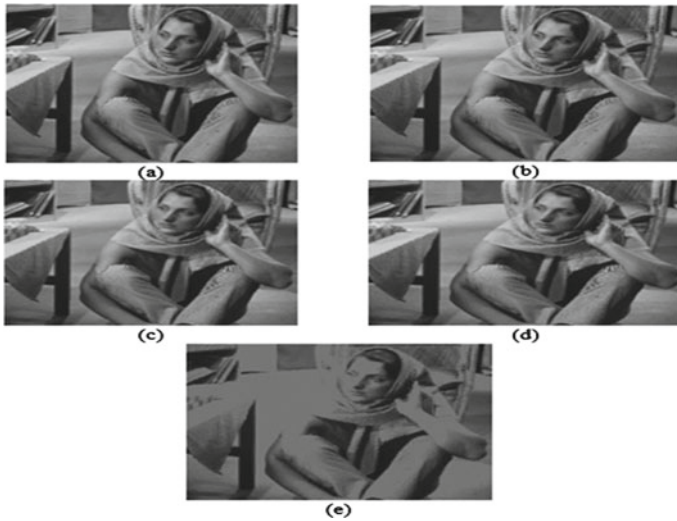


Fig. 3 Barbara a input image; Outcome of b BTC c AMBTC d DWT-AMBTC e Proposed method

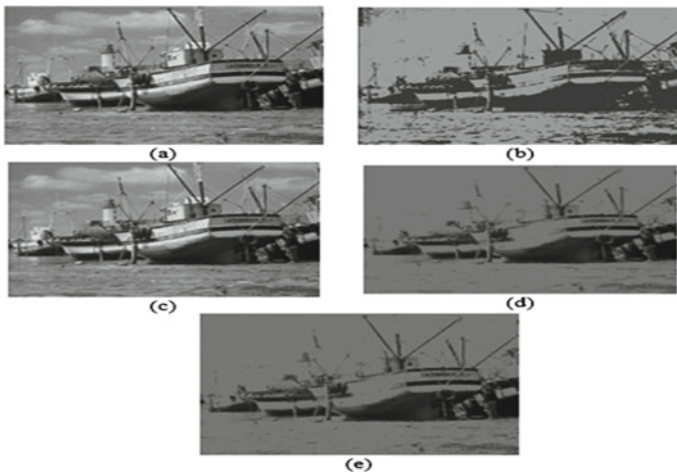


Fig. 4 Boat a input image; outcome of b BTC c AMBTC d DWT-AMBTC e Proposed method

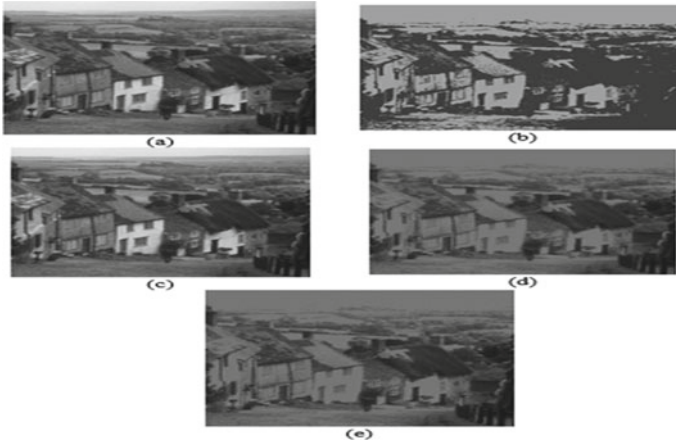


Fig. 5 Goldhill **a** input image; outcome of **b** BTC **c** AMBTC; **d** DWT-AMBTC **e** Proposed method

Fig. 6 Lena **a** input image; outcome of **b** BTC **c** AMBTC; **d** DWT-AMBTC **e** proposed method



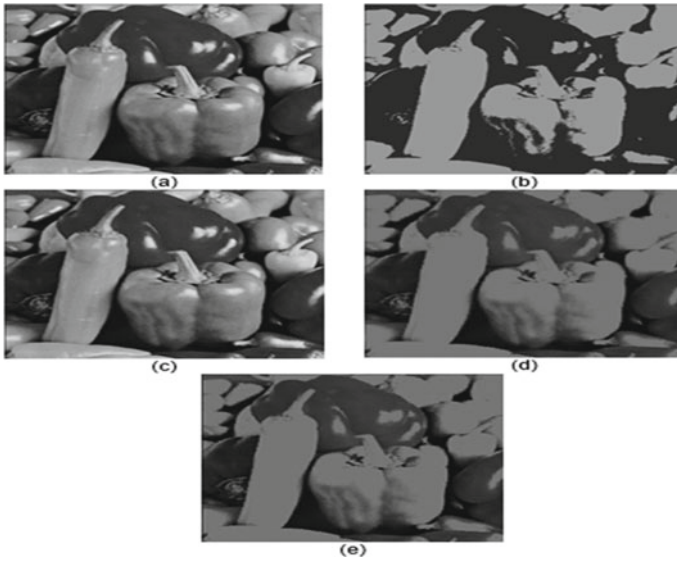


Fig. 7 Peppers **a** input image; outcome of **b** BTC **c** AMBTC **d** DWT-AMBTC **e** proposed method

existing methods and PSNR values are also comparable to other methods. Our work aims to increase the compression ratio with the same reconstructed property of images, which is shown in the graph (Fig. 8).

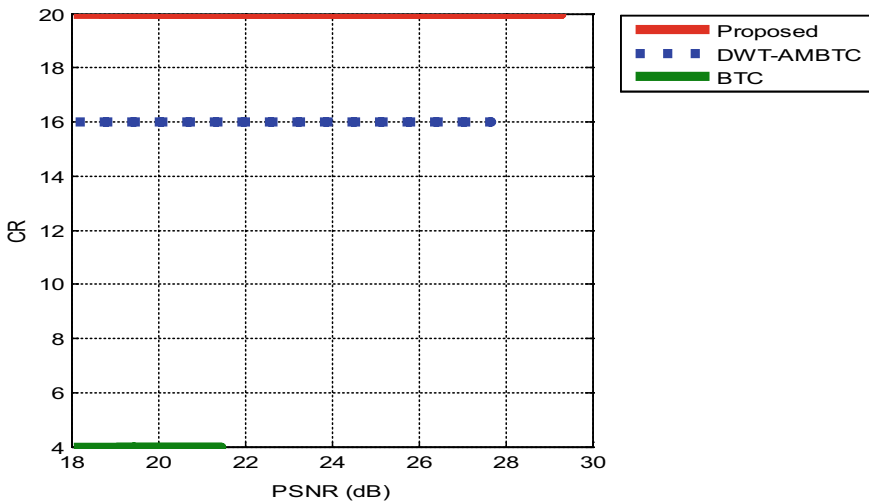


Fig. 8 Comparison of PSNR values (dB) versus CR values of the Boat, Lena, Peppers, Barbara, and Goldhill. Images of size 512×512 for several Grayscale test images

6 Conclusion and Future Scope

In our article, a novel lossy image compression method was offered by using SVD, DWT, and AMBTC. The suggested method used SVD to discard the small singular values and then reconstructed a compressed image. 1-level dyadic wavelet decomposition of the images was performed using the Haar wavelet filters. Here, we got approximate (LL) and detail coefficients (LH, HL, HH). In our aim to attain high compression, high-frequency components (detail coefficients) are discarded. The only approximate image has been compressed again by using AMBTC. The compression ratio has been obtained by multiplication of the SVD-based compression ratio with the AMBTC-based compression ratio, also to a DWT (The approximation sub-band that comprises one-fourth of the total image size is considered for compression.). The outcomes of our newly developed method have also been differentiated with different newest image compression methods. The objective and subjective results are showing the favorable position of our newly developed compression method over state-of-the-art methods. Better results at a compression ratio of 20:1 and a bit rate of 0.4 bits per pixels are shown. This performance evinces that our newly developed method is performing more in effect than standard Block Truncation Coding (BTC), AMBTC, and DWT-AMBTC. The primary outcome of this research was to enhance the compression and image quality of the images. We have taken the singular values (σ) 200 for reconstructing the image. For the Lena Image, we have got PSNR values 29.35 by our proposed method at the bit rate 0.4. For the BTC, AMBTC, and DWT-AMBTC, the PSNR values are 21.45, 35.37, and 27.66. For AMBTC, the PSNR value is 35.37, and the compression ratio is 4. For our proposed method, the PSNR value is 29.35, and the compression ratio is 20, which is far better than AMBTC at the comparable PSNR values.

The limitation of the work is that after discarding more singular values during the SVD process for reconstructing the image, it is difficult to get good reconstructed image quality.

The future scope is that some modification will be made in SVD so that mathematically, more compression can be achieved.

References

1. Ranjan R (2020) Canonical Huffman coding based image compression using wavelet. *Wireless Pers Commun* 117(3):2193–2206
2. Bruylants T, Munteanu A, Schelkens P (2015) Wavelet based volumetric medical image compression. *Signal Processing: Image Comm* 31:112–133
3. Brahim T, Laouir F, Boubchir L, Ali-Chérif A (2017) An improved wavelet-based image coder for embedded greyscale and colour image compression. *Int J Electron Commun (AEÜ)* 73:183–192
4. Latha PM, Fathima AA (2019) Collective compression of images using averaging and transform coding. *Measurement* 135:795–805

5. Usevitch BE (2001) A tutorial on Modem Lossy wavelet image compression: Foundations of JPEG-2000. *IEEE signal processing Magazine*
6. Delp EJ, Mitchell OR (1979) Image compression using block truncation coding. *IEEE Trans Commun* 27(9):1335–1342
7. Lema MD, Mitchell OR (1984) Absolute moment block truncation coding and its application to color image. *IEEE Trans Commun* 32(10):1148–1157
8. Cheng SC, Tsai WH (1993) Image compression using adaptive multilevel block truncation coding. *J Vis Commun Image Represent* 4(3):225–241
9. Cheng SC, Tsai WH (1994) Image compression by moment-preserving edge detection. *Pattern Recogn* 27(11):1439–1449
10. Desai UY, Mizuki MM, Masaki I, Horn BKP (1996) Edge and mean based compression, MIT Artif Intell Lab AI Memo 1584
11. Wu YG, Tai SC (1998) An efficient BTC image compression technique. *IEEE Trans Consum Electron* 44(2):317–325
12. Amarunnishad TM, Govindan VK, Abraham TM (2016) A fuzzy complement edge operator. In: *IEEE Proceedings of the Fourteenth International Conference on Advanced Computing and Communications 2016*, pp. 344–348. IEEE, Mangalore, Karnataka, India
13. Amarunnishad TM, Govindan VK, Abraham TM (2008) Improving BTC image compression using a Fuzzy complement edge operator. *Signal Process Lett* 88(12):2989–2997
14. Venkateswaran N, Kumar JA, Deepak TK (2006) A BTC-DWT hybrid image compression algorithm. In: *IEEE Proceedings of 2006 IET International Conference on Visual Information Engineering*, pp 244–248. Bangalore, India
15. Kumar A, Singh P (2011) Enhanced block truncation coding for gray scale image. *Int J Comput Techn Appl* 2(3):525–530
16. Kumar A, Singh P (2011) Futuristic algorithm for gray scale image based on enhanced block truncation coding. *Int J Comput Inform Syst* 2(5):53–60
17. Mathews J, Nair MS, Jo L (2012) Improved BTC algorithm for gray scale images using k-means quad clustering. In: *proc The 19th International Conference on Neural Information Processing, ICONIP 2012*, pp 9–17, Part IV, LNCS 7666, Doha, Qatar
18. Saua K, Basaka RK, Chanda A (2013) Image compression based on block truncation coding using Clifford Algebra. *International Conference on Computational Intelligence: Modelling Techniques and Applications (CIMTA) 2013*, *Procedia Technology*, 10, 699–706
19. Ghrare SE, Khobaiz AR (2014) Digital image compression using block truncation coding and Walsh Hadamard Transform Hybrid technique, In: *Proc IEEE 2014 International Conference on Computer, Communication, and Control Technology (I4CT 2014)*, pp. 477–480, Langkawi, Kedah, Malaysia
20. Rufai AM, Anbarjafari G, Demirel H (2014) Lossy image compression using singular value decomposition and wavelet difference reduction. *Digital Signal Process* 24:117–123
21. Cheremkhin PA, Kurbatova EA (2019) Wavelet compression of off-axis digital holograms using real/imaginary and amplitude/phase parts, *Nature research, Scientific Reports*
22. Farghaly SH, Ismail SM (2020) Floating-point discrete wavelet transform-based image compression on FPGA, *AEU. Int J Electronics Comm* 124:153363
23. Eskicioglu AM, Fisher PS (1995) Image quality measures and their performance. *IEEE Commun* 43(12):2959–2965
24. Yamsang N, Udomhunsakul (2009) Image quality scale (IQS) for compressed images quality measurement. *Proc The Int Multi Conf Engineers Comp Scient, Hong Kong* 1:789–794
25. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP (2004) Image quality assessment: From error measurement to structural similarity. *IEEE Trans Image Process* 13(4)

Cross-Project Defect Prediction by Using Optimized Light Gradient Boosting Machine Algorithm



Shailza Kanwar, Lalit Kumar Awasthi, and Vivek Shrivastava

Abstract Cross-project defect prediction (CPDP) is a process that trains the prediction model on the source project and predicts the defect of the target project. We present an amended LightGBM algorithm as a defect prediction model for the CPDP process. The hyperparameters of the LightGBM algorithm are optimized by using the Bayesian optimization process to maximize the area under the curve. The proposed model uses the values of optimized hyperparameters to train the data and then classify the faulty and non-faulty instances on test data. To deal with class imbalance issue, the borderline-SMOTE technique is used. Since the dataset used is class imbalanced, so accuracy and sensitivity can mislead the performance results. Therefore for performance evaluation, the false-positive ratio and false-negative ratio should be minimized. The proposed model is further compared with XGBoost and random forest algorithm, which are also optimized in the same way. The proposed model achieves an accuracy of 99.12%. The efficacy of the proposed model is further verified with ROC curves.

Keywords Cross-project defect prediction · LightGBM · Hyperparameter optimization · Bayesian optimization · Class imbalance

1 Introduction

Software is all around us, whether to deal with our financial tasks or to connect with a friend or nowadays due to pandemic, software systems play a significant role in academics. Software systems are becoming more sophisticated to deliver

S. Kanwar (✉) · V. Shrivastava
National Institute of Technology Delhi, Delhi, New Delhi 110040, India
e-mail: shailza@nitdelhi.ac.in

V. Shrivastava
e-mail: shvivek@nitdelhi.ac.in

L. K. Awasthi
National Institute of Technology Hamirpur, Hamirpur, Himachal Pradesh 177005, India

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
H. Sharma et al. (eds.), *Communication and Intelligent Systems*, Lecture Notes
in Networks and Systems 461, https://doi.org/10.1007/978-981-19-2130-8_73

933

maximum benefits, and it raises the number of faults that negatively influence the applications' reliability and resilience [1]. A divergence from software specifications or requirements is generally referred to as a software defect [2]. Such flaws may cause failures or yield unexpected outcomes. Many software quality assurance activities are used to reduce failures and increase software quality. The software defect prediction (SDP) technique is one of the above-mentioned quality assurance techniques that predicts the fault at early stages and reduces the testing time.

SDP is a mechanism to train the defect prediction (DP) model on a historical dataset and then employ the same to predict the probability of fault occurrence in the target projects. SDP further has two categories: within project defect prediction (WPDP) and CPDP. Within project defect prediction (WPDP) takes source data from last releases of itself for training of the prediction model. For a new target project, historical data is either absent or very minimal. So, in this case, source data is taken from different domain projects for training of the prediction model, and the process is termed CPDP.

The simplest method to train the model is to directly use labeled modules from the different projects (the source projects). But software defect datasets generally have the biased distribution. This biased distribution is called the class imbalance problem, in which one class has very few instances (minority class), while the other class has a very high number of instances (majority class). Therefore, constructing a successful CPDP model is a challenging task.

To solve the class imbalance issue, the borderline-SMOTE technique is used to resample the data and to predict the faults amended light gradient boosting machine (LGBM) algorithm is used. The amended LGBM method combines the Bayesian hyperparameter optimization and traditional LGBM algorithm to predict faults in the CPDP method. The correlation-based feature selection technique is used to select features in the dataset from the PROMISE repository.

The remainder of article is arranged in the following manner. The background and related studies of SDP are presented in Sect. 2. The structure of the proposed approach amended LightGBM and implementation details are introduced in Sect. 3. Section 4 presents experimental setup and results, and Sect. 5 brings the study to conclude.

2 Literature Review and Background

It first introduces the literature review on SDP and then background of LightGBM and Bayesian optimization.

2.1 Software Defect Prediction

SDP is a process for predicting software modules that are prone to failure. The use of SDP can reduce the number of demanded testing resources. Many techniques have been proposed to identify more effective SDP models [3–7]. The majority of techniques concentrated on WPDP models, which are only effective in cases when there is enough historical data. Recently, there has been increased interest in the topic of whether CPDP is appropriate to a project when there is a scarcity of historical data. Zimmerman et al. [8] are the first to conduct CPDP on a large scale and have shown that CPDP is crucial for projects which have scarce data to develop prediction model. Only 3.4% of the 622 CPDPs they attempted were effective. The qualities of the data and the method are critical to the CPDP's effectiveness.

He et al. [9] performed CPDP only on open-source projects and discovered that just 0.3–4.7% of the instances might reach good results, depend on the classification algorithm used. These researches have indicated that the CPDP problem was challenging.

To date, researchers have proposed many CPDP approaches, and literature surveys and meta-analysis [10] provide a more thorough study of these CPDP approaches. Some approaches concentrate on the selection of appropriate source projects. For example, Herbold [11] uses the concept of Euclidean distance between projects to choose the appropriate training data. Sun et al. [12] presented a collaborative filtering-based approach to choose the source project in CPDP. Some approaches focus on instance weight setting. Such as Ma et al. [13] proposed a transfer learning-based Naive Bayes algorithm (TNB). This approach shifted CP characteristics to the weights of training data, which were then used to develop a prediction model. Some approaches focus on feature selection and mapping. Such as Nam et al. [14] used TCA, a standard transfer learning approach, to decrease the disparity in feature distributions across different projects. FeSCH is a cluster-based approach suggested by Ni et al. [15]. Other machine learning approaches are considered in the remaining research. Panichella et al. [16], for example, presented CODEP a combination approach based on ensemble learning.

Qiao et al. [17] proposed deep learning techniques to predict defect counts in software projects. Their method involves utilizing publicly accessible datasets to build a deep learning model to predict defect counts. The suggested approach outperformed support vector regression (SVR), fuzzy support vector regression (FSVR) and decision tree regression in tests (DTR). The proposed method reduces the mean square error and squared correlation coefficient significantly. The drawback of their study is the absence of performance comparisons of the suggested approach with other algorithms that use more performance indicators to confirm the efficacy of their system.

Yuan et al. [18] suggested a novel approach ALTRA, which solves the problem of large data distribution by combining active learning with TrAdaBoost. TrAdaboost is used to estimate the weights of labeled modules in target and source projects. Ni et

al. [19] presented EASC, a supervised CPDP approach, and compared the existing supervised CPDP approaches with unsupervised CPDP approaches.

2.2 Light Gradient Boosting Machine Algorithm

Microsoft launched LightGBM in 2017 as an improvement framework based on the decision tree algorithm. It is prefixed with light which implies it is ultra-fast and GBM is abbreviated for gradient boosting machine [20]. LightGBM provides faster training speed, better accuracy and low memory usage. It is different from other tree-based algorithms in the way that tree in LightGBM expands vertically, whereas the trees in other algorithms expand horizontally, meaning LightGBM expands the tree leaf-wise while other expands depth-wise. Leaf-wise algorithms promote better minimization of losses as compared to depth-wise algorithms because the leaf with greatest delta loss is chosen to expand, and when the same leaf is grown, leaf-wise algorithms can minimize losses more than depth-wise. LightGBM uses the histogram-based decision tree approach. Its leaf growth technique, which limits depth and optimizes multithreads, helps to reduce memory usage. This allows for enormous volumes of data to be handled with higher efficiency, lower false alarms and lowered detection rates. The important parameters with default values are given in Table 1.

2.3 Bayesian Hyperparameter Optimization

Hyperparameter optimization is a technique for identifying which hyperparameters produce the best performance for a particular machine learning algorithm when evaluated on a validation set.

LightGBM has several hyperparameters, and it may be challenging to adjust these because hyperparameter selection has a major impact on the model's performance. As a result, it is critical to fine-tune these hyperparameters. The number of leaves, learning rate and other important factors that determine the LightGBM model's performance must be manually changed rather than obtained from training. There are many hyperparameter optimization techniques present such as grids search, random search and artificial neural network training. Grid search offers parallel processing, but it consumes a lot of memory. Random search seeks to obtain the best approximation of the function using random sampling over the search area, and due to this, there may be a chance that it may skip a global optima and cannot ensure an optimum solution. Bayesian optimization is an effective approach for hyperparameter optimization in which objective functions are difficult to evaluate globally. This section introduces the Bayesian hyperparameter optimization (BHO) technique, which was utilized to optimize the hyperparameters of LightGBM model. Most of the optimization problems are black-box optimization problems having a black-box function, and there is no analytical expression for such function or its derivatives. This is when Bayesian

Table 1 Important parameters of LightGBM algorithm

Parameter	Default Value	Description
Task	Train	Sets the activity that we want to do, either train or test
Application	Regression	Specify the task that we want to do regression or binary classification or multi-class classification
Data	Training data file	Specify the name/path of training data file
Num_iterations	100	Total boosting iterations count to be performed
Num_leaves	31	Specify the leaves count that a tree can have
Device	CPU	Specify the device to train model Select GPU for faster training
Max_depth	6	Indicate how far the tree grows to the maximum depth. This parameter deals with overfitting problem
Min_data_in_leaf	20	Specify the minimum count of data that one leaf can have
Feature_fraction	1	Indicates the fraction of features to be used for every iteration
Bagging_fraction	1	Indicates the proportion of data taken for each iteration and is used to accelerate the training process and prevent overfitting problems
Min_split_gain	1	Min gain to perform splitting
Min_child_weight	1	Minimum weight required for an instance of a child leaf
Lambda_l1	0	Used to combat overfitting
Lambda_l2	0	Used to combat overfitting
Num_class	1	Used only for multi-class classification

optimization approaches come in handy the most. The surrogate model is the model that is used by Bayesian optimization to approximate the objective function. Bayesian optimization uses Gaussian processes (GPs) as a surrogate model [21]. A Gaussian process is used to model the unknown objective function space.

3 Methodology

This section discusses the proposed methodology for analyzing the class imbalance problems for software defect prediction. Several real-time applications have recently revolutionized today's world by synchronizing with technology advancements. When dealing with large amounts of data and data that is unbalanced, data classification becomes a challenging undertaking. One of the most fundamental difficulties in data mining is class imbalance. Class imbalance issues are data samples that belong to more classes than other classes. According to the survey, data mining algorithms prioritize the major classes above minorities. Though minority classes are uncommon, they do exist.

3.1 Software Defect Prediction Framework

This research incorporates a fault prediction framework modeled using an amended LightGBM algorithm. This research is aimed to solve the class imbalance issue and maximize the accuracy of the prediction model. The modeling process is divided into six stages.

Stage I: data collection: In this stage the data collected from the relevant dataset which contains 94,148 instances is prepared for training, and classification labels are assigned to the training and testing datasets. The collected attributes are recognized and then imported into MS Excel files.

Stage II: data preparation: Before each model development, a preprocessing step is performed to improve the model's prediction capabilities by removing the redundancies.

Stage III: feature selection: a procedure for identifying the most important characteristics that influence the fault value of software projects.

Stage IV: resampling of the dataset: borderline-SMOTE technique is used to resample the data.

Stage V: machine learning modeling by utilizing the optimized LightGBM.

Stage VI: finally, the prediction results are evaluated and compared with XGBoost and random forest algorithms. The steps involved in the defect prediction process are discussed in below sections.

Data collection: The Jureczko datasets, which Jureczko and Madeyski produced and released in 2010, have become the extensively used datasets in current CPDP

research [22, 23]. The datasets are available in the public PROMISE repository, containing seventy-one versions of thirty-eight different projects, including six closed software projects and thirty-two open-candidate projects. Each instance in the dataset represents a version's class and comprises features spanning 24 features and a class label for defect proneness in that version's class. There may be multiple versions of a project in use in some projects. In this study, we have selected only one version (latest version) of each project which contains more than 100 instances.

Data preparation: Data preparation stage consists of handling missing data, noise removal, normalization and conversion of data. The dataset contains both numerical and categorical values. For smooth functioning of the modeling process, the categorical values of the 'class' attribute are converted into numerical values by label encoding. Label encoding assigns the values from 0 to $k - 1$, where 'k' is the number of classes.

Feature selection: The dataset contains many features; few of them may be irrelevant, which lowers the accuracy and increases the complexity of the model. The feature selection technique selects the most influential features which improve the accuracy of the model. LightGBM and XGBoost come up with inbuilt feature selection techniques. Feature selection is a part of feature engineering, and it is one of the complex parts of data science that has no universal solution. Although LightGBM and XGBoost have inbuilt feature selection capabilities, it is still necessary to select an adequate and reduced feature subset, which further enhances the prediction model's performance. In this research work, correlation coefficient-based feature selection technique is used.

Resampling using borderline synthetic minority oversampling technique: During processing, the imbalanced dataset is balanced using resampling techniques. The data is segregated into two types, namely faulty data and non-faulty data, and is assigned data labels. The data is split into training and testing sets (Training = Ant project Instances, Testing = Camel Project Instances). In this research, a borderline synthetic minority oversampling technique (BLSMOTE) algorithm (a variant of SMOTE) is used to balance the imbalanced data. If outliers observations of minority class fall in the majority class, it creates a line bridge with the majority class and raises an issue for SMOTE. BLSMOTE addresses this problem. During training, borderline-SMOTE algorithms try to understand the boundaries of every class where those borderline and close instances are less frequent than those far away from the boundary. All instances are categorized into three groups: noise, danger and safe. Noise instances are erroneous and reside in the regions covered by majority class instances. Danger instances reside in the zone around class boundaries and coincide with the instances of the majority class. Safe cases are comparatively simpler to spot and are the principal minority class representatives [24]. BLSMOTE begins with the classification of observations of minority classes, and then synthetic instances are generated along the lines between the danger instances and their closest neighbors.

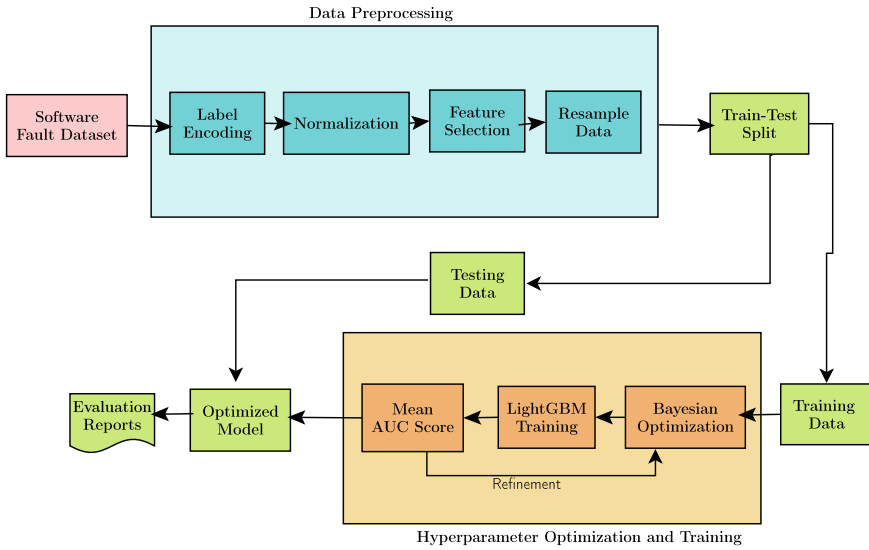


Fig. 1 Process flow diagram for the proposed model

3.2 LightGBM Algorithm with Bayesian Hyperparameter Optimization

Figure 1 depicts the proposed system. Firstly, data is preprocessed, and then LightGBM classifier is trained with the training data obtained after train–test split. To tune the hyperparameters, Bayesian optimization is employed. All of the hyperparameters do not need to be optimized. In this study, we tune six hyperparameters: ‘*bagging_fraction*’, ‘*lambda_l1*’, ‘*lambda_l2*’, ‘*feature_fraction*’, ‘*max_depth*’, ‘*min_split_gain*’, ‘*min_child_weight*’, and ‘*num_leaves*’. While we chose these eight hyperparameters for Bayesian optimization, we left several other critical hyperparameters at their default values. The optimization task focuses on the objective function that maximizes the AUC score with chosen hyperparameters. The mean of fivefold stratified cross-validation of training data is taken for the AUC score. BHO tries to maximize the mean AUC score after each iteration. It chooses the hyperparameters value corresponding to the best mean AUC score for predicting holdout test data when the specified number of iterations is completed. On holdout test data, we employ a variety of assessment criteria to evaluate the performance of the amended LightGBM algorithm.

Table 2 Confusion matrix for classification

Actuals	Predicted		
		Positive	Negative
Positive		True positive	False negative
Negative		False positive	True negative

3.3 Evaluation Metrics

The performance of the proposed model is evaluated by using evaluation metrics such as area under curve (AUC), sensitivity, accuracy and F1-score. Since the dataset is imbalanced, these metrics alone cannot properly evaluate the classification model. For class imbalance issue, accuracy sometimes misleads the classification results. In the case of class imbalance dataset, the classifier should minimize FPR and FNR. The ideal value for AUC is 1.0, which implies that the classifier perfectly classifies the faulty and non-faulty data. All these evaluation metrics are evaluated by the confusion matrix (CM) technique. It provides the summary of the prediction results of classifier algorithm. The CM shows how the classifier got confused while making predictions. Table 2 shows the form of the confusion matrix we have used to evaluate the performance. Receiver operating characteristic curve (ROC curve) evaluation metric is also used for evaluation. It plots the TPR versus FPR at different threshold values.

4 Experimental Results and Discussion

In the experiment, we employ a LightGBM classifier with Bayesian optimization to forecast problematic instances, and we assess performance using several performance measures. Then we compare the proposed model with other machine learning models (XGBoost and random forest). Open-source Python programming programs were used to simulate the work on Google Colab, which utilizes Tesla K80 GPU runs on Intel (R) Xeon (R) CPU having frequency 2.3 GHz and 13 GB of RAM.

This study utilizes the LightGBM algorithm for classification. The hyperparameters of the LightGBM method were first tuned using the Bayesian optimization algorithm, and then performance analysis was performed utilizing those optimized hyperparameters. The objective function in BHO tries to maximize the area under the curve (AUC) score using fivefold stratified cross-validation. We achieve an ideal set of parameters through repetitions of the Bayesian optimization algorithm. Since all the initial iterations yield highest performance in the proposed method as compared to other ensemble models and no significant improvement is expected, we configured the model to halt after 30 iterations. Categorical feature values were encoded using label encoding. The suggested model's best parameters are found in Iteration 16 in

Table 3 Hyperparameter optimization simulation results

Iter	Target	Bagging_ fraction	Feature_ fraction	Lambda_ l1	Lambda_ l2	Max_ depth	Min_child_ weight	Min_split_ gain	Num_ leaves
1	0.928	0.909	0.672	3.014	1.635	6.690	34.070	0.044	42.730
2	0.918	0.992	0.406	3.959	1.587	7.266	46.65	0.008	25.83
3	0.925	0.804	0.766	3.891	2.610	8.905	40.96	0.046	40.39
4	0.935	0.823	0.611	0.716	2.834	7.082	23.66	0.0271	40.26
5	0.931	0.891	0.554	0.093	1.853	7.442	32.76	0.094	38.32
6	0.925	0.871	0.449	3.488	0.180	7.66	35.18	0.021	26.71
7	0.937	0.863	0.391	2.851	1.316	8.944	9.592	0.021	27.39
8	0.930	0.930	0.302	2.332	0.733	5.634	9.967	0.065	26.9
9	0.938	0.839	0.395	4.105	0.291	8.343	9.324	0.097	33.84
10	0.935	0.995	0.583	3.696	0.117	6.128	10.41	0.030	26.49
11	0.945	0.898	0.290	0.593	0.526	8.858	6.781	0.084	43.18
12	0.910	0.929	0.134	1.115	0.134	8.607	5.063	0.080	44.88
13	0.920	0.878	0.592	1.363	0.124	6.269	49.47	0.095	44.83
14	0.945	0.983	0.891	0.044	0.049	8.914	18.71	0.005	43.33
15	0.901	0.978	0.234	4.96	0.344	5.31	49.07	0.067	35.64
16	0.952	0.959	0.870	0.214	0.320	8.813	5.853	0.011	33.79
17	0.913	0.836	0.353	0.178	2.926	5.078	45.2	0.0316	24.1
18	0.929	0.972	0.876	0.278	2.579	5.369	13.85	0.068	44.84
19	0.931	0.989	0.281	0.576	0.034	8.4	20.45	0.019	24.51
20	0.9267	0.878	0.731	4.651	2.986	5.558	24.51	0.021	24.3
21	0.948	0.905	0.576	0.458	2.679	8.421	9.022	0.034	40.1
22	0.885	0.865	0.105	1.095	2.48	5.44	41.55	0.005	44.75
23	0.921	0.825	0.441	0.138	0.116	7.898	49.12	0.098	24.87
24	0.921	0.994	0.809	4.865	2.838	8.922	48.62	0.098	44.64
25	0.907	0.842	0.134	0.148	0.020	7.94	13.34	0.067	35.59
26	0.942	0.833	0.633	0.979	2.625	8.503	5.028	0.082	24.09
27	0.938	0.973	0.718	4.999	2.986	8.684	14.05	0.093	44.53
28	0.889	0.894	0.104	4.356	2.778	8.969	27.69	0.032	43.72
29	0.929	0.987	0.736	0.487	2.876	8.67	37.8	0.035	24.47
30	0.924	0.897	0.695	0.074	0.189	5.185	29.93	0.033	24.14

Table 3. Table 4 illustrates the performance evaluation of the suggested model. In Fig. 2a, b, the confusion matrices summarize prediction outcomes on train data and test data for the proposed model. Another evaluation indicator is the receiver operating characteristic curve (ROC curve), which further checks the proposed model’s performance. Obtained test data AUC is 99% for the optimized LightGBM model on a dataset balanced with borderline-SMOTE.

Table 3 shows the simulation results of the LightGBM algorithm, which is hyper-tuned with Bayesian optimization. In this table, the target column contains the optimized AUC score. Iteration 16 has the highest target value; therefore, corresponding hyperparameter values form the set with optimized values of hyperparameters.

Table 4 shows the performance evaluation of optimized LightGBM for training and testing data. The proposed method optimizes the mean AUC score. The accuracy, sensitivity and AUC score should increase for the model. Since the dataset is

Table 4 Comparison based on AFE

S. No.	Data	Accuracy	AUC score	Sensitivity	FPR	FNR	F1-score
1	Train	0.9663	0.9667	0.9791	0.0198	0.0483	0.9658
2	Test	0.9912	0.9913	0.9951	0.0048	0.0128	0.9911

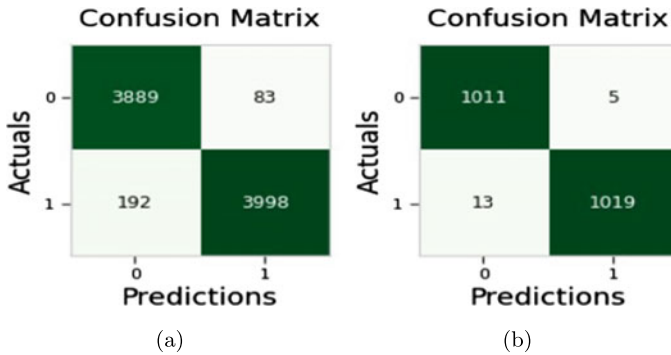


Fig. 2 a CM of training data prediction results of optimized LightGbm algorithm. b CM of training data prediction results of optimized lightGbm algorithm

class imbalanced, the false-negative rate and false-positive rate should decrease. The proposed method has shown better performance for test data.

The confusion matrix of the proposed optimized LightGBM method for training data is shown in Fig. 2a and for test data is shown in Fig. 2b. Figure 2a shows that for training data, the model has correctly classified 3889 positive faults and 3998 as non-faults. Figure 2b shows that in test data prediction results, the model has correctly classified 1011 positive faults and 1019 non-faults.

Figure 3a, b is the ROC curve of the optimized LightGBM for training and testing data, respectively. The ROC curve for training data predictions has covered 0.967 area and for testing data area under the curve is 0.991. Both values are very close to the ideal value of AUC; it implies the proposed optimized LightGMB method has almost perfectly classified the faulty and non-faulty instances.

The proposed model is compared with other ensemble algorithms, which are also optimized. The other algorithms are Xtreme gradient boosting (XGBoost) and random forest. The hyperparameters of these algorithms are also optimized by using Bayesian optimization. The comparison results in Table 5 show that LightGBM has significantly improved prediction performance in CPDP settings. The FPR and FNR values should be low for the class imbalance dataset, and the proposed method has shown minimum values for these both evaluation metrics.

The proposed model is compared with other ensemble algorithms, which are also optimized. The other algorithms are Xtreme gradient boosting (XGBoost) and random forest. The hyperparameters of these algorithms are also optimized by using

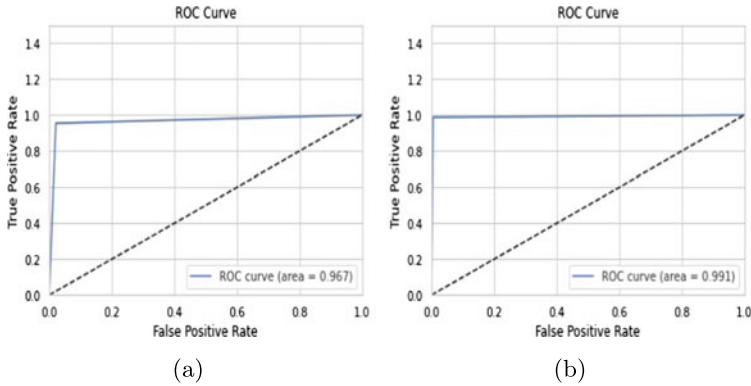


Fig. 3 **a** ROC curve of the optimized LightGBM for training data. **b** ROC curve of optimized LightGBM for testing data

Table 5 Comparison of optimized LightGBM with optimized XGBoost and optimized random forest algorithms

S. No.	Algorithm	Accuracy	AUC score	Sensitivity	FPR	FNR	F1-score
1	LightGBM	0.9912	0.9913	0.9951	0.0048	0.0128	0.9911
2	XGBoost	0.8354	0.8391	0.8037	0.2421	0.1016	0.8436
3	Random forest	0.9013	0.9014	0.9061	0.0917	0.1057	0.9008

Bayesian optimization. The comparison results in Table 5 show that LightGBM has significantly improved prediction performance in CPDP settings. The FPR and FNR values should be low for the class imbalance dataset, and the proposed method has shown minimum values for these both evaluation metrics. The proposed method produces complex trees as compared to other ensemble models. It follows a leaf by leaf split approach rather than a level by level approach, and it leads to obtain higher accuracy as compared to other ensemble models. The hyperparameters of the proposed method are also tuned to avoid overfitting problem.

The ROC curves for all the algorithms also proved the efficacy of the proposed model. The ROC curve of the proposed model for test data is shown in Fig. 3b, and the ROC curve of XGBoost and random forest for test data is shown in Fig. 4a, b, respectively.

The ROC curve of the proposed model has covered the maximum area among three, and XGBoost has covered the minimum area. It implies that the proposed optimized LightGBM algorithm is the best classifier among the three.

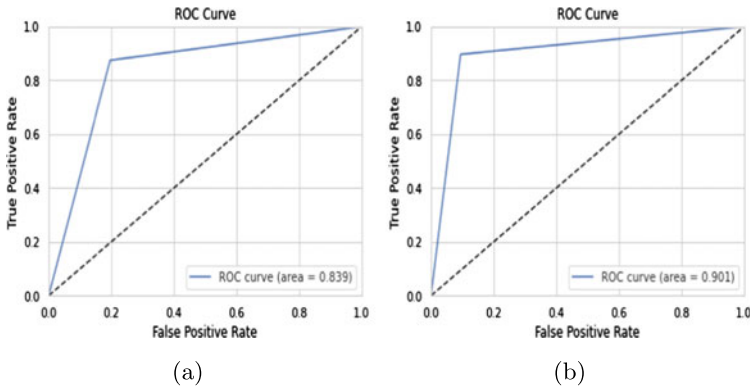


Fig. 4 **a** ROC curve of the optimized XGBoost for testing data. **b** ROC curve of the optimized random forest for testing data

5 Conclusion and Future Scope

In this study, a fault prediction model for CPDP case is proposed. This model is an amended LightGBM algorithm that classifies the faulty and non-faulty instances by using optimized hyperparameters in the classification model. The hyperparameters are optimized by the Bayesian optimization method, which utilizes the Gaussian process as a surrogate model. The proposed algorithm is compared with other ensemble algorithms. The proposed model achieves an accuracy of 99.12%, which is significantly better than random forest (90.13%) and XGBoost (83.54%). The proposed model's performance is significantly better than the optimized random forest and optimized XGBoost. The presented approach is feasible not only in software fault prediction but also in other classification problems having imbalance dataset and large dataset which takes time for training process. In future, instead of considering the source project based on the similarity score, a recommendation-based model can be proposed to select the most appropriate source project to train the defect prediction model.

References

1. Bowes D, Hall T, Petrić J (2018) Software defect prediction: do different classifiers find the same defects? *Softw Qual J* 26(2):525–552
2. Fenton NE, Neil M (1999) A critique of software defect prediction models. *IEEE Trans Softw Engineering* 25(5):675–689
3. Arisholm E, Briand LC, Johannessen EB (2010) A systematic and comprehensive investigation of methods to build and evaluate fault prediction models. *J Syst Softw* 83(1):2–17
4. D'Ambros M, Lanza M, Robbes R (2012) Evaluating defect prediction approaches: a benchmark and an extensive comparison. *Emp Softw Eng* 17(4):531–577

5. Dejaeger K, Verbraken T, Baesens B (2012) Toward comprehensible software fault prediction models using Bayesian network classifiers. *IEEE Trans Softw Eng* 39(2):237–257
6. Elish KO, Elish MO (2008) Predicting defect-prone software modules using support vector machines. *J Syst Softw* 81(5):649–660
7. Bowes D, Hall T, Gray D (2014) Dconfusion: a technique to allow cross study performance evaluation of fault prediction studies. *Autom Softw Eng* 21(2):287–313
8. Zimmermann T, Nagappan N, Gall H, Giger E, Murphy B (2009) Cross-project defect prediction: a large scale experiment on data vs. domain vs. process. In: Proceedings of the 7th joint meeting of the European software engineering conference and the ACM SIGSOFT symposium on the foundations of software engineering, pp 91–100
9. He Z, Shu F, Yang Y, Li M, Wang Q (2012) An investigation on the feasibility of cross-project defect prediction. *Autom Softw Eng* 19(2):167–199
10. Hosseini S, Turhan B, Gunarathna D (2017) A systematic literature review and meta-analysis on cross project defect prediction. *IEEE Trans Softw Eng* 45(2):111–147
11. Herbold S (2013) Training data selection for cross-project defect prediction. In: Proceedings of the 9th international conference on predictive models in software engineering, pp 1–10
12. Sun Z, Li J, Sun H, He L (2021) Cfps: collaborative filtering based source projects selection for cross-project defect prediction. *Appl Soft Comput* 99:106940
13. Ma Y, Luo G, Zeng X, Chen A (2012) Transfer learning for cross-company software defect prediction. *Inf Softw Technol* 54(3):248–256, 106940
14. Nam J, Pan SJ, Kim S (2013) Transfer defect learning. In: 2013 35th international conference on software engineering (ICSE), pp 382–391. IEEE
15. Ni C, Liu W-S, Chen X, Qing G, Chen D-X, Huang Q-G (2017) A cluster based feature selection method for cross-project software defect prediction. *J Comput Sci Technol* 32(6):1090–1107, 106940
16. Panichella A, Oliveto R, Lucia AD Cross-project defect prediction models: L'union fait la force. In: 2014 software evolution week-IEEE conference on software maintenance, reengineering, and reverse engineering (CSMR-WCRE), pp 164–173. IEEE
17. Qiao L, Li X, Umer Q, Guo P (2020) Deep learning based software defect prediction. *Neuro-computing* 385:100–110, 106940
18. Yuan Z, Chen X, Cui Z, Yanzhou M (2020) Altra: cross-project software defect prediction via active learning and tradaboost. *IEEE Access* 8:30037–30049, 106940
19. Ni C, Xia X, Lo D, Chen X, Gu Q (2020) Revisiting supervised and unsupervised methods for effort-aware cross-project defect prediction. *IEEE Trans Softw Eng*
20. Ke G, Meng Q, Finley T, Wang T, Chen W, Ma W, Ye Q, Liu T-Y (2017) Lightgbm: a highly efficient gradient boosting decision tree. *Adv Neural Inf Process Syst* 30:3146–3154, 106940
21. Jia W, Chen X-Y, Zhang H, Xiong L-D, Lei H, Deng S-H (2019) Hyperparameter optimization for machine learning models based on Bayesian optimization. *J Electron Sci Technol* 17(1):26–40, 106940
22. Jureczko M, Madeyski L (2010) Towards identifying software project clusters with regard to defect prediction. In: Proceedings of the 6th international conference on predictive models in software engineering, pp 1–10
23. Jureczko M, Spinellis D (2010) Using object-oriented design metrics to predict software defects. Models and methods of system dependability. *Oficina Wydawnicza Politechniki Wrocławskiej*, pp 69–81
24. Han H, Wang W-Y, Mao B-H (2005) Borderline-smote: a over-sampling method in imbalanced data sets learning. In: International conference on intelligent computing, pp 878–887. Springer

XGBoost Hyperparameters Tuning by Fitness-Dependent Optimizer for Network Intrusion Detection



Miodrag Zivkovic , Luka Jovanovic , Milica Ivanovic ,
Nebojsa Bacanin , Ivana Strumberger , and P. Mani Joseph 

Abstract Network intrusion detection systems are frequently utilized for attack detection and network protection. However, one of the frequent issues intrusion detection systems face is the false positive detections. The research proposed in this paper suggests the hybrid FDO-XGBoost approach for tackling this issue, and according to the extensive simulations and comparative analysis against other common approaches, the proposed method is capable of achieving higher overall classification accuracy when compared to pure XGBoost, random forest and others. The FDO is utilized for adaptive search of the optimal architecture of the XGBoost. Proposed method is validated against widely used NSL-KDD benchmark dataset. The experimental findings indicate that the suggested FDO-XGBoost approach significantly outperforms other approaches in terms of accuracy, having precision and recall with average values of 0.82 and 0.77, respectively. Our method also achieved higher detection rate with all five tested attack classes (normal, probe, Dos, U2R, R2L), slightly outperforming PSO-XGBoost by 1–2% but with a much noticeable difference when it comes to other similar methods.

M. Zivkovic (✉) · L. Jovanovic · M. Ivanovic · N. Bacanin · I. Strumberger
Department of Informatics and Computing, Singidunum University, Danijelova 32,
11000 Belgrade, Serbia
e-mail: mzivkovic@singidunum.ac.rs

L. Jovanovic
e-mail: luka.jovanovic.191@singimail.rs

M. Ivanovic
e-mail: milica.ivanovic.17@singimail.rs

N. Bacanin
e-mail: nbacanin@singidunum.ac.rs

I. Strumberger
e-mail: istrumberger@singidunum.ac.rs

P. M. Joseph
Department of Mathematics & Computer Science, Modern College of Business and Science,
PO Box 100, 133 Muscat, Sultanate of Oman
e-mail: drjosephmani@mcbs.edu.om

Keywords Intrusion detection systems · FDO · FDO-XGBoost · Swarm intelligence · Optimization

1 Introduction

Ever more rapid developments in fields such as information technologies, computing, networking, artificial intelligence and big data, as well as the ever increasing complexity of the systems used, have led to malicious actors developing increasingly advanced attack methods using various vectors of approach. Network security in turn faces more elaborate threats with every passing day, and the need for more competent and potent network intrusion detection system (NIDS) [29] is becoming apparent. The role of a NIDS is to transform intrusion detection into classification and pattern recognition by applying associated algorithms in order to collect, filter, model as well as classify actions on a network [28].

According to the methods they utilized, NIDS can be separated into two groups: those utilizing misuse detection and others relying on anomaly detection. The former constructs an intrusion model of abnormal behaviors and labels actions that correspond with this model as intrusions, while the latter method assembles a model of normal network behaviors and flags actions not in alignment with the model as intrusions.

With an ability to capture and analyze network packets, select the most significant features and associate them with previously documented attack patterns, NIDS can be utilized and observed in real time.

Despite significant advantages over traditional network security systems such as firewalls, NIDS does face some early stage limitations. The system tends to occupy a large portion of available resources, to have a high rate of false positives, and a limited ability to detect and deal with new attacks. This leads to a need for frequent manual intervention. Great efforts have been made by researchers to improve NIDS, and many technologies have been introduced to these systems such as machine learning, feature selection, data mining as well as many others. The introduced methods make the system reproducible while improving adaptability.

Machine learning methods are popular candidates for improving NIDS performance and ensemble learning, a method in which multiple algorithms are applied, providing a more stable performance in comparison with a single algorithm approach. The extreme gradient boosting (XGBoost) [13] methodology utilizes tree structures applying second-order derivatives and regularization terms. The introduced methods result in an increased efficiency and remarkable improvements when detecting minority attack groups. Due to this, XGBoost has been applied to many academic as well as industrial problems [32]. Despite many advantages, the XGBoost model possesses many parameters that govern performance. Determining adequate values for these parameters manually dramatically increases complexity.

Swarm intelligence, a subset of artificial intelligence (AI) algorithms, often referred to as nature-inspired metaheuristics models the behavior of animal groups

that form a collective intelligence by acting independently while exchanging information with others. This mechanism eventually leads the group toward optimal regions within a search space. A relatively novel swarm intelligence algorithm, the fitness-dependent optimizer (FDO) [1] based on particle optimization (PSO) [21], models bee swarming, reproduction and collective decision making. It has proven promising when applied to optimization problems, demonstrating great stability during both exploration and exploitation phases.

This work proposes a FDO-XGBoost hybrid model that hybridizes machine learning and swarm intelligence optimization algorithms. The proposed model utilizes the superior search capabilities of the FDO to optimize the parameters of XGBoost.

The main contributions of this work are as follows:

1. Development of a novel FDO-XGBoost model, based on XGBoost classifier that uses FDO to dynamically optimize its parameters, in turn improving performance of NIDS. Increasing detection accuracy of various types of attacks, particularly when detecting minority attack groups.
2. The evaluation of proposed FDO-XGBoost models performance. Measurements of both overall metrics and the metrics of each class are taken, and a comparison against XGBoost as well as other ensemble learning classification methods is performed. The NSL-KDD dataset is utilized as a benchmark for the conducted evaluations.

The remainder of this work is organized according to the following. Section 2 discussed related works in the field. Section 3 introduces the FDO algorithm showing the governing equations, rules and pseudocode. Section 4 covers the technical aspects applied while constructing the FDO-XGBoost model. The conducted experiments and a comparative analysis are presented in Sect. 5. Finally, Sect. 6 offers a conclusion to the work.

2 Preliminaries and Related Work

2.1 Preliminaries

One of the most popular machine learning techniques, gradient boosted decision trees (GBDT), is the basis of the XGBoost algorithm. Unlike GBDT, XGBoost is faster because it supports parallel processing and it is able to reach high accuracy faster and easier by utilizing linear classifiers. Furthermore, XGBoost is also able to get pretty good estimates of the cost function when executing Taylor expansion because it brings in a second derivative that has higher accuracy as a result [20].

In the optimization process of the XGBoost model's objective function, results of every new step depend on the results from previous steps. Equation for the t -th objective function is shown as follows: [20]

$$(obj)^t = \sum_{i=1}^n l(y_i, \hat{y}_i^{t-1} + f_t(x_i)) + \Omega(f_t) + \text{constant} \tag{1}$$

where t -th rounds' loss term is shown as l , constant term is shown as constant, and model's regularization term is shown as Ω .

The equation for the regularization term Ω along with customization parameters γ (lasso regression) and λ (ridge regression) is shown below. These two parameters help prevent the overfitting of the XGBoost model. If their values are low, the tree structure becomes more complex. On the other side, if the values of these parameters are high, the tree structure becomes less complex.

$$\Omega(f_t) = \gamma \cdot T_t + \lambda \frac{1}{2} \sum_{j=1}^T w_j^2 \tag{2}$$

Approximation of the objective function is achieved by utilizing second-order Taylor expansion on Eq. (1) as shown below:

$$obj^{(t)} = \sum_{i=1}^n \left[l(y_i, \hat{y}_i^{t-1}) + g_i f_t(x_i) + \Omega(f_t) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t) + \text{constant} \tag{3}$$

where the first and the second derivatives are shown as g and h , respectively, and are depicted as follows:

$$g_i = \partial_{\hat{y}_i^{t-1}} l(y_i, \hat{y}_i^{t-1}) \tag{4}$$

$$h_i = \partial_{\hat{y}_i^{t-1}}^2 l(y_i, \hat{y}_i^{t-1}) \tag{5}$$

At this point, the solutions can be acquired. If we take regularization term from Eq. (2) and the first and second derivatives from Eqs. (4) and (5) and put them in Eq. (3), we get the final equations as follows:

$$w_j^* = - \frac{\sum g_i}{\sum h_i + \lambda} \tag{6}$$

$$obj^* = - \frac{1}{2} \sum_{j=1}^T \frac{(\sum g_i)^2}{\sum h_i + \lambda} + \gamma \cdot T \tag{7}$$

where the score of loss function is shown as obj^* and the solution weights are shown as w_j^* . The lower the value of obj^* , the better the composition of the tree is.

2.2 Related Work

There have been numerous applications of machine learning algorithms for NIDS [18], but XGBoost turned out to be the most optimal one. Particularly on the NSL-KDD dataset, XGBoost had the most accurate results compared to the SVM, NB and random forest. The other models turned out to be too slow on large datasets, unstable and struggled with processing the missing values. Still, XGBoost has the ability to obtain the best set of parameters which is then able to fix other parameters' values in order to have a good overall performance [20].

Swarm intelligence algorithms have been previously used in designing the machine learning algorithms, especially in intrusion detection [3]. XGBoost parameters have a heavy impact on how the final model will perform. The reason why the swarm intelligence algorithms are used is because they are able to find the most optimal solutions from the presented data and with the parameters of the appropriate algorithm. As a result, many traits of the machine learning models such as input features, parameters and weights were able to be successfully optimized [30]. Another advantage of using the swarm intelligence algorithms is that they can also be used for testing and selecting the most favorable combination of algorithms so that the most optimal results could be achieved [2].

A combination of swarm intelligence and machine learning algorithms has been successful in resolving many prediction problems [26]. In the work of Duan et al., ICA-XGBoost model has been proposed in order to calculate the strength of recycled aggregate concrete [16]. The performance of ICA-XGBoost has been tested and compared to the performance of ICA-ANN, ICA-SVR and ICA-ANFIS models, where the novel ICA-XGBoost performed the best. Le et al. presented a paper where PSO-XGBoost was conducive in controlling the heating load for the buildings. XGBoost estimates the heating load first, and then PSO optimizes XGBoost model's performance. The proposed technique was compared with the classic XGBoost model, support vector machine (SVM), random forest (RF), Gaussian process (GP) and classification and regression trees (CART), where many building assets were taken into consideration such as overall height, roof area and wall area. PSO-XGBoost outperformed all the other models once again. These examples clearly show a vast usage of the previously mentioned algorithm combination.

Even though a union of these two algorithms results in enhanced performance, a huge variety of experiments have not been conducted as most results only show comparative analysis of the overall metrics. Also, in NIDS it is somewhat difficult to detect uncommon attacks such as R2L and U2R, but Dos attacks are able to be detected with ease. The reason for that is the shortage of data on R2L and U2R attacks. Hence, the importance for improving network intrusion detection for less common types of attacks [20]. When it comes to malicious node detection, NIDS is even able to tell their future behavior. A combination of feature subset selection (FSS) that is based on multiobjective particle swarm optimization (MOPSO) and fast learning network (FLN), performance of NIDS was improved regarding the evaluation criteria [27].

SI-based methods have been frequently employed in recent years in other domains as well, especially to target NP-hard challenges that belong to the computer science and information technology fields. Important applications include global numerical optimization problem [10], wireless sensor networks efficiency, localization and prolonging the overall lifetime [33], cloud task scheduling [12], artificial neural networks training and feature selection [4–6, 8, 11, 14, 17, 23, 24], assisting prediction of COVID-19 cases [34] and MRI classification CNN optimization [7, 9].

Taking into consideration previously mentioned studies, we propose a novel combination of FDO and XGBoost. We are using FDO to enhance the XGBoost and are assessing the performance of FDO-XGBoost on the NSL-KDD dataset. Our findings show that the overall performance of the proposed FDO-XGBoost model has been advantageous in most classes.

3 Fitness-Dependent Optimizer

Modeled after reproductive behaviors observed in bee swarms, the algorithm’s primary functions mimic bee scouts when locating a suitable hive among multiple potential candidates. Scout bees are represented by search agents, and each one poses a possible solutions in the algorithm. Optimal hive selection, from among good candidates, is considered a process of convergence toward an optima.

The algorithm starts by creating randomized agents $X_i = (i = 1, 2, 3...n)$ scattered across the search space. Their locations outline new possible hive positions and potential solutions for the algorithm. Agents try and attain higher-grade solutions by searching additional locations in a random manner. When a better candidate is located, the previous is discarded. If the direction an agent has chosen to move in does not lead toward a more promising candidate, it will assume its previous course, in hopes of attaining a better result. Nevertheless if this course proves less fruitful, the agent will continue to move toward the current located optima, assuming it is the best solution attained thus far.

Mimicking bees, the artificial agents initially traverse the search space randomly guided by a mix of random walk and fitness weight mechanisms. Whenever an agent moves, it does so by adding *pace* to its current position and does so in hopes of locating a more optimal solution. This movement is described in Eq. (8):

$$X(i, t + 1) = X(i, t) + \text{pace} \tag{8}$$

where X is a search agent (artificial bee scout), the current agent is represented by i , and the current locations by t , while *pace* is the speed and direction of the agents movement. Fitness weight fw mostly defines *pace*, while the direction of *pace* remains fully governed by a random mechanism. The value of fw is calculated according to Eq. (9):

$$fw = \left| \frac{x_{i,t}^* \text{ fitness}}{x_{i,t} \text{ fitness}} \right| * wf \tag{9}$$

where $x_{i,t}^* \text{ fitness}$ represents the fitness function value of the optimal solution attained so far, $x_{i,t} \text{ fitness}$ the fitness function value of the current solution, while the weight factor is represented by wf . The role of wf is to regulated fw , and it can be either 0 or 1. When the value is 1, convergence is high, while the chance of coverage is low. Moreover if the value is 0, it is not affecting Eq. (9) and can be discarded. Setting the value to 0 can result in an increase in search stability, except in cases when the fitness function value is fully dependent on problem’s optimization where the result can be opposite. Regardless fw should have a value in the $[0, 1]$ range. In some cases, when testing the current global optima or a solution identical to it, fw can assume the value of 1. Furthermore in the case of $x_{i,t}^* \text{ fitness} = 0$, fw will be equal to 0. In a case where $x_{i,t} \text{ fitness} = 0$, division by zero needs to be avoided. These issues are countered by the following rules outlined in Eq. (10):

$$\left\{ \begin{array}{l} fw = 1 \text{ or } fw = 0 \text{ or } x_{i,t} \text{ fitness} = 0, \text{ pace} = x_{i,t} * r \\ fw > 0 \text{ and } fw < 1 \left\{ \begin{array}{l} r < 0, \text{ pace} = (x_{i,t} - x_{i,t}^*) * fw * (-1) \\ r \geq 0, \text{ pace} = (x_{i,t} - x_{i,t}^*) * fw \end{array} \right. \end{array} \right\} \tag{10}$$

where r represents a random value in the $[-1, 1]$ range. Among many random walk implementations, Levy fight has been selected due to its distribution curve providing stable movements [31].

The mathematical complexity for each iteration of the FDO algorithm is $O(p * n + p * CF)$, p showing population size, n problem dimension and CF the cost of the objective function. The algorithm has a space complexity of $O(p * CF + p * \text{pace})$ for all iterations, where pace is used to store the best previous space. While space complexity remains constant, time complexity depends on the number of iterations. A simple calculation mechanism, in comparison with other algorithms in the same class [21], gives FDO an advantage, only needing to determine fitness weight and one random number per agent.

3.1 FDO Single-Objective Optimization

When the FDO algorithm is tackling a single-objective optimization problem (FDOSOP), the agent population’s random generation within the search space is governed by upper and lower bounds. In every iteration, the best global agent is selected, and Eq. (9) is computed giving fw . Following this fw is evaluated, in case of $fw = 1$ or 0 , as well as if $x_{i,t} \text{ fitness} = 0$, row 1 in Eq. (10) is used to compute pace . Alternatively, if $0 < fw < 1$, a random value for r is generated in the $[-1, 1]$ range. In case of $r \geq 0$, row 3 in Eq. (10) is used for pace computation, while in

```

Initialize agent population  $X_{i,t} = (i = 1, 2, 3, \dots, n)$ 
while number of iterations < iteration limit ( $t$ ) do
  for each agent  $X_{i,t}$  do
    Find the fittest agent  $X_{i,t}$ 
    Create random-walk  $r$  within a  $[-1, 1]$  range
    if  $X_{i,t} == 0$  then
      |  $fw = 0$ 
    else
      | Compute  $fw$  according to Eq. (9)
    end
    if  $fw == 0$  or  $fw == 1$  then
      | Compute  $pace$  according row 1 in Eq. (10)
    else
      if  $randomnumber \geq 0$  then
        | Compute  $pace$  according to row 2 in Eq. (10)
      else
        | Compute  $pace$  according to row 3 in Eq. (10)
      end
    end
    Compute  $X_{i,t}$  according to Eq. (8)
    if  $X_{i,t+1,fitness} < X_{i,t,fitness}$  then
      | Move agent and store new value
    else
      Compute  $X_{i,t+1}$  according to Eq. (8) using previous  $pace$  value
      if  $X_{i,t+1,fitness} < X_{i,t,fitness}$  then
        | Move agent and store new value
      else
        | Do not move agent
      end
    end
  end
end
end

```

Fig. 1 Original FDO algorithm pseudocode

case of $r < 0$, row 2 in Eq. (10) and an negative value of fw are used. The random assignment of fw sign is done to guarantee a randomized search across all direction.

The mechanism of randomization in the FDO algorithm guides both direction and pace amplitude. In the majority of cases, only direction is randomly selected, while pace amplitude is governed by fw . Additionally, whenever a new potential solution is located, it is compared to the current best using an appropriate fitness function in accordance with the pseudocode as shown in Fig. 1. If a better solution is found, it is saved and the old solution is discarded. However, if the new solution is not an improvement, the FDO algorithm allows the agent to go back to exploring its previous direction with the previous pace value if still available.

When applied to maximization problems, minor alterations are made to the FDO algorithm. An inverted version of Eqs. (9), (11) is used. The algorithms criteria for selecting the optimal agent are also adjusted, if $X_{i,t+1,fitness} < X_{i,t,fitness}$ is replaced by an altered criteria if $X_{i,t+1,fitness} > X_{i,t,fitness}$ for both instances in the presented pseudocode in Fig. 1.

$$fw = \left| \frac{X_{i,t, fitness}}{X_{i,t}^* fitness} \right| * wf * (-1) \tag{11}$$

4 Proposed FDO-XGBoost Model

The XGBoost model is regulated by general, booster and learning target parameters which need to be tuned manually in order to increase its performance. The conducted research applies the FDO in an attempt to choose and optimize the most influential parameters for the model shown in Table 1.

The proposed FDO-XGBoost model pipeline is shown in Fig. 2.

The FDO-XGBoost model is constructed according to the following:

1. Based on the number of parameters that require optimization, select appropriate dimension count, following this generates a randomized agent population. Each search agent has a six-dimensional position vector with a range that covers the entire search space. Every dimensional component of the positional vector corresponds to an XGBoost parameter, meaning that the initialization ranges of components are different. The i -th agent's positional vector in iteration t can be expressed according to Eq. (12):

$$X_{i,t} = [x_{i,t}^{\text{eta}}, x_{i,t}^{\text{max_depth}}, x_{i,t}^{\text{min_child_depth}}, x_{i,t}^{\text{gamma}}, x_{i,t}^{\text{subsample}}, x_{i,t}^{\text{colsample_bytree}}] \tag{12}$$

Table 1 XGBoost model parameters

Parameter	Default value	Range	Explanation
eta	0.3	[0, 1]	Learning rate, reducing the weight of each step
max_depth	6	[0, ∞]	Maximum tree depth, a higher value makes the model learn more specific and local samples
min_child_weight	1	[0, ∞]	Minimum leaf weight, a higher value helps the model avoid learning local optimal solution
gamma	0	[0, ∞]	Relates to the loss function
subsample	1	(0, 1]	Governs randomness rates for trees preventing overfitting
colsample_bytree	1	(0, 1]	Governs sampling feature proportions

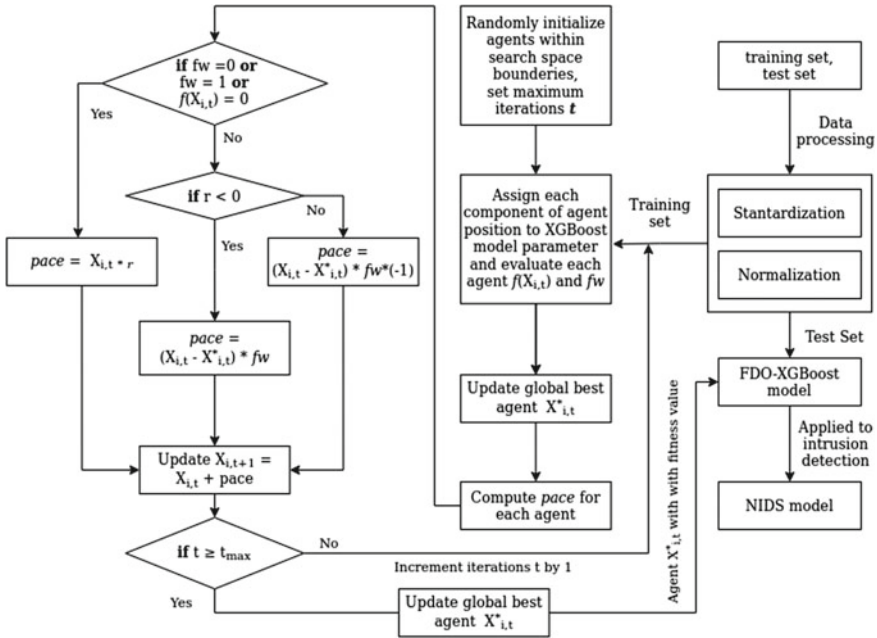


Fig. 2 FDO-XGBoost model pipeline

Considering that all agents share search space, *pace* can be initialized for each agent in the first iteration. The vector *pace* of the *i*-th agent in iteration *t* can be seen in Eq. (13):

$$pace_{i,t} = [pace_{i,t}^{eta}, pace_{i,t}^{max_depth}, text\ pace_{i,t}^{min_child_depth}, pace_{i,t}^{gamma}, pace_{i,t}^{subsample}, pace_{i,t}^{colsample_bytree}] \tag{13}$$

Following this, the performance of the training set is taken as the initial fitness value. The *i*th agents fitness value in the *t* iteration can be seen in Eq. (14):

$$F_{i,t} = (X_{i,t} \rightarrow XGBoost|_{trainingset})_{[metric = X-Rcurve]} \tag{14}$$

Individual best agent *i* in iteration *t* is expressed in Eq. (15):

$$X_{i,t}^* = \max(F_{i,j}), 0 \leq j \leq t \tag{15}$$

2. The position and pace of every agent are updated accordingly. Boundaries are enforced to avoid positions exceeding the search space. New positions are then assigned, and fitness value is computed and compared to the previous optima. If an improvement is observed, the global best is updated.

3. Further iterations are executed until a termination criterion is met, presenting the optimal fitness value and best obtained position.

5 Experiments and Discussion

5.1 Dataset, Preprocessing and Metrics

Similar to the work of [20], the dataset that is being used for comparing different intrusion detection methods is NSL-KDD [25]. One of the key elements for increasing the detection accuracy of the cyber attacks is having only necessary and distinct records. Too many repetitive records may slow down the process, but due to the nature of our model, unnecessary data can be successfully eliminated. The NSL-KDD comes with four classes of attacks (probe, Dos, U2R and R2L) and one normal class [15]. Each of these attack classes can contain multiple attack types. For example, in case of attack class Dos, there are attack types such as back, land, neptune, pod, smurf, teardrop and so on [19]. Training and test data and their class distributions are depicted in Fig. 3.

Each record has 41 features of which nine are discrete and the rest 32 features are continuous. Measurement methods used may vary between features; therefore, in order to assure the consistency of the data and be able to compare it, it needs to be standardized. The number of records is displayed as m , i th eigenvalue of the j th data is displayed as X_{ij} where the value of i is greater than or equal to 1, but smaller than or equal to m . The standardization equation is as follows:

$$\hat{X}_{ij} = \frac{\hat{X}_{ij} - AVG_j}{STAD_j} \tag{16}$$

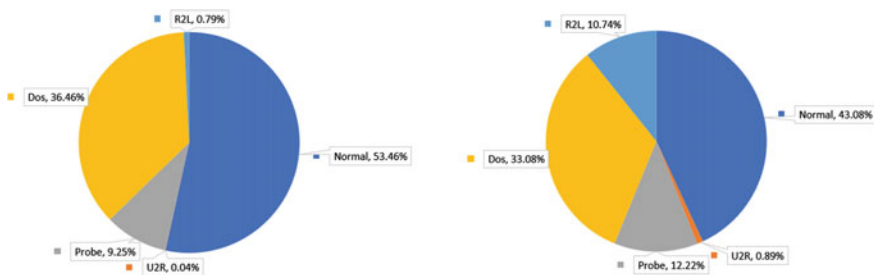


Fig. 3 Class distribution of the training data is shown on the left side (125,973), whereas the class distribution of the test data is shown on the right side (22,544). [20]

where j th feature data's average is shown as AVG_j and j -th feature column data's average absolute error is shown as $STAD_j$.

$$AVG_j = \frac{1}{m} \sum_{i=1}^m X_{ij} \quad (17)$$

$$STAD_j = \frac{1}{m} \sum_{i=1}^m |X_{ij} - AVG_j| \quad (18)$$

With min and max normalization methods, the data is further normalized as follows:

$$\hat{X}'_{ij} = \frac{\hat{X}_{ij} - X_{\min}}{X_{\max} - X_{\min}} \quad (19)$$

where the j th feature's minimum and maximum values are shown as X_{\min} and X_{\max} .

5.2 Metrics

Let us consider one class a positive and other classes negative cases. For this particular experiment and in order to do the evaluation of this model, usual metrics such as precision, recall and f-measures were utilized. Furthermore, we are also utilizing the precision-recall (P-R) curve because it can be used on a great deal of negative cases in order to capture their influence on the models' performance [22]. For example, in the case of classifying our smallest class which is U2R, the other four classes get to be considered negative cases. That way, the performance of our model on U2R is reflected on the P-R curve. Previously described case is unachievable with receiver operating characteristics (ROC) curve because it is not sensitive to the distribution of the classes.

The calculation of the P-R curve's area is presented by the measure called average precision (AP). If we want a better performance, the value of the AP needs to be larger too. Alongside these metrics, mean average precision (mAP) and macro-averaging (macro) are utilized as well.

5.3 Results and Comparative Analysis

After optimizing the XGBoost model with FDO, the obtained FDO-XGBoost model is validated on the NSL-KDD benchmark dataset, and the results are given in Table 2. The experimental findings suggest that the highest precision was obtained on U2R, while also having the lowest recall value. These values suggest that even though the

Table 2 Confusion matrix representing optimal parameters on training data

Class	Precision	Recall	F-score	Samples
Normal	0.68	0.98	0.80	9771
Probe	0.82	0.54	0.66	2421
Dos	0.97	0.85	0.89	7458
U2R	1.00	0.01	0.01	200
R2L	0.97	0.05	0.10	2754
Average/total	0.82	0.77	0.74	22,604

Table 3 Comparison of AP values between AdaBoost, bagging, random forest, XGBoost, PSO-XGBoost and the proposed FDO-XGBoost

Class	AdaBoost	Bagging	Random forest	XGBoost	PSO-XGBoost	FDO-XGBoost
Normal	0.37	0.86	0.89	0.85	0.90	0.91
Probe	0.44	0.80	0.80	0.76	0.79	0.81
Dos	0.82	0.93	0.94	0.93	0.94	0.95
U2R	0.01	0.09	0.08	0.12	0.15	0.16
R2L	0.35	0.32	0.37	0.48	0.49	0.50

samples were predicted correctly, the total amount of samples is very low, and the model predicted a large number of samples from U2R as other classes. Therefore, F-measure can be used to balance precision and recall. In other words, the accuracy, recall and f-measure must be considered together. Other classes perform much better according to the results in Table 2.

The results for the P-R curve (AP values) of the FDO-XGBoost method for each class are shown in Table 3. The results for random forest, bagging, AdaBoost, XGBoost and PSO-XGBoost models were retrieved from [20]. Visual representation of the results of the three XGBoost methods is shown in Fig. 4. The findings suggest that for the classes that contain higher number of samples (i.e., normal, probe and Dos), all models achieve much higher detection rate. The reported values between PSO-XGBoost, random forest, bagging and the proposed FDO-XGBoost are similar, but still in favor of the proposed method (1–2%). PSO-XGBoost was ranked second, while random forest finished on the third position. For the minority of classes, such as U2R and R2L, the proposed FDO-XGBoost was again superior, slightly outperforming the PSO-XGBoost on second (by 1–2%), while the difference over the other methods was more obvious: around 5–6% for U2R and 12–15% for R2L classes, respectively.

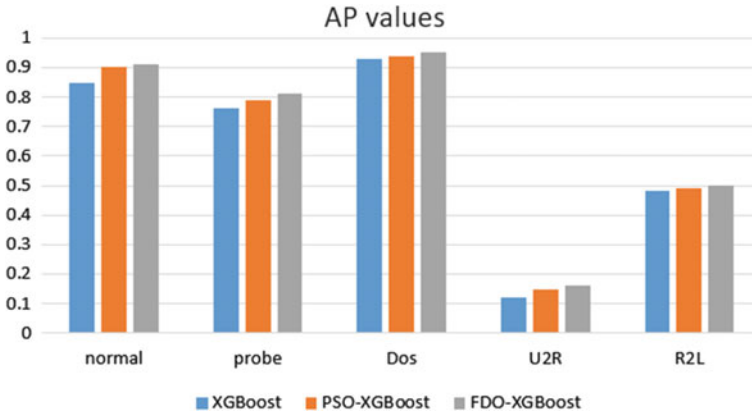


Fig. 4 Visual comparison of AP values between XGBoost, PSO-XGBoost, and the proposed FDO-XGBoost

6 Conclusion

This manuscript proposes FDO-based XGBoost hybrid approach to improve the accuracy of the NIDS, one of the most important challenges in network security. XGBoost has been chosen as it can be utilized to tackle the multi-classification tasks successfully, while the FDO is responsible for finding the optimal hyperparameter values of the XGBoost model. The experimental findings suggest that the FDO-XGBoost model obtained the best accuracy values, when compared to other similar methods. This research also indicates a great potential in applying swarm intelligence approaches in NIDS that opens up a possibility for future work. In the future, we plan to implement other swarm intelligence algorithms to enhance the XGBoost model and also to use the proposed model to solve other classification tasks.

References

1. Abdullah JM, Ahmed T (2019) Fitness dependent optimizer: inspired by the bee swarming reproductive process. *IEEE Access* 7:43473–43486
2. Anochi J, Almeida V, Campos Velho H (2021) Machine learning for climate precipitation prediction modeling over south America. *Remote Sens* 13:2468. <https://doi.org/10.3390/rs13132468>
3. Arul A, Subburathinam K, Sivakumari S (2015) A hybrid swarm intelligence algorithm for intrusion detection using significant features. *Sci World J* 574589. <https://doi.org/10.1155/2015/574589>
4. Bacanin N, Bezdán T, Zivkovic M, Chhabra A (2022) Weight optimization in artificial neural network training by improved monarch butterfly algorithm. In: *Mobile computing and sustainable informatics*, pp 397–409. Springer

5. Bacanin N, Petrovic A, Zivkovic M, Bezdán T, Chhabra A (2021) Enhanced salp swarm algorithm for feature selection. In: International conference on intelligent and fuzzy systems, pp 483–491. Springer
6. Bacanin N, Vukobrat N, Zivkovic M, Bezdán T, Strumberger I (2021) Improved Harris Hawks optimization adapted for artificial neural network training. In: International conference on intelligent and fuzzy systems, pp 281–289. Springer
7. Basha J, Bacanin N, Vukobrat N, Zivkovic M, Venkatachalam K, Hubálovský S, Trojovský P (2021) Chaotic Harris Hawks optimization with quasi-reflection-based learning: an application to enhance CNN design. *Sensors* 21(19):6654
8. Bezdán T, Cvetnic D, Gajic L, Zivkovic M, Strumberger I, Bacanin N (2021) Feature selection by firefly algorithm with improved initialization strategy. In: 7th conference on the engineering of computer based systems, pp 1–8
9. Bezdán T, Milosevic S, Venkatachalam K, Zivkovic M, Bacanin N, Strumberger I (2021) Optimizing convolutional neural network by hybridized elephant herding optimization algorithm for magnetic resonance image classification of glioma brain tumor grade. In: 2021 zooming innovation in consumer technologies conference (ZINC), pp 171–176. IEEE
10. Bezdán T, Petrovic A, Zivkovic M, Strumberger I, Devi VK, Bacanin N (2021) Current best opposition-based learning salp swarm algorithm for global numerical optimization. In: 2021 zooming innovation in consumer technologies conference (ZINC), pp 5–10. IEEE
11. Bezdán T, Stoean C, Naamany AA, Bacanin N, Rashid TA, Zivkovic M, Venkatachalam K (2021) Hybrid fruit-fly optimization algorithm with k-means for text document clustering. *Mathematics* 9(16):1929
12. Bezdán T, Zivkovic M, Tuba E, Strumberger I, Bacanin N, Tuba M (2020) Multi-objective task scheduling in cloud computing environment by hybridized bat algorithm. In: International conference on intelligent and fuzzy systems, pp 718–725. Springer
13. Chen T, Guestrin C (2016) Xgboost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, pp 785–794
14. Cuk A, Bezdán T, Bacanin N, Zivkovic M, Venkatachalam K, Rashid TA, Devi VK (2021) Feedforward multi-layer perceptron training by hybridized method between genetic algorithm and artificial bee colony. *Data Sci Data Anal Opp Challenges* 279
15. Dhanabal L, Shantharajah SP (2015) A study on nsl-kdd dataset for intrusion detection system based on classification algorithms
16. Duan J, Asteris PG, Nguyen H, Bui XN, Moayedi H (2021) A novel artificial intelligence technique to predict compressive strength of recycled aggregate concrete using ica-xgboost model. *Eng Comput* 37(4):3329–3346
17. Gajic L, Cvetnic D, Zivkovic M, Bezdán T, Bacanin N, Milosevic S (2021) Multi-layer perceptron training using hybridized bat algorithm. In: Computational vision and bio-inspired computing, pp 689–705. Springer
18. Haidar A, Verma B (2018) Monthly rainfall forecasting using one-dimensional deep convolutional neural network. *IEEE Access* 1. <https://doi.org/10.1109/ACCESS.2018.2880044>
19. Ieracitano C, Adeel A, Gogate M, Dashtipour K, Morabito FC, Larijani H, Raza A, Hussain A (2018) Statistical analysis driven optimized deep learning system for intrusion detection
20. Jiang H, He Z, Ye G, Zhang H (2020) Network intrusion detection based on pso-xgboost model. *IEEE Access* 8:58392–58401. <https://doi.org/10.1109/ACCESS.2020.2982418>
21. Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of ICNN'95-international conference on neural networks, vol 4, pp 1942–1948. IEEE
22. Liu Z, Bondell H (2019) Binormal precision-recall curves for optimal classification of imbalanced data. *Stat Biosci* 11. <https://doi.org/10.1007/s12561-019-09231-9>
23. Milosevic S, Bezdán T, Zivkovic M, Bacanin N, Strumberger I, Tuba M (2021) Feed-forward neural network training by hybrid bat algorithm. In: Modelling and development of intelligent systems: 7th international conference, MDIS 2020, Sibiu, Romania, October 22–24, 2020. Revised Selected Papers 7, pp, 52–66. Springer International Publishing

24. Bacanin N, Alhazmi K (2022) Training multi-layer perceptron with enhanced brain storm optimization metaheuristics. *Comput Mat Continua* 70(2):4199–4215. <https://doi.org/10.32604/cmc.2022.020449>, <http://www.techscience.com/cmc/v70n2/44706>
25. Protic D (2018) Review of kdd cup '99, nsl-kdd and kyoto 2006+ datasets. *Vojnotehnicki glasnik* 66:580–596. <https://doi.org/10.5937/vojtehg66-16670>
26. Qiao W, Huang K, Azimi M, Han S (2019) A novel hybrid prediction model for hourly gas consumption in supply side based on improved whale optimization algorithm and relevance vector machine. *IEEE Access* 7:88218–88230. <https://doi.org/10.1109/ACCESS.2019.2918156>
27. Einy S, Cemil OZ (2020) Network intrusion detection system based on the combination of multiobjective particle swarm algorithm-based feature selection and fast-learning network, vol 2021. <https://doi.org/10.1155/2021/6648351>
28. Shi T, Zhao Y (2016) Overviews of network intrusion evasion and defense techniques. *Netinfo Secur* 1:70–74
29. Shone N, Ngoc TN, Phai VD, Shi Q (2018) A deep learning approach to network intrusion detection. *IEEE Trans Emer Topics Comput Intel* 2(1):41–50
30. Sun Y, Xue B, Zhang M, Yen G (2018) An experimental study on hyper-parameter optimization for stacked auto-encoders, pp 1–8. <https://doi.org/10.1109/CEC.2018.8477921>
31. Yang XS (2010) *Nature-inspired metaheuristic algorithms*. Luniver press
32. Zheng H, Yuan J, Chen L (2017) Short-term load forecasting using emd-lstm neural networks with a xgboost algorithm for feature importance evaluation. *Energies* 10(8):1168
33. Zivkovic M, Bacanin N, Tuba E, Strumberger I, Bezdán T, Tuba M (2020) Wireless sensor networks life time optimization based on the improved firefly algorithm. In: 2020 international wireless communications and mobile computing (IWCMC), pp 1176–1181. IEEE
34. Zivkovic M, Venkatachalam K, Bacanin N, Djordjevic A, Antonijevic M, Strumberger I, Rashid TA (2021) Hybrid genetic algorithm and machine learning method for covid-19 cases prediction. In: *Proceedings of international conference on sustainable expert systems: ICSES 2020*, vol 176, p 169. Springer

Temperature Estimation in Multi-Core Processors Using Statistical Approach for Task Scheduling



Leena Ladge and Y. S. Rao

Abstract Multiple-core systems are now widely used in application-specific systems and in computations which require high performance. As the world moves from uni-core processors to multi-core processors, there is a need for efficient thermal management. Thermal aware scheduling considers the processor temperature while allocating the tasks to the cores and maintains the temperature below a certain value. In this paper, we present a combination of two statistical models that will help to determine the temperature of individual cores in a multi-core system. Our linear regression model estimates the current temperature of a system from the commonly available parameters of CPU power, CPU usage, etc. Further, our ARIMA model predicts the temperature of the individual core for the next time instance. After deploying this model, an average mean squared error of 1.65°C was obtained. This estimated core temperature can be used to implement thermal aware task schedulers which can then allocate the task to the core with the minimum temperature.

Keywords Multi-core processor · Core temperature estimation · Core temperature prediction · Linear regression model · ARIMA model

1 Introduction

Most of the worlds computers are found in embedded systems. Embedded systems span all aspects of modern life, and there are many examples of their use like telecommunication systems, consumer electronics, household appliances, home automation system, and transportation systems. A significant proportion of those computers are real-time systems. Many real-time systems deal with tasks that involve interaction with the real world. These systems sometimes have to deal with large loads while meeting strict real-time constraints. Real-time systems (RTS) are playing a vital role

L. Ladge (✉)

SIES Graduate School of Technology, Navi, Mumbai 400706, India
e-mail: ladge.leena@siesgst.ac.in

Y. S. Rao

Sardar Patel Institute of Technology, Mumbai 400058, India

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
H. Sharma et al. (eds.), *Communication and Intelligent Systems*, Lecture Notes in Networks and Systems 461, https://doi.org/10.1007/978-981-19-2130-8_75

963

in our society. The important aspect of such systems is that they have to function correctly as well as produce the required result within the specified time. Earlier, RTS were used for few safety-critical domains such as avionics and spacecrafts, but today because of cyber-physical systems (CPS) and Internet of Things (IoT) revolution, RTS is becoming essential in many emerging domains such as smart vehicles, power grids, implantable devices, and robots. In the past few decades, technological developments have picked up tremendous pace. Technologies such as cloud computing, IoT, blockchain and many more which were talked about only on paper a few years back, have now come to life. High-performance and faster processors have played quintessential roles in the rapid development of these technologies.

Multiple-core systems are now widely used in application-specific systems and in computations which require high performance. A multi-core architecture consists of more than one core or processing unit. This reduces the size of hardware, communication delay among multiple cores, thereby enhancing the performance [1]. In order to increase the functionality, computing capacity and also to reduce the number of chips, multi-core processors are being used in real-time embedded systems. The number of cores in a multi-core chip increases due to the technological evolution. These trends in technologies affect the temperature of a chip [2].

2 Literature Survey

Various types of thermal management schemes have been proposed before by many researchers. El Sayed et al. have proposed the algorithm in [3], which reduces the globally shared resources and inter-core communication. The proposed partitioning algorithm will greatly reduce blocking times, improve overall system performance, and reduce energy consumption. In [4], the author has implemented three-phase allocation policy, with first phase of windowing, second phase to allocate task to core, based on energy requirements of task and the operating frequency of the core, third phase is about temperature-aware scheduling. The paper in [5] uses about 21–22 in-built hardware performance counters to estimate the temperature of the processor using linear regression. However, the scope of this work was restricted to uni-processor systems. Chrobak et al. [6] gave an overview of the domain of temperature-aware task scheduling. They have also discussed the challenges and multitude of possibilities that exist in this domain. Chantem et al. [7] presented an optimal phased steady-state mixed integer linear programming (MILP)-based solution that considers the impact of scheduling and assignment decisions on MPSoC thermal profiles to directly minimize the chip peak temperature.

Coskun et al. [8] suggested that conventional thermal management techniques respond only when the temperature rises above threshold level. They have proposed a ARMA-SPRT-based model which could dynamically adapt and predict future temperature of each core. However, they have predicted the future temperature on the basis of previous core temperatures without taking into consideration the impact of other system variables on the core temperature.

Paper [9] states that the different order of execution of the same hot and cool jobs can have different resulting CPU temperatures. One aspect covered in this paper was that there is a need to consider other associated parameters while predicting the CPU temperature in a future time interval. A probabilistic approach to solve the problem of energy minimization was proposed by Zitterell and Scholl [10]. They have proposed a method using statistical execution profiles to determine the expected energy demand after the execution of a task. Zhang et al. [11] primarily dealt with systems having many processors. They used machine learning methods such as multi-layer perceptron, Lasso linear regression Gaussian process model to predict the thermal profile of the entire system. However, their focus was on developing the thermal profile of the entire system. The application of machine learning for the purpose of thermal management on both single and multi-core systems was proposed in [12]. Some methods that were identified in this paper for the purpose of thermal management were, Bayesian learning, neural networks, reinforcement learning, and regression. An analysis of various methods of approximating energy consumption is presented by Martin et al. [13]. They have also discussed the tools to check power and performance along with the applications of machine learning using two case studies. In paper [14], the authors have proposed a gradient boosting machine learning model for temperature prediction of host in cloud data center. Nan Wu and Yuan Xie have presented the survey of application of machine learning computer architecture and system, especially for system modeling and design methodology [15]. They have mentioned how machine learning is useful for enhancing power management.

Thermal-aware scheduling considers the processor temperature while allocating the tasks to the cores and maintains the temperature below a certain limitations. Software approach to reduce operating temperature is relatively simple, low cost, and low risk compared to hardware capabilities. Industrial studies have shown that a reduced operating temperature can increase device lifespan. The cost of chip packaging can be reduced with the help of efficient thermal management.

In this paper, we present a combination of linear regression model and ARIMA model to predict the temperature of all the individual cores in the future time instance. The linear regression model takes into consideration the impact of several core-wise parameters on core temperature. The ARIMA model predicts the core temperature in the next time instance based on previous core temperatures.

3 Methodology

3.1 Generation of Data Set

To estimate the core temperature, a set of some common parameters are considered that can be retrieved on any system irrespective of its architecture. Those parameters are mentioned in the Table 1. Due to the unavailability of data sets with the selected specific parameters, the data set was generated. For this purpose, the MSI After-

Table 1 System parameters

CPU1 temperature	CPU1 clock	CPU1 usage
CPU2 temperature	CPU2 clock	CPU2 usage
CPU3 temperature	CPU3 clock	CPU3 usage
CPU4 temperature	CPU4 clock	CPU4 usage
RAM usage	CPU power	CPU clock

burner utility[16] is used. The Sysbench Benchmark was run in the background and meanwhile MSI Afterburner logged the specified parameters into the file. The data was logged for a period of five minutes. The benchmark was executed again once it was completed until five minutes of data was recorded. Once this data was logged, the data was processed and then converted into a Current Sheet View (CSV) file in a format suitable for the model. A similar method of data logging was done in [11]. As mentioned in Table 1, the CPU temperature is overall processor level temperature, CPU1 temperature is Core1 temperature, and similar is the notation used for other three cores.

3.2 Training the Model

In order to build an algorithm to select core on the basis of temperature for task scheduling, the objective is to create a forecasting model that can determine the core that will have minimum temperature in the next time instance. However, instead of using just previous temperatures, we developed a linear regression model that will first estimate the current temperature based on current core parameters. In order to determine the value of individual core temperatures at a future time instant, it is important to understand and consider the impact of associated input features such as core clock, core usage, CPU power, and RAM usage. This current temperature estimation model tries to capture exactly that.

We started with observing the data of Core1 input and output variables that we had collected. Table 2 shows the correlation matrix between the CPU temperature and associated parameters of CPU Core1. This indicates how the overall temperature of the CPU or processor is influenced by the various parameters associated with Core1. These parameters include the usage of Core1, clock period of Core1, usage of processor, clock period of processor, power consumption of processor, and RAM usage.

Another important reason why this model was developed is that not all systems have an in-built core-wise temperature sensor. Thus, this model tries to estimate the current temperature values based on the values of some of the commonly used parameters which have been described already in the previous section.

Table 2 Correlation matrix of CPU temperature and associated parameters of Core1

CPU temperature	1.000000
CPU1 usage	0.831966
CPU usage	0.884877
CPU1 clock	0.741857
CPU clock	0.747557
CPU power	0.910259
RAM usage	0.323550

To develop a current temperature estimation model, we experimented with three different models—linear regression model, polynomial regression model, and neural network model. These models are developed using the scikit-learn [17] and Tensor-Flow libraries in Python [18].

Linear Regression Model Linear regression is a mathematical model that tries to establish a linear relation between the input variables and the output variable. It does so by determining a best fit line such that the cumulative perpendicular distance of all the points from the line is minimum. Table 3 shows the output of the linear regression model obtained for Core1 temperature, and Table 4 shows the errors that were obtained for Core1.

Polynomial Regression Model Mathematically, polynomial regression and linear regression are very similar to each other. The polynomial regression tries to establish a nonlinear relationship between input and output variables. It tries to determine a higher-order function. Keeping the input and output variables same as before, we developed the polynomial regression model. Table 5 shows the output that was obtained for Core1, and Table 6 shows the cumulative errors obtained for Core1.

Table 3 Comparison of results of linear model for Core1

Data set No.	Actual value	Predicted value
257	52	51.332543
258	48	46.808059
259	47	45.731939
260	48	48.810954
261	47	46.988168

Table 4 Linear regression results for Core1

Mean absolute error	0.7245736328261119
Mean squared error	0.8126314146147705
Root mean squared error	0.9014607116312782

Table 5 Comparison of results of polynomial model for Core1

Data set No.	Actual value	Predicted value
257	52	51.433046
258	48	47.962438
259	47	46.412287
260	48	48.571103
261	47	46.846243

Table 6 Polynomial regression results for Core1

Mean absolute error	5.820999974684468
Mean squared error	278.8384608428288
Root mean squared error	16.698456840164265

Neural Network Model The neural network model is the collection of neurons which are joined together with edges. All the neurons and edges have some random weights assigned which keep changing as and when learning progresses. The output of a given neuron is calculated by the weight of the given neuron and the incoming input. These weights will then be changed using the back-propagation of error and gradient descent to get the closest possible result. Table 7 shows the output that was obtained for Core1, and Table 8 shows the cumulative errors obtained for Core1.

Current Temperature Estimation Model The results obtained from the above three models are summarized in Table 9.

Table 7 Comparison of results of neural network model for Core1

Data set No.	Actual value	Predicted value
257	52	51.778770
258	48	49.454193
259	47	47.523792
260	48	49.659706
261	47	49.961990

Table 8 Neural network model results for Core1

Mean absolute error	1.3836712763859675
Mean squared error	3.491497303818379
Root mean squared error	1.8685548704328645

Table 9 Mean square error results of three different models

Model	Root mean squared error
Linear regression model	0.9014607116
Polynomial regression model	16.698456840
Neural network model	1.8685548704328

Linear regression performed better in terms of root mean squared error as compared to the other two models. Therefore, the linear regression model is used for the purpose of current temperature estimation.

3.3 Testing the Model

Testing ARIMA Model auto-regressive integrated moving average model is a time series model which uses the past values of a variable and learns from it. It then uses it to forecast the value of the variable to give the closest result. ARIMA model is the combination of AR model that is auto-regressive model which focuses only on the past values of the given variable and MA model that is moving average model which focuses only on the past forecasted error. The mathematical equation of ARIMA is as shown in Eq. (1) where C is the constant, β_i is the autoregressive parameter to be estimated, θ_i is the moving average parameter to be estimated and ϵ_t is the white noise.

$$Y_t = C + \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \dots + \beta_p Y_{t-p} + \epsilon_t - \theta_1 \epsilon_{t-1} - \theta_2 \epsilon_{t-2} - \dots - \theta_q \epsilon_{t-q} \tag{1}$$

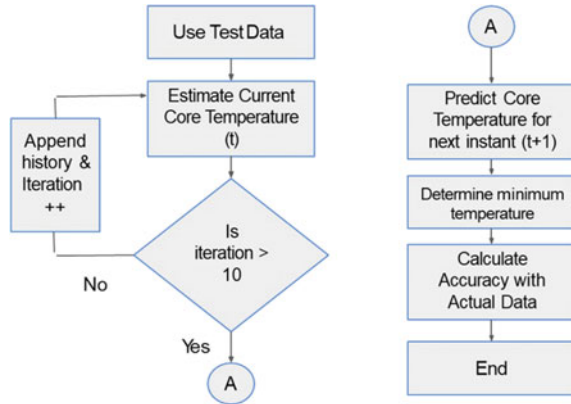
Equation (1) represents the proposed ARIMA model, where ‘ p ’ is the order of the AR model, referring to the number of lags of Y . ‘ q ’ is the order of the MA model, referring to the number of lagged forecast errors. In another words,

Predicted $Y_t = \text{Constant} + \text{Linear combination lags of } Y + \text{linear combination of forecast errors.}$

In thermal-aware scheduling for multi-core processors, it is required to select a CPU core which would have the minimum temperature after completion of the task. Therefore, we have to find the future temperature of the core by using the temperature history of that core and other attributes which affect the temperature. Future CPU core temperature is predicted based on estimated CPU core temperature history. Y_t is derived using $Y_{t-1}, Y_{t-2}, \dots, Y_{t-p}$.

Testing Algorithm During the testing phase, for the first 10 iteration, i.e., till sufficient data is obtained, only linear regression is performed. This is done to consider the impact of associated variables on CPU core temperature. However, after 10 iterations, both linear regression and ARIMA forecasting are used. The flowchart for core temperature prediction for next time instance is shown in Fig. 1.

Fig. 1 Flowchart for temperature prediction in the next time instance



4 Results and Discussion

After combining the linear regression model and ARIMA model, the methodology mentioned in Fig. 1 is followed. The results shown in Table 10 were obtained. Both predicted and expected temperatures for all the cores are displayed. Further, our model suggests the core having the minimum predicted temperature for the next time instance. This suggestion can then be considered by a task scheduler, which can then allocate the task to that particular core.

In Table 10, it is interpreted that, for the pseudotask number 20, the predicted temperature of the Core2 is minimum so Core2 is selected. Same for task number 21 where Core4 has the minimum temperature so it is selected by the algorithm. The mean squared error between the predicted and the expected values of all the four cores is mentioned in Table 11.

The expected and predicted values for all the individual cores can be visualized using the box plot as shown in Fig. 2.

The execution time for the various models used is given below.

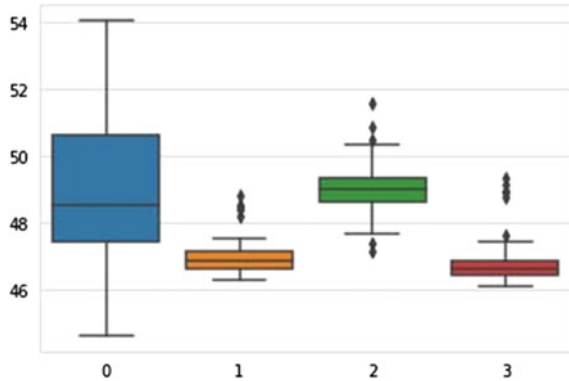
Table 10 Comparison of all four cores and selection of core

Number 20	
Core1: predicted = 47.68 expected = 51.00	Core2: predicted = 46.47 expected = 48.00
Core3: predicted = 47.39 expected = 51.00	Core4: predicted = 46.61 expected = 48.00
Number 21	
Core1: predicted = 48.28 expected = 49.00	Core2: predicted = 46.93 expected = 48.00
Core3: predicted = 48.35 expected = 47.00	Core4: predicted = 46.67 expected = 47.00
Number 22	
Core1: predicted = 50.45 expected = 50.00	Core2: predicted = 46.86 expected = 48.00
Core3: predicted = 49.20 expected = 51.00	Core4: predicted = 46.43 expected = 48.00

Table 11 Error metrics of all the individual cores

Core	Root mean squared error
Core1	1.7831255734914426
Core2	1.2410225835357465
Core3	3.174885572958408
Core4	1.4193039729912649

Fig. 2 Comparison of temperature of all four cores



Linear regression model: 0.011 s, polynomial model: 0.19 s, neural network: 07.70 s, and ARIMA model: 87.94 s.

5 Conclusion

There are many challenges involved in temperature-based scheduling in multi-core processors such as at core-level temperature cannot be obtained directly. Therefore, there is a need of estimating the temperatures at core levels. Also, it is necessary to predict the future temperature of the core for the efficient thermal management. We have used combination of linear regression model and ARIMA model to consider the impact of associated processor features while predicting the future temperature of all the four cores with an average mean squared error of 1.65°C. This serves as an input to the thermal-aware task schedulers.

Even though we have proposed a methodology, the work can be enhanced by creating more representative data set while running a suite of benchmarks. Also, the task-related attributes can also be included for accurate prediction along with processor-level features. Further, the temperatures are determined only for the $(t + 1)$ instance. However, the time instance for which the value is to be predicted should be determined by the characteristics of the tasks.

References

1. Mishra A, Tripathi AK (2014) Energy efficient voltage scheduling for multi-core processors with software controlled dynamic voltage scaling. *Appl Math Model* 38(14):3456–3466
2. Stavrou K, Trancoso P (2007) Thermal-aware scheduling for future chip multiprocessors. *EURASIP J Embedded Syst* 1–15:2007
3. El Sayed MA, Saad ESM, Aly RF, Habashy SM (2021) Energy-efficient task partitioning for real-time scheduling on multi-core platforms. *Computers* 10(1):10
4. Moulik S (2021) Reset: a real-time scheduler for energy and temperature aware heterogeneous multi-core systems. *Integration* 77:59–69
5. Kumar A, Shang L, Peh L-S, Jha NK (2006) Hybdtm: a coordinated hardware-software approach for dynamic thermal management. In: 2006 43rd ACM/IEEE design automation conference, pp 548–553. IEEE
6. Chrobak M, Dürr C, Hurand M, Robert J (2008) Algorithms for temperature-aware task scheduling in microprocessor systems. In: International conference on algorithmic applications in management, pp 120–130. Springer
7. Chantem T, Sharon Hu X, Dick RP (2010) Temperature-aware scheduling and assignment for hard real-time applications on mpsoes. *IEEE Trans Very Large Scale Integr (VLSI) Syst* 19(10):1884–1897
8. Coskun AK, Rosing TS, Gross KC (2009) Utilizing predictors for efficient thermal management in multiprocessor socs. *IEEE Trans Comput Des Integr Circuits Syst* 28(10):1503–1516
9. Zhou X, Yang J, Chrobak M, Zhang Y (2010) Performance-aware thermal management via task scheduling. *ACM Trans Arch Code Opt (TACO)* 7(1):1–31
10. Zitterell T, Scholl C (2010) A probabilistic and energy-efficient scheduling approach for online application in real-time systems. In: Proceedings of the 47th design automation conference, pp 42–47
11. Zhang K, Guliani A, Ogreneci-Memik S, Memik G, Yoshii K, Sankaran R, Beckman P (2017) Machine learning-based temperature prediction for runtime thermal management across system components. *IEEE Trans Parallel Distrib Syst* 29(2):405–419
12. Pagani S, Sai Manoj PD, Jantsch A, Henkel J (2018) Machine learning for power, energy, and thermal management on multicore processors: a survey. *IEEE Trans Comput Des Integr Circuits Syst* 39(1):101–116
13. García-Martín E, Rodrigues CF, Riley G, Grahn H (2019) Estimation of energy consumption in machine learning. *J Parallel Distrib Comput* 134:75–88
14. Ilager S, Ramamohanarao K, Buyya R (2020) Thermal prediction for efficient energy management of clouds using machine learning. *IEEE Trans Parallel Distrib Syst* 32(5):1044–1056
15. Wu N, Xie Y (2021) A survey of machine learning for computer architecture and systems. [arXiv:2102.07952](https://arxiv.org/abs/2102.07952)
16. Msi afterburner. <https://www.msi.com/Landing/afterburner/graphics-cards>
17. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V et al (2011) Scikit-learn: machine learning in python. *J Mach Learn Res* 12:2825–2830
18. Tensorflow. <https://www.tensorflow.org/>

A Generic Ontology and Recovery Protocols for Human–Robot Collaboration Systems



Kamil Skarzynski, Marcin Stepniak, Waldemar Bartyna,
and Stanislaw Ambroszkiewicz

Abstract Recently, human–robot interactions and collaborations have become an important research topic in robotics. Humans are considered as integral components of human–robot collaboration (HRC) systems, not only as objects (e.g. in health care) but also as operators and service providers in manufacturing. Sophisticated and complex tasks are to be collaboratively executed by devices (robots) and humans. We introduce a generic ontology for HRC systems. Description of humans is a part of the ontology. Critical and hazardous (for humans) situations, as well as corresponding safeguards, are defined based on an ontology. The ontology is an extension of the ontology introduced in our previous system. The architecture of the system, a software platform for automatic task accomplishment, is extended to HRC systems. Experiments carried out in a simulated HRC system are to verify the ontology and the architecture.

Keywords SOA paradigm · Human safety · Ontology · Human–robot collaboration

1 Introduction

Physical human–robot interactions are becoming more and more important in many applications, from health care and service robotics to manufacturing. During the interactions, robots must not cause any harm to humans.

Standards ISO 10218 and ISO/TS 15066 [1] provide some guidelines and requirements for designing, planning and executing human–robot collaborative tasks in the form of safeguarding for protecting humans.

K. Skarzynski (✉) · M. Stepniak · W. Bartyna · S. Ambroszkiewicz
Siedlce University of Natural Sciences and Humanities, 3 Maja 54, 08-110 Siedlce, Poland
e-mail: kamil.skar@gmail.com

S. Ambroszkiewicz
Institute of Computer Science, Polish Academy of Sciences, Jana Kazimierza 5, 01-248 Warsaw,
Poland

The ISO standards are not formal and not sufficient to define generic safeguards. No generally accepted solution has been proposed so far to implement these requirements universally. Existing solutions are dedicated to specific scenarios and cannot be generalized to an ontology and protocols. Hence, ISO/TS 15066 may be seen only as a good and approved guidance to create a formal ontology for defining safeguards.

IEEE Standard Ontologies for Robotics and Automation (Std 1872-2015) [2] defines a core ontology that allows for representation, reasoning, and communication of knowledge in the robotics and automation (R&A) domain. This ontology includes generic concepts as well as their definitions, attributes, constraints, and relationships.

From Introduction of IEEE Std 1872-2015 document:

The growing complexity of behaviors that robots are expected to present naturally entails the use of increasingly complex knowledge as well as the need for multi-robot and human-robot collaboration. ... Ontology plays a fundamental role in this context.

In IEEE Std 1872-2015 ontology, humans are needed only for a semi-autonomous robot, i.e. a robot accomplishing a task in which the robot and a human operator plan and conduct the task. It requires various levels of human interaction via *HumanRobot-Communication*, i.e. a transfer of information between humans and robots. Hence, here the human–robot interactions are extremely limited and are not physical.

Humans can perform operations (provide services) that are parts of complex tasks to be accomplished in HRC systems. Humans may be also “objects” of such operations in health care and rescue actions in accidents with human victims.

We are going to define a simple and generic ontology where full human–robot interactions can be described, that is, where humans can be service providers, as well as “objects”.

Based on of this ontology, we explore the possibility of the human–robot collaboration in unstructured environments challenged by frequent failures of robots and humans. Generic protocols for recovery from failures are a solution to this problem. If a robot or a human fails, another robot or a human may accomplish the task.

Human presence and activity in HRC systems may cause critical and hazardous situations for humans. Constraints on human–robot interactions are required in the spirit of the famous Three Laws of Robotics by Isaac Asimov [3].

The constraints, corresponding safeguards, and recovery protocols can be defined on the basis of the proposed ontology. They naturally extend the SO-MRS architecture [4]) where no humans were involved in task accomplishing.

The main contribution of the paper consists of a simple universal upper ontology describing HRC systems, SO-HRCS architecture, and new protocols related to the new module, i.e. Safeguards. Since an occurrence of a critical situation is considered a failure, the protocol for failure handling and recovery from SO-MRS is adopted for SO-HRCS.

The paper is organized as follows: basic concepts and the literature are presented and discussed in Sect. 2. Section 3 presents the software architecture of SO-HRCS. In Sect. 4, we present a generic ontology for the system. Section 5 showcases our approach. Finally, we present our conclusions in Sect. 6.

2 The Idea and Related Work

Heterogeneous open distributed system (HODS for short) consists of environment, and of devices (machines and robots) as well as humans that operate in the environment and may change local states of the environment. For sophisticated tasks, a collaboration of devices and people may be necessary. A device and a human may be considered as objects of the environment, and their state may be subject to change, e.g. their positions.

It is supposed that the devices may be heterogeneous and can be added to the system as well as to be removed without affecting its basic functionality, i.e. the ability for task accomplishing. There are also humans that can perform some elementary tasks. Hence, the class of the tasks is not fixed and depends on the joint capabilities of the devices and people that are currently in the system. Since such tasks cannot be hard-coded in the system, there must be a language for the task specification. Intuitively, a task is an intention to change the local state of the environment. That is, a task consists of precondition and effect. Sometimes, the precondition is not necessary. Precondition specifies the initial local state of the environment, whereas the effect specifies the desired local state of the environment after the task accomplishing. Preconditions and effects are formulas in a formal language. So that, a formal representation of the environment (ontology) is needed. The task is represented by $(\phi \rightarrow \psi)$ where ϕ is the precondition formula, and ψ is the effect formula. Note that here \rightarrow is not the logical implication. Its meaning is to change a local state of the environment.

The principal goal of a HODS is to plan and accomplish complex sophisticated tasks. It is clear that a generic infrastructure in the form of a software platform based on ontology and protocols is needed. In this paper, it is supposed that any HRC system is a HODS.

IEEE Standard (Std 1872-2015) [2] is quite complex. In particular, the process is defined as a primitive notion *that happens in time and has temporal parts or stages*. A simple static ontology without time and processes is sufficient. The simplicity should be viewed here as an advantage. Such ontology was proposed in [4] for adaptive multi-robot systems without humans.

In our approach, processes can be defined using the Service Oriented Architecture (SOA) paradigm from Information Technology, where devices and humans are represented by the services they can provide.

2.1 Related Work

Most of the related work has been presented in [4].

Let us only cite the view on the research on multi-robot Systems by Chitic et al. [5]: *Despite many years of work in robotics, there is still a lack of established software architecture and middleware, in particular for large scale multi-robot systems. Many research teams are still writing specific hardware orientated software that is*

very tied to a robot. This vision makes sharing modules or extending existing code difficult. A robotic middleware should be designed to abstract the low-level hardware architecture, facilitate communication and integration of new software.

The literature concerning *middleware* for HRC systems is rather limited.

Lasota et al. [6] surveyed and categorized prior research addressing safety during human–robot interaction (HRI). The authors identified four main methods of providing safety: control, motion planning, prediction, and consideration of psychological factors. They concluded that ... *ensuring safe HRI remains an open problem. Novel, robust, and generalizable safety methods are required to enable safe incorporation of robots into homes, offices, factories, or any other setting.*

Magrini et al. [7] proposed a specialized and dedicated safety framework to ensure the coexistence of a human operator in a robotic cell in which a standard industrial robot is in motion. It is based on online monitoring of relative human–robot distance using depth sensors.

Extensive literature reviews of the aspects of safety and failures in human–robot interactions can be found in: Honig et al. [8], Robla-Gómez et al. [9] and Villani et al. [10] both concerning industrial environments. Compared to mentioned works, safety in our system is treated on a higher abstraction level and is described using the ontology. Such an approach provides a more flexible and universal way of specifying dangerous situations and reactions to those situations (safeguards).

As to ontology for HRC systems, besides already mentioned IEEE Std 1872-2015, some publications are, in fact, based on this standard ontology. They emphasize the importance of the ontology for Robotics without proposing essentially new (comparing to IEEE Std 1872-2015) ontology; e.g. Fiorini et al. [11] and Kumar et al. [12] to mention the most important ones.

3 Software Infrastructure (Middleware) for HRC Systems

Software infrastructure is necessary to automatically accomplish complex tasks in an HRC system. The architecture of such infrastructure is presented in Fig. 1 (left

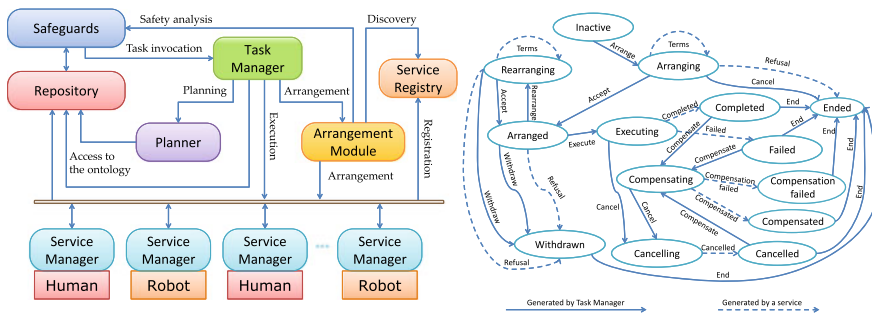


Fig. 1 SO-HRCS architecture, and the state transition diagram of failure recovery protocol

part) where Task Manager, Service Managers, Service Registry, Repository, Planner, Arrangement Module, and Safeguards are software applications that communicate using generic protocols shown in the right part of Fig. 1.

It is supposed that devices and people (providing services in an HRC system) are not isolated, i.e. there is a minimum communication in the system in the form of a network. Each device and each human (e.g. via a mobile phone) can receive and send messages.

Devices and people are represented by their abilities (services) to change local states of the environment.

Each device and each human is autonomous and may provide some services (via its Service Manager) to a client called Task Manager. If a Task Manager has a task to be accomplished, it sends a request (intention) to a Service Manager. Then, the corresponding service may accomplish the task, if it has enough resources and capabilities. If the service agrees, it sends (via its Service Manager) a commitment to the Task Manager. If the terms of the commitment are satisfactory for the Task Manager, it can send a request for the service invocation.

So that, devices and humans provide some services that correspond to some types of elementary tasks they can accomplish. The formal specification (expressed in a language of the common ontology, e.g. OWL-S [13]) of the type of a service consists of a precondition and an effect.

Note that the service type has the same syntactical form as the task.

The type of service (provided by a device or a human) must be published via Service Manager to a Service Registry. A Task Manager may discover the service in Service Registry, and invoke it. This constitutes the essence of the SOA paradigm applied to HRC systems.

Repository is a realization of the common knowledge of the environment representation (ontology, see Sect. 4), and storage of the current maps of the environment, i.e. instances of the ontology. Since the environment may be changed by devices and people, the maps must be updated.

If a Task Manager wants to realize a complex task, i.e. a sequence or partial order of elementary tasks, then some services, that may jointly accomplish the complex task, should be discovered in a Service Registry, and composed into a workflow via the Arrangement Module.

Task Manager is responsible for constructing an abstract plan via the Planner in the form of partial order of service types. Then, appropriate concrete services should be arranged into a concrete plan via the Arrangement Module. Finally, the workflow is executed and the process is monitored. If a failure occurs due to a broken communication or inability of a service to fulfil the arranged commitment, then failure recovery mechanisms must be applied. Simple mechanisms in the form of protocol, see right part of Fig. 1, consist in re-planning, and changing some parts of the workflow to continue the task execution. This constitutes the system architecture proposed in our previous work.

Since people are involved in HRC systems, additional failures related to unexpected events, i.e. critical situations where human safety is at risk, must be introduced in order to protect the people. For this very reason, an additional component of the

software infrastructure is necessary. It is Safeguards where the critical situations are defined.

Critical situations are taken into account by Planner when an abstract plan is needed for a task. Any abstract plan and concrete plan should exclude evident critical situations. Generally, occurrences of critical situations, during task accomplishing, cannot be excluded at the level of planning. It is reasonable to introduce a mechanism for risk assessment in the planning to reduce the occurrences as much as possible.

Each process of a task accomplishing, supervised by a Task Manager, and each service execution, supervised by an appropriate device or a human, must react to the critical situations. The reaction, called *safeguard*, should be of the following form. If a *critical situation* occurs, then an action is executed that transform the critical situation into a *safe situation*.

A critical local situation, as well as the corresponding safe situation, should be defined as formulas in the language of ontology.

Generally, a situation is defined as a formula and identified with the set of local states satisfying the formula.

Let a formula defining a critical situation be called a *critical formula* and a formula defining a safe situation be called a *safe formula*.

Some critical situations and safe situations are universal and independent of concrete applications.

Definition 1 Let ϕ denote a critical situation, and $\psi_1, \psi_2, \dots, \psi_n$ be a finite sequence of safe formulas. Then, *safeguard* is defined as $\phi \rightarrow (\psi_1, \psi_2, \dots, \psi_n)$.

The meaning of a safeguard is as follows. If ϕ is true in the current local state of the environment, then change the state to a safe state where at least one of the formulas $\psi_1, \psi_2, \dots, \psi_n$ is true in that state. The change must be possible, so that one of the tasks $(\phi \rightarrow \psi_1), (\phi \rightarrow \psi_2), \dots, (\phi \rightarrow \psi_n)$ can be accomplished, either by a single service or by a composite service invoked by the appropriate Task Manager. Actually, it is a strong condition.

A task is a safeguard if the task precondition is a critical formula, and the task effect is a safe formula. An example of a safeguard is shown in Sect. 5.

Safeguards are stored in the Safeguards module.

Introduction of safeguards requires an extension of: the ontology of SO-MRS, functionalities of Task Manager and Service Manager, and corresponding protocols. Task Manager must take into account the critical situation in the planning.

A service provider (a device or a human) must monitor the local environment during its service execution. If a critical situation occurs, then the service provider is obliged to report this to the Task Manager, and take appropriate means (if possible) to change the critical situation to a safe one.

If the service provider cannot manage, it means a *critical failure* that must be reported to the Task Manager that is responsible for a recovery from the failure, i.e. for a transformation to a safe local state, if possible. A safeguard stored in Safeguards module is used by the Task Manager as a task to be accomplished in order to change the critical situation into a safe one.

To summarize, the proposed software infrastructure of the HRC system for automatic complex task accomplishing consists of services represented by Service Managers provided by devices and humans, Service Registry, Task Manager (with Planner and Arrangement Module), Safeguards, and Repository of the current maps of the environment. The interactions between them are based on generic protocols for publishing, discovering, composing elementary services, arranging, executing, monitoring and recovery from failures.

An example of a simple task realization in SO-HRCS is shown in Fig. 3.

Note that ontology is the basis for the protocols. It allows specifying local states of the environment, tasks, service types, intentions, commitments, critical situations for humans, and situations resulting from failures. The protocols and the ontology determine the new architecture called Service-Oriented Human-Robot Collaboration System (SO-HRCS for short). SO-HRCS is a substantial extension of SO-MRS (Service-Oriented Multi-Robot System) architecture introduced in Skarzynski et al. [4].

SO-HRCS architecture allows several independent Task Managers, Service Registries, and Repositories. Note that the presented approach is at a higher level of abstraction than Robot Operating System (ROS) that is usually used to implement services on the devices.

3.1 Protocol for Failure Handling and Recovery

Since some ideas and methods are adopted from electronic business transactions, a realization of a task is called *a transaction*. Participants of a transaction are the services involved in the corresponding task accomplishing.

A transaction is successfully completed if its task is accomplished. The transaction mechanism designed for handling failures has the following properties.

1. Failed services may be replaced by other services during task accomplishing.
2. General plan may be changed.
3. Transaction ends either after successful completion of the task, or inability to complete the task or cancellation of the task.

In distributed systems, a communication protocol specifies the format of messages exchanged between two or more communicating parties, message order, and actions taken when a message is sent or received. Based on the OASIS Web Services Transaction (WS-TX 1.2) standards (2009) <https://www.oasis-open.org/committees/ws-tx/>, a transaction protocol, called Failure Recovery Protocol (FRP, for short) proposed in Skarzynski et al. [4], is adopted for HRC systems. FRP defines states of services, and types of messages exchanged between Task Manager and services, see Fig. 1.

Task Manager uses FRP to initialize particular phases of service invocation, monitor their progress, and take additional actions, e.g. compensations. Task Manager

invokes a service by sending the input data (specified in the commitment) to the service via its Service Manager. The service sends messages (via its Service Manager and according to the protocol) to notify Task Manager about the performance of the delegated sub-task specified in the intention.

After successful execution, the service (via its Service Manager) sends to the Task Manager the confirmation of sub-task completion, e.g. changing local situation in the environment to the one specified in the intention. Task Manager can also stop the service execution before its completion. This may be caused by the task cancellation by the client, a failure during the execution of other services in the plan (that cannot be replaced), or by changes in the environment making the current plan infeasible.

A robot (or a human as a service provider) may not be able to complete a sub-task. In this case, its Service Manager notifies the Task Manager by sending a detailed description of the problem. On this basis, the Task Manager can take appropriate actions. If a Service Manager is not able to send such information, the Task Manager must invoke appropriate cognitive service (a patrolling robot or a human, if available) to recognize the situation resulting from the failure.

Compensation is performed either after a cancellation of a sub-task execution by a service or after the occurrence of a failure that interrupts the execution. It is designed to restore the original state of the environment before the execution. Since restoring that situation is sometimes impossible, the compensation may change the situation, resulting from the failure, to a situation from which the task realization can be continued. Note that even for simple transportation tasks (that seem to be simple) a universal failure recovery mechanism and corresponding compensations are not easy to design and implement. A concrete plan should contain predefined procedures for failure handling and compensations.

Note that in the protocol, humans and devices are viewed only as service providers. Critical situations, that occur during sub-task execution by a service, and cannot be managed by the service, are considered failures in the protocol.

3.2 *Services*

There are the following three kinds of services in HRC systems:

1. Physical services that may change local situations in the physical environment.
2. Cognitive services that can recognize situations described by formulas of the language of the ontology.
3. Software services that process data.

Physical services and cognitive services can be performed by humans.

Complete service description consists of the following elements:

- Name of the type of service, i.e. name of an action that the service performs.
- Specification of the inputs and outputs of the service.

- The condition required for service invocation (precondition), and the effect of service invocation.
- Service attributes as information about the static features of a service, e.g. operation range, cost, and average realization time.

Precondition and effect are defined as formulas of a formal language (e.g. OWL [13] or Entish [14]) describing local situations in the environment. Entish is a simplified version (without quantifiers) of the first-order logic. It has logical operators (*and*, *or*), names of relations (e.g., *isIn*, *isAdjacentTo*), names of functions (e.g., *action*, *range*), and variables. A precondition formula is a description of the initial situation, and the effect formula is a description of the desired final situation.

Concrete service providers are devices, sensors, humans, and computers (servers). Each of them may provide several different services.

4 Ontology for HRC Systems

Upper ontology is a general structure of the representation of the environment of an HRC system. It is a formal and abstract description of concepts (objects) and relations between them. A concrete model of the environment of an HRC system (called a map) is an instance of the upper ontology, where the objects and the relations are specified.

The upper ontology is based on the following general concepts:

- Attributes of objects, e.g.: colour, weight, volume, position, rotation, shape, texture, etc. They are recognizable and measurable by devices and/or humans in an HRC system.
- Predefined relations on attributes.
- Relations between objects are defined based on predefined relations.
- Type of object is uniquely determined by the object construction.
- Physical type is defined only by some attributes and relations between them.
- Abstract type is defined by already defined types (physical and/or abstract), and relations between objects of these types. For example, the abstract type *Building* consists of several other abstract types like storeys, passages, rooms, stairs, lifts, etc. The internal structure of an object of type room is composed of objects of physical types such as walls, floor, ceiling, windows, and doors, as well as the relations between these objects.
- Object is an instance of its type with concrete attribute values, concrete sub-objects, and concrete relations hold between attributes, and between the sub-objects.

Upper ontology is defined as a hierarchy of types in the form of a tree. Parent type–child type relation in the tree means inheritance, i.e. child inherits from its parent. The root of the tree is *Object*. It has two main branches starting with *PhysicalObject* and *AbstractObject* from the root, see Fig. 2. These two types separate physical

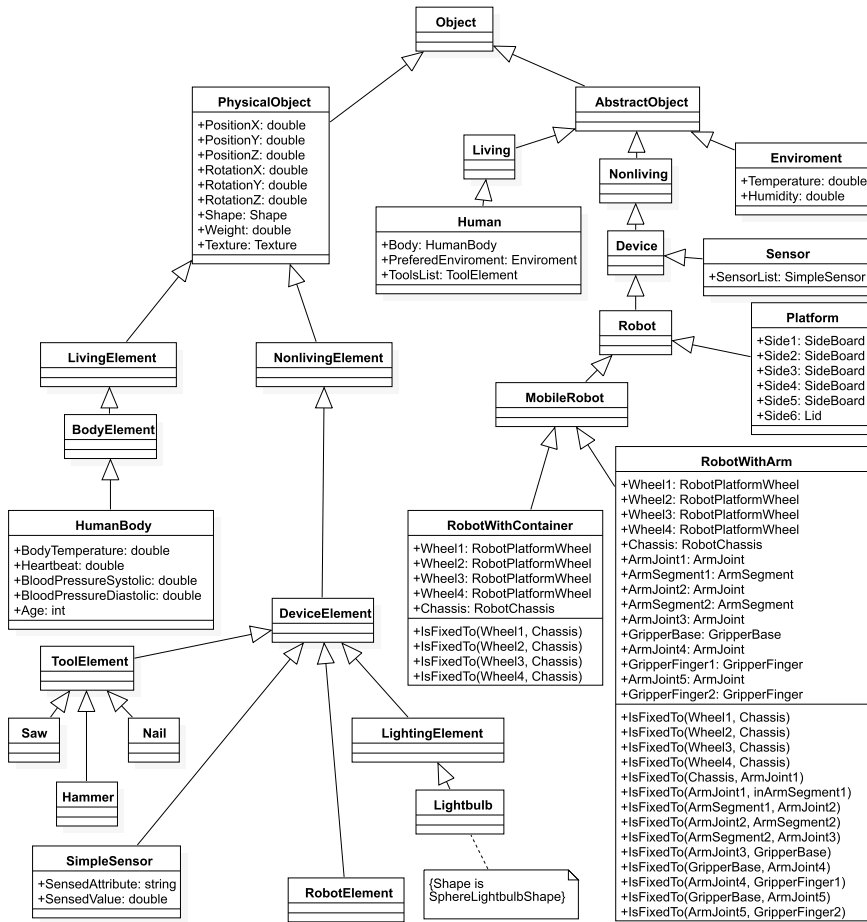


Fig. 2 A part of the upper ontology for HRC systems

objects (directly recognizable by robots and humans) from abstract objects that are hierarchically composed of physical objects, relations between them, and attributes.

The physical types are leaves of the *PhysicalObject* branch. The abstract types are leaves of the *AbstractObject* branch.

Intermediate types (nodes of the tree that are not leaves in both branches) serve for inheritance and taxonomy.

In order to add a new type to the ontology one has to specify:

- Parent type, i.e. the type that the new type inherits from.
- List of additional attributes of the new type.
- List of types of obligatory sub-objects, i.e. types of objects that are integral parts of the type being defined, e.g. legs in the case of the type of table.

- List of constraints and relations for the attributes. List of obligatory relations between sub-objects.

Let us again stress that any physical type is defined as a collection of attributes with restricted ranges and relations between them. Any abstract type is defined recursively from physical types. Hence, the attributes and predefined relations between them are the basic elements for the construction of the upper ontology.

A particular object (as an instance of its type) is defined by specifying concrete values of its attributes satisfying the relations and specifying its sub-objects (if it is of abstract type) that satisfy the relations between them.

The upper ontology is a hierarchy of types. An instance of the ontology is called a map of the environment. It is important that to support an automatic map creating and updating (by mobile robots and sensor networks, also by humans), the attributes must be recognizable and measurable by robot sensors and humans.

The original upper ontology proposed in Skarzynski et al. [4] is extended here by two branches of the hierarchy tree, see Fig. 2. The first one (called *LivingElement*) is in the *PhysicalObject* branch. The second one (called *Living*) is in the *AbstractObject* branch.

PhysicalObject type is inherited by *NonlivingElement* and *LivingElement*.

HumanBody is an elementary type that inherits from *BodyElement* type, which in turn inherits from *LivingElement* type.

One can imagine other types that inherit from *LivingElement* type. *BodyElement* type may be inherited by, for example, *DogBody* type and *HorseBody* type yet to be defined.

HumanBody type is composed of such basic attributes as body temperature, heart rate, blood pressure, etc., that must be defined first.

NonlivingElement type is inherited by *DeviceElement* that is, in turn, inherited by *RobotElement*, *ToolElement*, *SimpleSensor* and *LightingElement*.

SimpleSensor is designed to sense and store information about a specific place in the environment. It is composed of the list of pairs: *SensedAttribute* (e.g. humidity, light intensity, temperature, pollution), and its value. It is used to define abstract *Sensor* devices by aggregating individual sensors, e.g. a weather station (for measuring temperature, humidity, light intensity, etc.) to recognize the current local state of the environment.

All these physical types are components for creating abstract object types.

AbstractObject type is inherited by two types *Living* and *Nonliving* that are dedicated to service providers in HRC systems.

Human, that inherits from *Living* type, is defined by the following three attributes. *Body* value is an element of the elementary physical type *HumanBody*. *ToolList* is a list of *tools* that the *Human* can operate with. *PreferredEnvironment* is an element of abstract type *Environment* and defines non-critical conditions for the *Human*, like temperature, humidity, and radiation. Perhaps some additional attributes are needed for complex scenarios to be realized in HRC systems, and for the roles of humans in such scenarios.

Nonliving type is inherited by *Device* that is, in turn, inherited by *Robot* and *Sensor*.

Robot is inherited by *MobileRobot*, *RobotWithArm*, *Platform*, and *RobotWithContainer* that specify concrete types of robots composed of concrete physical elementary objects of type *RobotElement*.

The types shown in Fig. 2 are self-explained by their names. Actually, for the sake of presentation, it is only a small part of the proposed upper ontology for HRC systems.

The ontology is still under construction. Although the main types, physical and abstract ones, seem to have sense, a lot of research is needed to develop types that adequately represent a real environment where robots and humans can safely collaborate in accomplishing sophisticated tasks.

5 Sample Scenario

The goal of the sample scenario is to show how to automate the process of managing indoor lights: switching on lights when a person enters any of the rooms and switching off when he/she leaves the room. The simplicity of the scenario will help in explaining and understanding the system operation.

A camera was used for detecting and calculating human location in 3D space and then updating objects states in the object map of the environment.

To implement the scenario in the SO-HRCS system, it was necessary to:

1. Define the appropriate types in the ontology, and then define and add objects to the object map,
2. Define the type of service and then add the service that modifies the light intensity,
3. Calibrate the video capture software to define the translation between the position in the captured picture to the position in the object map,
4. Create a critical formula and a safe formula in Safeguards module.

(1) For the scenario, a definition of *LightBulb* type was added. It represents a light source consisting of the basic attributes inherited from *PhysicalObject*, and the new *LightIntensity* attribute. Based on the defined type, objects representing light bulbs in all the rooms can be created.

The *HumanBody* type represents a person, position in the object map and his/her identity by adding *PersonFaceId* attribute. *PersonFaceId* is used by object and face recognition software deployed on Jetson Nano to identify person and update objects positions.

(2) Based on object types, it's possible to describe a type of service that will be performed on specific physical devices. A declarative description of the service type based on the ontology is required to enable automatic discovery, arrangement and execution. For this scenario, the type of service that changes the light intensity is needed. Definitions of precondition and effect for this service type are as follows:

Precondition:

Effect: `LightBulb<LightBulb>.LightIntensity = ?LightIntensityValue
<Integer>`

The service managing lighting takes one parameter specifying the intensity of light in the form of an integer. Based on the type of service, it is possible to create a specific service in the Services Manager. Then, Service Manager registers the service in Service Registry by sending the service description including the service range - name of the room object that the controlled light bulb is in.

(3) The camera is used as a cognitive service that continuously updates the current state of the environment in its range. Correct configuration of the video capture software is necessary so it would be possible to update the position of the human object in the object map. The association between the object map and the recognized person is accomplished by a dedicated *PersonFaceId* attribute, the value of which is set accordingly when creating a *HumanBody* object, so that the system can monitor multiple people at the same time. Calibration also requires specification of the camera position (in an object map context) and coordinates of four points on an image provided by the camera, so detected human position on an image can be translated to object map position. When updating information about objects in the object map, existing relations are automatically evaluated by the Repository. This is necessary to represent the current state of the environment and is used when defining critical and safe formulas.

(4) Entering a dark room can be dangerous. This situation will be defined as a critical one. Safeguards are defined in Safeguards module by specifying the critical formula and the corresponding set of safe formulas. In this scenario, two types of critical situations are considered: when a person appears in a room and when a person leaves the room. The formulas for the first safeguard are defined as follows:

Critical formula: `human<HumanBody>IsIn room<CuboidRoom>`
 Safe formula: `room<CuboidRoom>.LightIntensity = 100`

The formula definition uses an *IsIn* relation, which allows for a simpler notation of spatial relations between objects. When a human enters a dark room:

1. Object map will be updated, and Safeguards module will detect a critical situation,
2. To achieve a safe state of the environment, Safeguards module commissions the task with the safe formula to the Task Manager which will result in turning on the light in the room.

Similarly, the formulas for turning off the light in the room can be defined:

Critical formula: `!(human<HumanBody>IsIn room<CuboidRoom>)`
 Safe formula: `room<CuboidRoom>.LightIntensity = 0`

By adding a negation of the previous critical situation, one can define the situation when a person is not in the room. Executing the safe formula as a task intention will

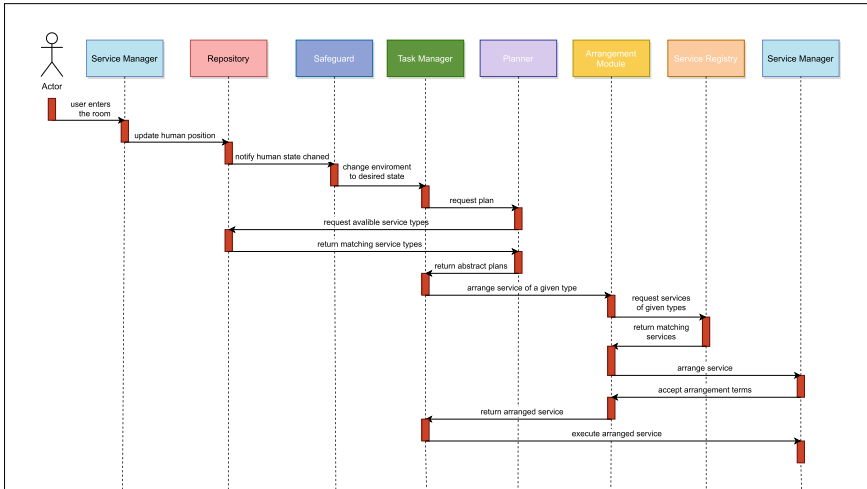


Fig. 3 Sequence diagram for the realization of the experiment scenario

result in switching off the light. As shown, the safeguards mechanism can be used not only to react to hazardous situations but also to ensure human comfort or even save energy.

The scenario can be realized after setting up the system (Fig. 3). At the start of the scenario, the light in the empty room is switched off. After entering the room, the human position in the Repository is updated by the cognitive service. The *IsIn* relation is evaluated by the Repository and the safeguards defined in the Safeguards module are checked. When critical formula of human presence in a dark room is detected, the task to switch on the light is sent to the Task Manager. Task Manager requests a potential abstract plan from the Planner module. Planner during planning phase requests required information from the Repository about the ontology (service types, object maps, etc.). After receiving the plan, Task Manager arranges discovered concrete services via the Arrangement Module and Service Registry. After arrangement services are executed via Service Manager.

6 Conclusions

This paper aimed to present our approach to solving the problem of automating the process of accomplishing tasks in heterogeneous IoT systems. The automation in the resulting system also considers declaring tasks, composing plans, arranging services, handling failures, monitoring the state of the environment and ensuring human safety. The very simple test scenario showcases the system operations and its declarative nature. That level of automation and flexibility is enabled by introducing

a common ontology. Task and service definitions, states of the environment, safety rules, and communication between system components are grounded in the ontology. Universal mechanisms that use these definitions were developed to allow the potential use of the system in diverse applications without the necessity of reimplementa-tion. This approach and methodology may be used by researchers and can be adjusted to different ontologies and their formats.

Further works will consist of more complex and diverse scenarios including human rescue missions and human-provided services. Those scenarios will require extend-ing the ontology, defining new service types and safeguards.

References

1. ISO/TC 299: ISO/TS 15066:2016. Robots and robotic devices—collaborative robots (2016). <https://www.iso.org/standard/62996.html>. Publisher: International Organisation for Standardisation
2. IEEE standard ontologies for robotics and automation (2015) IEEE Std 1872-2015:1–60. <https://doi.org/10.1109/IEEESTD.2015.7084073>
3. Isaac A (1942) Runaround. *Astounding Sci Fiction* 29(1):94–103
4. Skarzynski K, Stepniak M, Bartyna W, Ambroszkiewicz S (2018) So-mrs: a multi-robot system architecture based on the soa paradigm and ontology. In: Giuliani M, Assaf T, Giannaccini ME (eds) *Towards autonomous robotic systems*. Springer International Publishing, Cham, pp 330–342. ISBN 978-3-319-96728-8
5. Chitic S-G, Ponge J, Simonin O (2014) Are middlewares ready for multi-robots systems? In: *International conference on simulation, modeling, and programming for autonomous robots*. Springer, pp 279–290
6. Lasota PA, Fong T, Shah JA et al (2017) A survey of methods for safe human-robot interaction. *Found Trends® Robot* 5(4):261–349
7. Magrini E, Ferraguti F, Ronga AJ, Pini F, Luca AD, Leali F (2020) Human-robot coexistence and interaction in open industrial cells. *Robot Comput Integr Manuf* 61:101846
8. Honig S, Oron-Gilad T (2018) Understanding and resolving failures in human-robot interaction: literature review and model development. *Frontiers Psychol* 9:861
9. Robla-Gómez S, Becerra VM, Llata JR, Gonzalez-Sarabia E, Torre-Ferrero C, Perez-Oria F (2017) Working together: a review on safe human-robot collaboration in industrial environments. *IEEE Access* 5:26754–26773
10. Villani V, Pini F, Leali F, Secchi C (2018) Survey on human-robot collaboration in industrial settings: safety, intuitive interfaces and applications. *Mechatronics* 55:248–266
11. Fiorini SR, Bermejo-Alonso J, Gonçalves P, de Freitas EP, Alarcos AO, Olszewska JI, Prestes E, Schlenoff C, Veera Ragavan S, Redfield S et al (2017) A suite of ontologies for robotics and automation [industrial activities]. *IEEE Robot Autom Mag* 24(1):8–11
12. Sampath Kumar VR, Khamis A, Fiorini S, Carbonera JL, Alarcos AO, Habib M, Goncalves P, Li H, Olszewska JI (2019) Ontologies for industry 4.0. *Knowl Eng Rev* 34
13. McGuinness DL, van Harmelen F (2004) OWL web ontology language overview. <http://www.w3.org/TR/owl-features/>. World Wide Web Consortium (W3C) Recommendation
14. Ambroszkiewicz S (2004) Entish: a language for describing data processing in open distributed systems. *Fundamenta Informaticae* 60(1–4):41–66

Analysis, Modeling, and Forecasting of Day-Ahead Market Prices in Indian Power Exchange



Madhuri Saha and Nitai Pal

Abstract A reliable and well-understood spot electricity pricing model is desirable for minimizing risks and maximizing profits for the power trading business. Understanding the characteristics of spot market electricity prices will enhance the business confidence of the power market participants. In the Indian power market, however, spot electricity prices are highly volatile with the presence of seasonality and spikes. Such price volatility causes difficulty in predicting future energy prices. In this work, an attempt is made for analysis of day-ahead market prices in Indian Power Exchange (IEX) after visualizing its characteristics at all different time frames (i.e., yearly, monthly, weekly, daily, and hourly). The work also includes application of classical as well as neural network techniques for modeling and forecasting of the same price data to compare forecasting performances. While price data visualization using different statistical tools gives important information on price variations, modeling and forecasting price datasets show higher accuracy with neural network techniques when compared with autoregressive models. Further, neural network techniques with multivariate inputs yield better forecasting edges in comparison with the univariate inputs. The findings of the work can assist sellers/buyers to anticipate price volatility with substantial accuracy of spot electricity price forecasting in the wholesale electricity market of India.

Keywords Competitive energy market · Day-ahead market price of IEX · Price forecasting

M. Saha · N. Pal (✉)

Department of Electrical Engineering, Indian Institute of Technology (ISM), Dhanbad, Jharkhand 826004, India

e-mail: nitai@iitism.ac.in

M. Saha

e-mail: madhuri.saha@rpsg.in

1 Introduction

The Electricity Act 2003 introduced several reforms to make the Indian energy market efficient. One of such reforms was the introduction of Traders and Power Exchanges who can arrange buy or sell of electricity based on market-driven prices. Two energy exchange utilities, namely Indian Energy Exchange Limited (IEX) and Power Exchange of India Limited (PXIL), are operating since 2008 [1]. Although transaction volume of electricity through power exchanges and traders has increased at the annual growth rates of 25% and 7%, respectively, from 2009–2010 to 2018–2019 [2], price discovery in competitive market is still evolving. In 2018–2019, 53.52 BU electricity was transacted through IEX and PXIL, which is only 3.9% (approx.) of aggregate electricity generated (i.e., 1376.1 BU including RES [3]) for the same period.

Indian energy market is dominated by the long-term power purchase agreements (LT-PPA) (87% [2]) between the generation and the distribution companies (discoms). Based on such agreement, the discoms requisition power from the specified generation companies through self-scheduling on day-ahead basis to meet its expected demand. Such self-scheduling is confined within individual silos without much visibility of other economic options. Thus, the LT-PPAs lead to the possibility of scheduling costlier generations leaving behind non-utilization of the available cheaper options. On the contrary, price of electricity transacted on day-ahead basis through Power Exchanges is transparent, economic, and market driven. Therefore, to make the Indian electricity market more efficient, focus is needed to shift from long-term to short-term like Day-ahead or Real-time Market contracts [1] which can be executed through Power Exchanges. Implementation of such reforms will also enhance renewable energy penetration and grid stability. However, interests of both buyers and sellers are to be protected in such future competitive market to achieve a robust framework.

In Indian power exchanges, one of the most crucial product segments is day-ahead market (DAM) which allows trading in 15-min intervals for delivery of electricity on the next day. To address congestion in the intra-regional transmission systems across the country, DAM has thirteen pre-defined bid areas under five broadly divided regions, viz. North (N1, N2, N3), East (E1, E2), West (W1, W2, W3), South (S1, S2, S3), and North-East (A1, A2) [4]. For a particular period, the intersection point of aggregate supply and demand curves on price-quantity axes gives market clearing price (MCP) [5].

To attract investors for inducing more investment in Indian wholesale electricity market, their confidence level needs to be boosted up. Such enhanced market confidence can be achieved by correct projection of trend of spot market pricing and accurate forecasting of future prices with help of suitable models. However, modeling of electricity wholesale prices is a complex approach [5–8]. While we observe strong seasonality in yearly, monthly, weekly, daily, and intra-day spot market price variation [9], we also observe high price volatility (13.14% for 2018–2019 in IEX [2]).

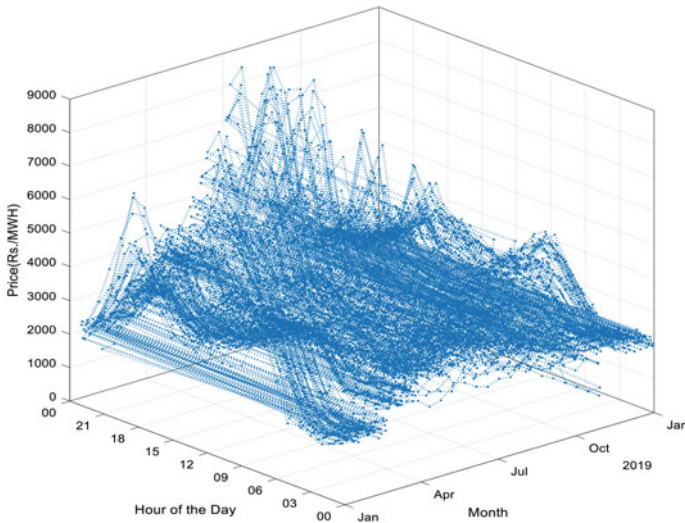


Fig. 1 Variation of electricity spot market prices of IEX in 2019

Such price volatility arises out of various driving forces like coal price variation, variable renewable energy generation including hydro-power, transmission congestion, irregular load, market restrictions, growth in transmission and distribution infrastructure, change in trading margin, government taxes, and other relevant taxes, etc [10]. Figure 1 shows a 3-D image of hourly and monthly variations of MCP collected in 2019 for DAM of IEX. The variation clearly shows seasonality, volatility, and spikes.

Although limited studies [1, 5, 9, 11–16] are conducted on spot price forecasting in deregulated Indian power market, authors used either single or hybrid time series modeling techniques, e.g., stochastic, artificial intelligence, causal/regression, etc. We propose a novel approach to

- i. visualize and analyze the Indian electricity spot price historical time series data at different time frames
- ii. forecasting price data using both ARIMA and neural network techniques and comparing the accuracy.

The work aims to pursue the following objectives:

- A. Study of Historical Time Series Price Data of IEX through data visualization with different
 - a. Period: Daily, Weekly, Monthly, Yearly, Five Years
 - b. Bid Areas: Thirteen areas across the country and MCP
 - c. Time Zones: Average (Round the clock, RTC), Peak and Non-Peak
 - d. Time of Day: Morning, Day, and Night.
- B. Analyzing the trend, seasonality and identification of plausible reasons behind price variations.

- C. Modeling and Forecasting of Price data using
 - a. ARIMA and SARIMA
 - b. Neural Network architecture.
- D. Measuring and comparing accuracy in forecasting models.

In this work, MATLAB (version 2018a) is used. The computations are carried out with the use of MATLAB Apps, viz. Econometric, Neural Net Fitting, and Neural Net Time Series. The rest of the work is structured as follows. Section 2 discusses data and methodologies used. Section 3 describes results and discussion. Section 4 represents the conclusion.

2 Data and Methodologies

2.1 Data

Datasets. The datasets are accessed from different publicly available sources. The first dataset, the spot electricity prices of DAM in IEX, is sourced from the IEX website [4]. The data in INR/MWh collected for five complete years from January 2015 to December 2019 represent:

- i. Hourly MCP and prices of thirteen bidding areas
- ii. Daily MCP—Average, Peak, and Non-Peak
- iii. Daily MCP—Morning, Day and Night.

Although there are multiple factors which actually causes variations in the spot market electricity prices as mentioned above, we chose three such representative datasets in the work related to weather, economy, and power and energy [15]. The second dataset is all India mean maximum and mean minimum temperature at monthly resolution collected [17] for January 2015 to December 2017. The third dataset represents economic data of annual and quarterly Gross Domestic Product estimates and is sourced [18] for January 2015 to December 2017. The fourth dataset archives [19] reports on hydro, fuel (coal consumption and stock), and power supply (peak demand and energy) in monthly resolution for the Northern Region and all India from January 2015 to December 2019.

Selection of Period. The choice of time period of spot electricity price data of IEX for five years from 2015 to 2019 lies in the rationale of capturing the steady annual growth of electricity volume transacted through power exchanges [2]. Throughout this period, the yearly volume of electricity transacted through power exchanges surpassed the same transacted through traders [2]. Such trend demonstrated enhanced confidence of power market participants on transactions through power exchanges over other trading platforms. Due to the nationwide pandemic situation, we restrict our study on time series spot electricity price data till FY 2019–2020.

2.2 Methodology

The work follows three steps. In the first step, price data are visualized and analyzed for specific events. The second step revolves around model selection for forecasting using techniques of ARIMA and neural networks. The third and final step is price forecasting and measuring of forecasting accuracy. While DAM prices from the first data source are used in all of the above steps, data from the second to fourth sources are used in the second and third steps as external inputs to build neural network models and corresponding forecasting.

Data Visualization and Analysis. The time series price characteristics are visualized at all different time frames (Ref. 1.A) using all three data types (Ref. 2.1) by drawing line diagrams, trend lines, histograms, and box plots. In the work, we observe wave-forms covering different timelines for particular data types. Figure 2 shows the time series of hourly MCP for the entire span of five years from 01.01.2015 to 31.12.2019 having a total of 43,824 data count.

On visualization, the price line diagram shows seasonality and spikes. There is a considerable variation between minimum and maximum price, viz. INR/MWh 524.88 and 16,910.00, respectively, clocked on July 3, 2017, at 05:00 h and on October 1, 2018, at 19:00 h.

Figure 3 represents the hourly price variation of all bid the areas and MCP using box plots for the aforesaid five-year time frame. The plot shows huge price variations within individual bid areas, across the bid areas, and MCP.

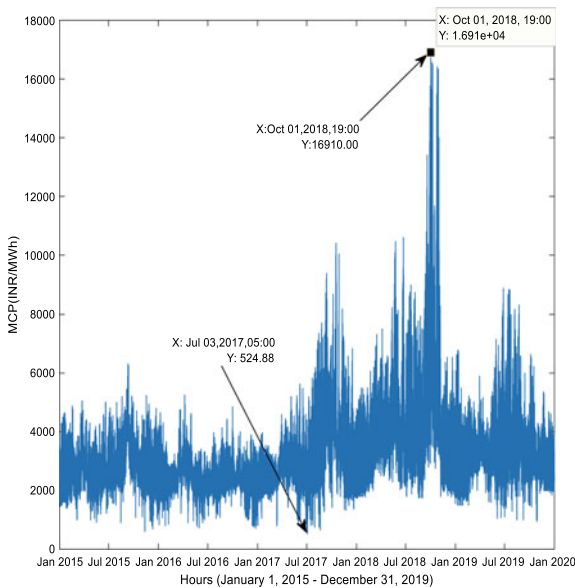


Fig. 2 Variation of hourly MCP: January 1, 2015–December 31, 2019

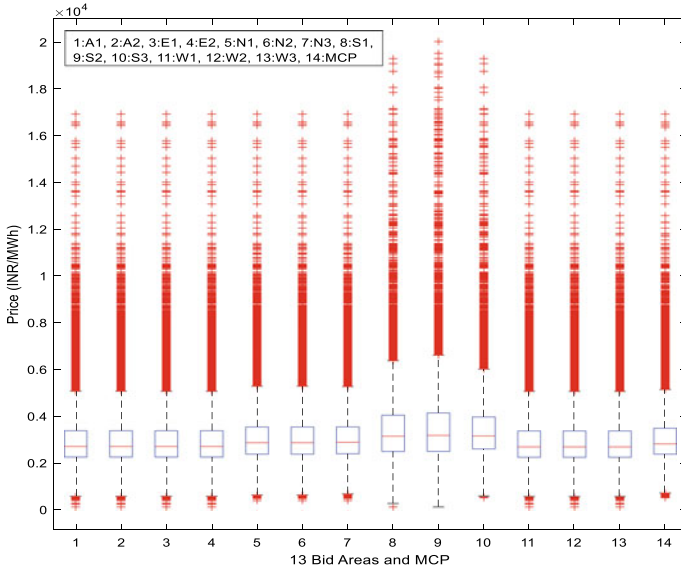


Fig. 3 Box plots: price variations of bid areas and MCP

Figure 4a, b exhibit change in daily MCP at different time slots, e.g., RTC Average, Peak, and Non-Peak [5, 20] and at different times of the day, viz. Day, Night, and Morning with significant ups and downs over January 01, 2017, to December 31,

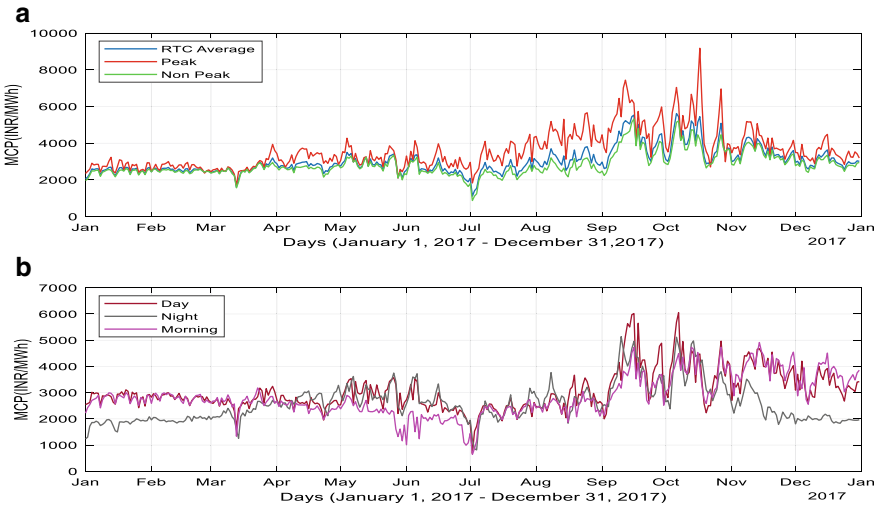


Fig. 4 Daily MCP variation. a upper panel—average, peak, and non-peak. b lower panel—day, night, and morning

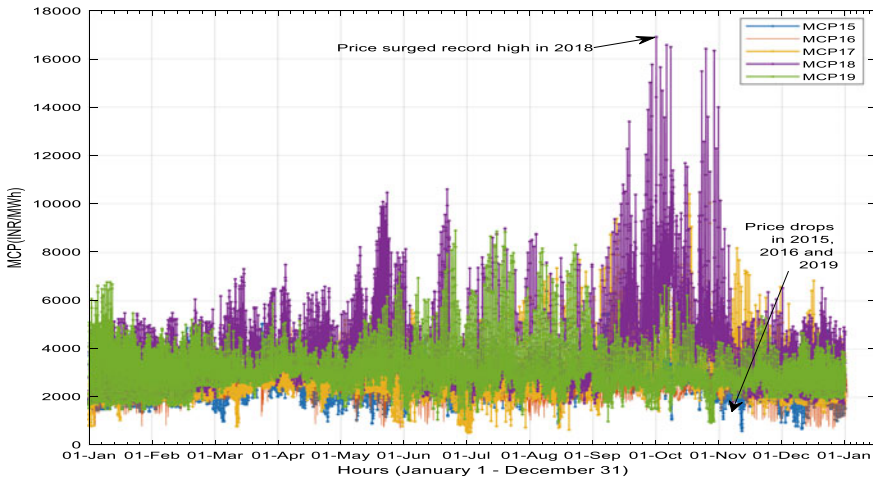


Fig. 5 Year-on-year comparison of price variation of hourly MCP

2017. The time series also shows huge spikes and significant seasonality during the year 2017.

Figure 5 compares year-on-year price variation of hourly MCP for the five calendar years from 2015 to 2019. Five time series plots indicate yearly surge and drop of spot market electricity prices.

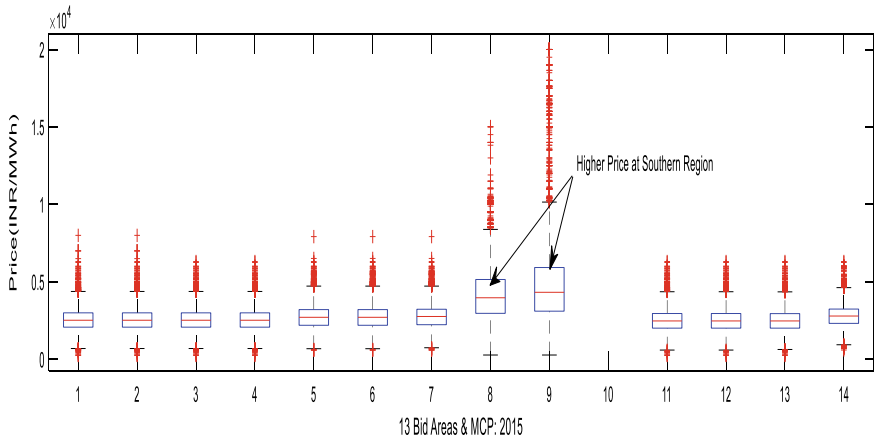
Figure 6a, b visualizes of price variation of 13 bid areas and MCP using box plots and line diagrams for the year 2015.

Figure 7a, b demonstrates price variation of 13 bid areas and MCP using box plots and line diagrams for the year 2019. Figures 6a, b and 7a, b clearly show that price differences among bid areas and MCP are narrowing down from 2015 to 2019, leading to the desired pricing strategy of ‘one nation one price’.

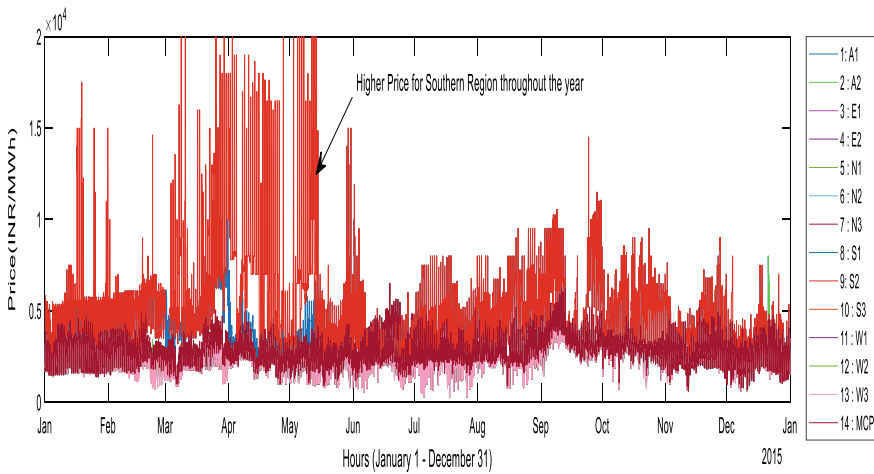
In monthly analysis, Fig. 8 exhibits hourly price variations of MCP for December 2019 with price drops in all five weekends [7]. The surge occurs on weekdays for 2 (two) hour lines at 7 pm and 9 am.

The same data of December 2019 when plotted (Ref. Fig. 9a, b) for daily price variation clearly show the lowest price on Day 1 (Sunday) and the highest price on Day 10 (Tuesday). The Contour plot clearly shows variations in price at all different Times of the Day (hour) and Day of the Month (date) in December 2019.

In daily analysis, Fig. 10 reflects price variations on a typical day for its 24 h. The daily curve shows different prices at different hours of the day. The curve indicates night lean, morning peak, day lean, and evening peak prices. Interestingly, the morning peak price is 118% (approx.) more than the evening peak price and, also, the day lean price is at a 193% (approx.) higher value than the same at night lean period.



(a)



(b)

Fig. 6 Hourly price variation of MCP and Bid areas in 2015. **a** upper panel—box plot. **b** lower panel—line diagram

Modeling and Forecasting. For modeling and forecasting, we apply both statistics and artificial intelligence using the stochastic and artificial neural network [21] models.

We study conditional mean models (ARIMA and SARIMA) of linear regression for univariate time series data. The non-seasonal ARIMA (Autoregressive Integrated Moving Average) model [16, 23] integrates both the Autoregressive (AR) and Moving Average (MA) processes.

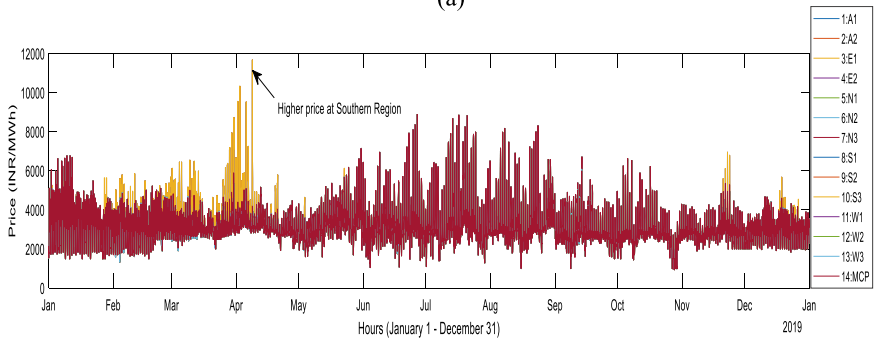
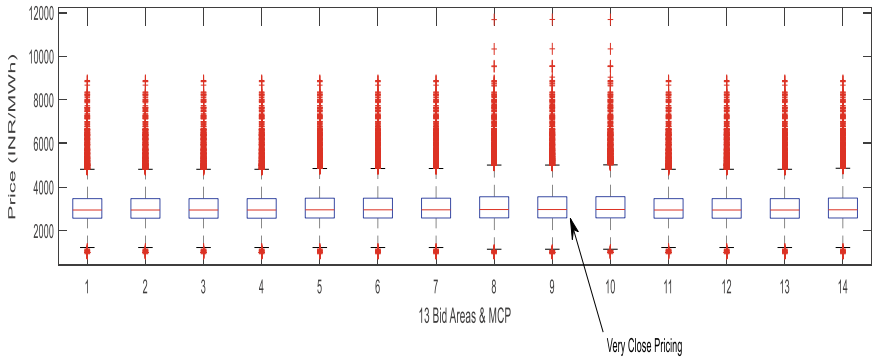
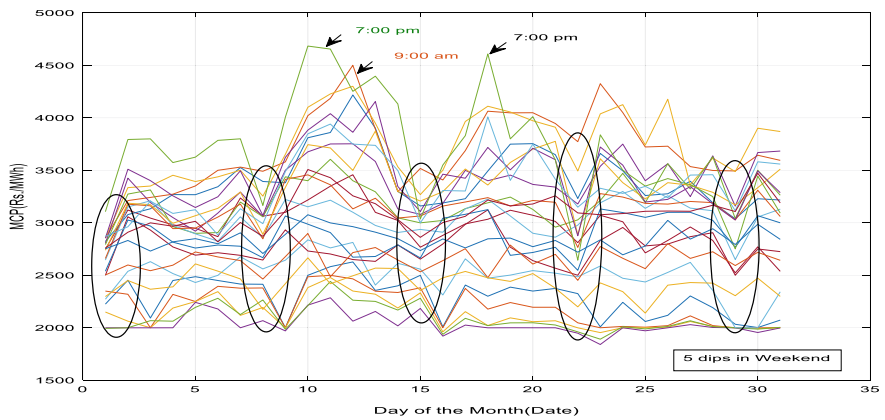
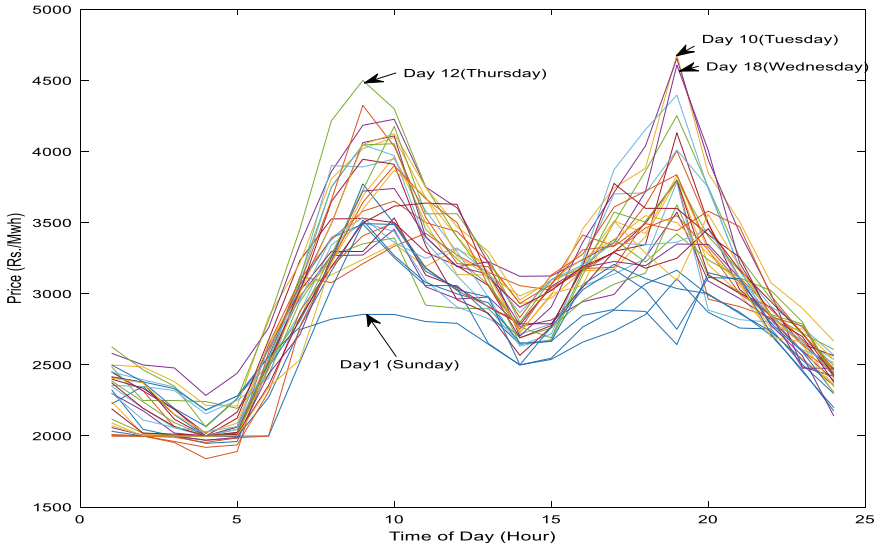
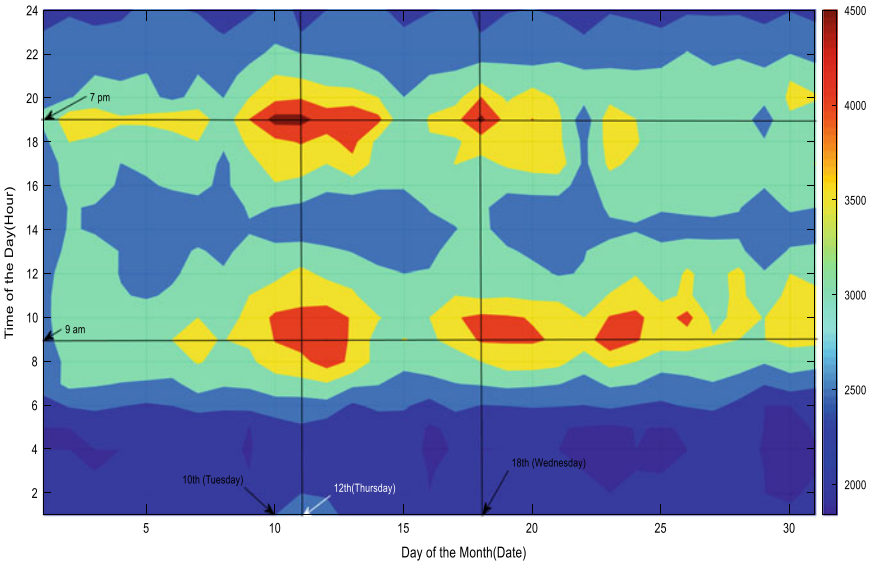


Fig. 7 Hourly price variation of MCP and bid areas in 2019. **a** upper panel—box plot. **b** Lower panel—line diagram





(a)



(b)

Fig. 9 Price variation of daily MCP in December 2019. **a** upper panel—line diagram. **b** lower panel—contour plot

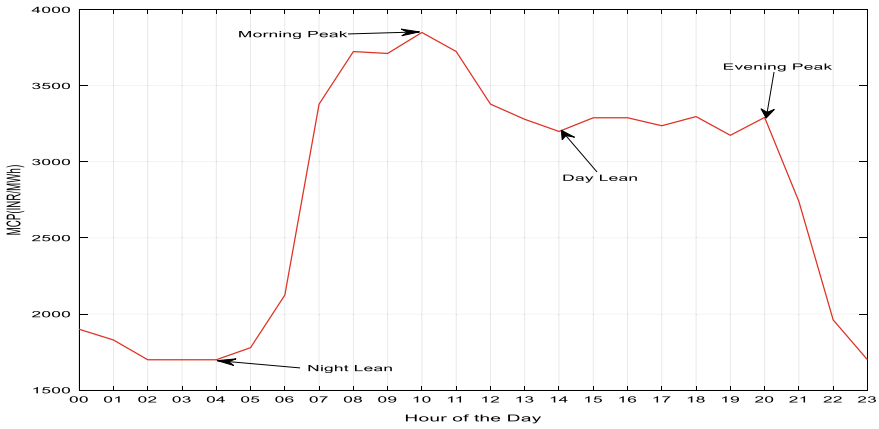


Fig. 10 A typical daily spot market price variation curve

The AR model which considers the dependency of the data on its lagged value (p) is shown in Eq. (1).

$$y_t = c + \sum_{i=1}^p \varphi_i y_{t-i} + \varepsilon_t \tag{1}$$

where y_t is the time series data; c is the constant; φ_i is AR coefficients at lag, $i = 1, 2, \dots, p$; and ε_t is the error coefficient known as Gaussian white noise series with mean = 0 and variance = σ_ε^2 .

Similarly, the MA model which uses the dependency of the data on residual error of moving average with lagged value (q) is shown in Eq. (2).

$$y_t = \mu + \sum_{i=0}^q \theta_i \varepsilon_{t-i} \tag{2}$$

where μ is the expected value of y_t with usual value = 0; θ_i is the weights of the present and future observations with $\theta_0 = 1$ and $i = 1, 2, \dots, q$.

The above AR and MA models can be combined together to represent ARIMA model of the order (p, d, q) where ‘ p ’, ‘ d ’, and ‘ q ’ denote non-seasonal autoregressive order, differentiating count to convert the series data stationary and forecast error order, respectively. The ARIMA model is shown in Eq. (3).

$$y_t = c + \sum_{i=1}^p \varphi_i y_{t-i} + \varepsilon_t + \sum_{i=0}^q \theta_i \varepsilon_{t-i} \tag{3}$$

where φ_i, θ_i is not equal to zero, and $\sigma_\varepsilon^2 > 0$.

The ARIMA model, combined with the seasonal components, gives the Seasonal ARIMA or SARIMA model. In general, SARIMA is represented by ARIMA $(p, d, q) \times (P, D, Q)_S$ where P, D , and Q are all seasonal AR order, differencing term, and MA order. S is the time period of recurrence of the seasonal pattern.

In the study, we follow an eight-step process:

- In step 1, datasets are imported for training and testing. The training dataset is used for analysis and preparing model. Testing dataset is used to measure accuracy and back-testing the selected model.
- In step 2, exploratory data analysis is performed. Dataset is transformed to stationary for modeling. Augmented Dickey–Fuller (ADF) test is carried out to verify stationarity.
- In step 3, the pattern of auto-correlation and partial auto-correlation Factors (ACF & PACF) is identified to determine the applicability of AR and MA Model and identify AR and MA Lag(s).
- Step 4 is about creating ARIMA and SARIMA models using econometric modeler app in MATLAB.
- Step 5 is on determining the best-fit model based on lower values of Akaike information criterion (AIC) and Bayesian information criterion (BIC).
- In Step 6 and Step 7, back-testing is performed with the forecast model using the test dataset, and the accuracy of the model is measured by calculating the error of forecast vs actual.
- Finally in Step 8, Monte Carlo Simulation is performed to understand the risks and uncertainty of forecasting.

For the first set of models, the hourly MCP price dataset for 2019 with the 8762 data count is chosen as training dataset. The hourly MCP for the first week of January 2020 with 169 data count is considered as testing dataset. Four models are developed, and details of the models with AIC and BIC values and forecasting accuracies are given in Table 1. The graphs of the actual vs forecasts for the three best-fit models are shown in Fig. 11a, b.

For the second set of models, the hourly MCP dataset for five years (i.e., 01.01.2015 to 31.12.2019) is considered training data with a 43,825 data count. The testing dataset is considered hourly MCP for three months, viz. January 2020 to March 2020 with a 2185 data count. The details of the three SARIMA models developed are given in Table 2. The graph of time series data with actual vs forecasts is exhibited in Fig. 12.

ANN exhibits improved forecasting performance compared to other standard models, especially for price time series with features of a complex and chaos model [24, 25]. Therefore, in the work, the prices under DAM are further forecasted using ANN-based models having inputs with the same historical data as used in ARIMA models and related exogenous variables [26]. The three different neural network concepts, viz. Non-linear Autoregressive Network (NAR), Non-linear Autoregressive network with external inputs (NARX), and Neural Net Fitting (NNF), are applied for modeling and forecasting of spot market electricity prices.

Table 1 Hourly MCP for the year 2019: Goodness of Fit measures

SI	Model Name	AIC	BIC	MAE	MAPE
1	ARIMA (1,1,1)	-1.1783e + 04	-1.1755e + 04	606.78	24.15
2	SARIMA (1,1,1) with seasonality 24	-1.6623e + 04	-1.6595e + 04	278.70	9.67
3	SARIMA (3,1,3) with seasonality 24 and constant	-1.7407e + 04	-1.7351e + 04	291.23	10.32
4	SRIMA (3,1,3) with seasonality and w/o constant	-2.0560e + 04	-2.0496e + 04	307.82	11.17

Note Performance of the three best-fit models (SI. 2, 3, and 4) is shown in Fig. 11 as Forecast 1, Forecast 2, and Forecast 3, respectively. The best performing model was chosen as SARIMA (1,1,1) with seasonality 24 (SI. 2) having the least value of MAE and MAPE

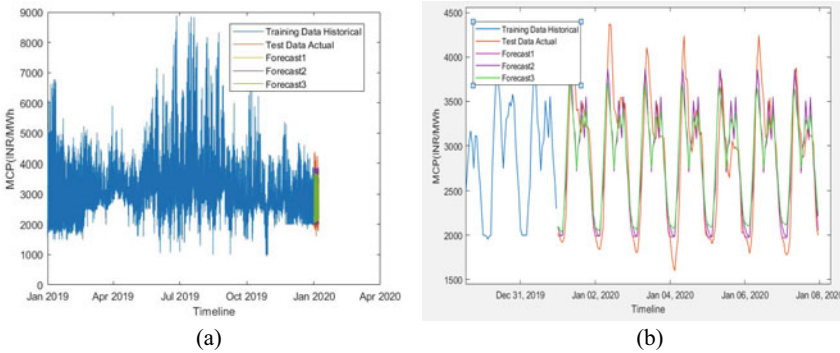


Fig. 11 Hourly MCP-2019. **a** left panel—best-fit SARIMA models with forecast. **b** Right panel—close view of actual vs forecast

Table 2 Hourly MCP for five years: goodness of fit measures

SI	Model Name	AIC	BIC	MAE	MAPE
1	SARIMA (1,1,1) with seasonality 24	-7.5837e + 04	-7.5803e + 04	359.73	14.58
2	SARIMA (3,1,3) with seasonality 24 and Constant	-7.8862e + 04	-7.8792e + 04	344.94	13.90
3	SRIMA (3,1,3) with seasonality and lag (1,1)	-9.5401e + 04	-9.5314e + 04	350.26	14.08

Note Performance of the above three best-fit models (SI. 1, 2, and 3) is shown in Fig. 12 as Forecast 1, Forecast 2, and Forecast 3, respectively. Best performing model is chosen as SARIMA (3,1,3) with seasonality 24 (SI. 2) and constant having least value of MAE and MAPE

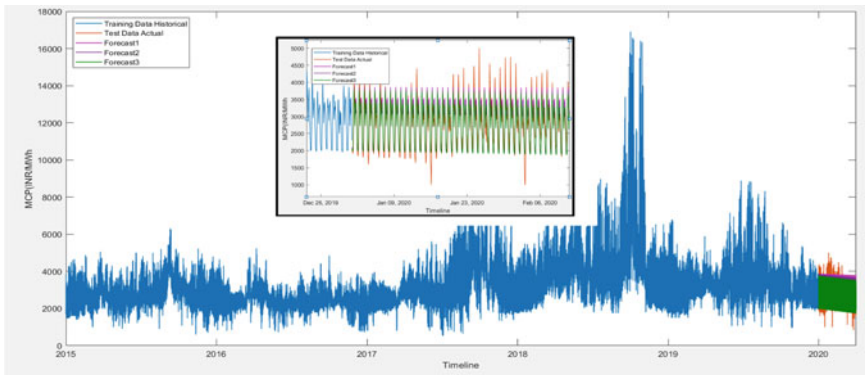


Fig. 12 Hourly MCP (five years): SARIMA models with forecasts along with close view of actual vs forecast in inset

Under the NAR concept [27], the training of network starts from the open loop condition with the response of the actual targets. The network is transformed to a closed loop after training with the response of forecasted values. The model’s general forecasting equation is demonstrated in Eq. (4).

$$y(t) = f(y_{t-1}, y_{t-2}, \dots, y_{t-d}) + \varepsilon_t \tag{4}$$

where the function $f(\cdot)$ of the time series $y(t)$ remains unknown before training. Such training aims at approximating the function through optimization of weight (w) and bias (b) with d as the number of delays. NAR model is demonstrated in Eq. (5).

$$y(t) = \alpha_0 + \sum_{j=1}^k \alpha_j \varphi \left(\sum_{i=1}^{\gamma} \beta_{ij} y_{t-i} + \beta_{0j} \right) + \varepsilon_t \tag{5}$$

where α_0 & β_{0j} represent the constants corresponding to the output unit and hidden unit j , respectively; k represents count of hidden layers; α_j represents the weight between hidden unit j and output unit; φ represents the activation function; γ represents count of entries; β_{ij} represents the term corresponding to the weight of input unit i and hidden unit j and ε_t is the error term.

For price forecasting using the NAR model, four-step process is followed. In step 1, the dataset is imported and processed for training and testing. The training dataset is used for analysis and preparing model. The testing dataset is used to measure accuracy and back-testing the selected model. In step 2, the best network architecture is selected after trial and error by choosing layers, hidden neurons, delay values, and regression values, the dataset is trained with Bayesian Regularization, and results are exported. In step 3, back-testing with a forecast model is performed with test data.

Finally, in step 4, the accuracy of the model is measured by calculating the error between forecast and actuals.

The first NAR model is chosen with a daily time series peak price dataset for five years (01.01.2015–31.12.2019) with an 1826 data count. The total dataset is divided into three components (default): Training-1278 (70% of total), Validation-274 (15% of total), and Testing-274 (15% of total). The network outline and actual vs forecast pricing with a close view are shown in Figs. 13 and 14a, b.

For the second NAR model, hourly price data of bid area ‘N2’ are considered for 2019 (01.01.2019–31.12.2019) with a data count 8760. Like the first NAR model, training, validation, and testing datasets are distributed in the same ratio for developing models. The actual vs forecast pricing, regression plots, and close view are shown in Fig. 15a–c.

For both of the above NAR models, the optimum number of hidden layers and delay counts have experimented on a trial-and-error basis. The models having the highest ‘R’ values are selected over other experimental models. The results using Bayesian optimization for the experiments are shown in Tables 3 and 4.

NARX (Non-linear Autoregressive with External inputs) is a dynamic recurrent and powerful neural network model with quicker convergence efficiency [30]. NARX

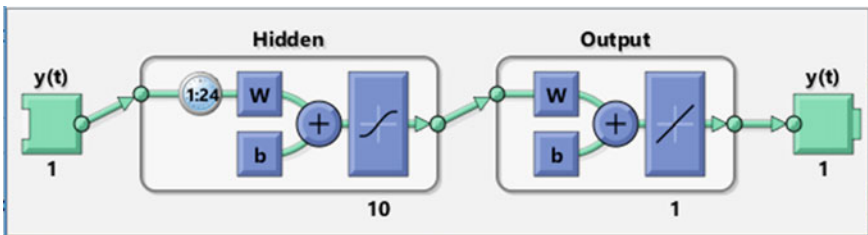


Fig. 13 NAR network configuration: daily peak data (5 years)

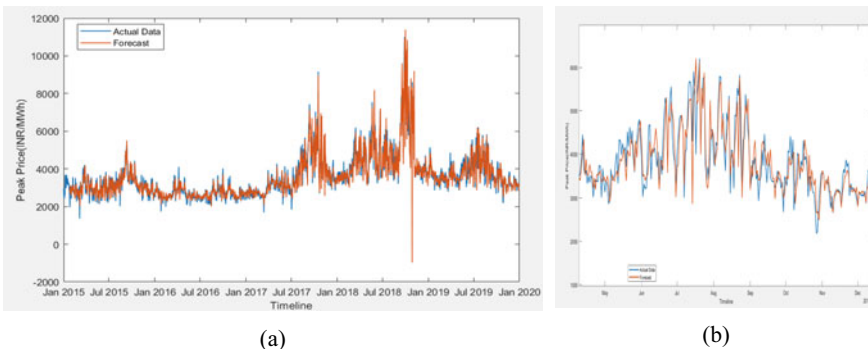
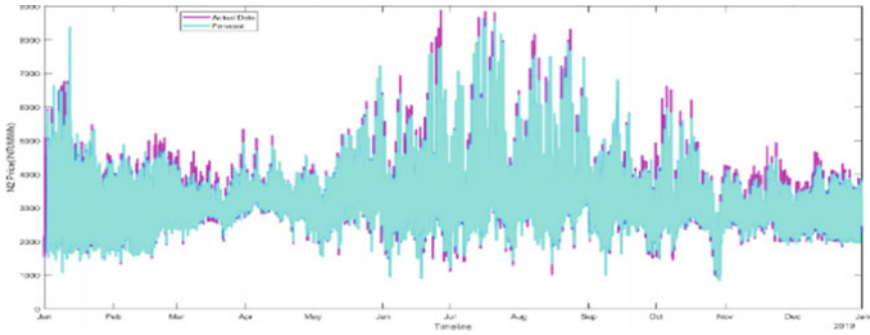
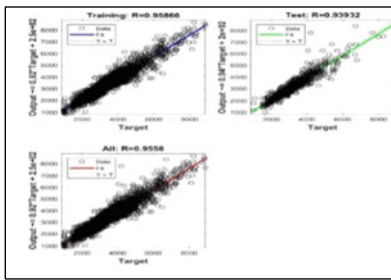


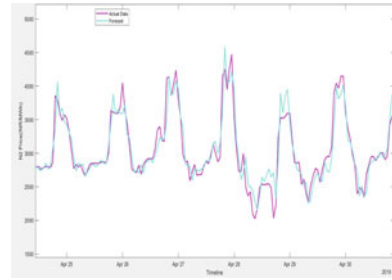
Fig. 14 Daily peak price (Five years): a Left panel: actual vs NAR forecast. b Right panel: close view of actual vs forecast



(a)



(b)



(c)

Fig. 15 N2 pricing. (a) Upper panel: actual vs NAR forecast. (b) Lower panel: left-regression plots. (c) Lower panel: right-close view of actual vs forecast

Table 3 NAR models: Daily peak price data for five years

Model no.	Hidden layer neuron	Delay	R-training	R-testing
1	10	2	8.97909e-1	7.30344e-1
2	5	5	9.05864e-1	8.20921e-1
3	10	24	9.68530e-1	8.25455e-1

Note Model no. 3 is selected based on the highest ‘R’ values among the other experimented models

Table 4 NAR models: hourly DAM price data of N2 bid area for one year

Model no.	Hidden layer neuron	Delay	R-Training	R-Testing
1	10	2	9.03173e-1	9.02399e-1
2	5	5	9.07683e-1	8.92831e-1
3	10	24	9.58657e-1	9.39319e-1

Note Model no. 3 is selected based on the highest ‘R’ values among the other experimented models

model considers the past values of the time series and the exogenous variables corresponding to past values of the same time series. The model’s mathematical expression can be demonstrated in Eq. (6).

$$y(t) = f(y(t - 1), y(t - 2), \dots, y(t - n_y), x(t - 1), x(t - 2), \dots, x(t - n_x)) \tag{6}$$

where $f(\cdot)$ is the non-linear multilayer perceptron; $y(t)$ is the time series and $x(t)$ is the exogenous input at time step t respective; n_y and n_x are the numbers of delays for output and input, respectively.

Spot market electricity prices depend on many external inputs (exogenous) like weather factors, demand and supply status of electricity, coal stock position, supply of renewables [22], economic factors of the country, and so on [10]. In developing the NARX model, the target dataset is considered hourly price data of the N2 bid area for the year 2017 (08.01.2017 to 31.12.2017) with a total of 8593 data counts. Thirteen input datasets for same period is considered as follows: northern region-energy (MU) not supplied, peak demand (MW) not met, hydro generation (GWh), coal consumption and stock (kton), all India-quarterly GDP data, monthly average maximum and minimum temperature, bid area N2-hourly price of previous week same hour, previous day same hour, previous 24-h average, and working day and weekday indication. The results of the models with similar datasets as earlier are shown in Table 5. The details of model, selected on the most optimized ‘R’ values, are shown in Fig. 16, 17a, b.

Table 5 NARX models: hourly DAM price data of N2 bid area for one year

Model no.	Hidden layer neuron	Delay	R-training	R-testing
1	10	2	9.14331e-1	8.93410e-1
2	5	2	9.06280e-1	8.94621e-1
3	5	24	9.59279e-1	9.17891e-1

Note Model no. 3 is selected based on the highest ‘R’ values among the other experimented models

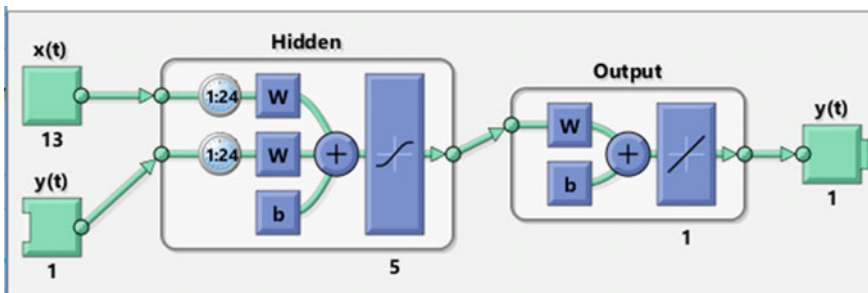


Fig. 16 NARX network configuration: hourly N2 data (1 year)

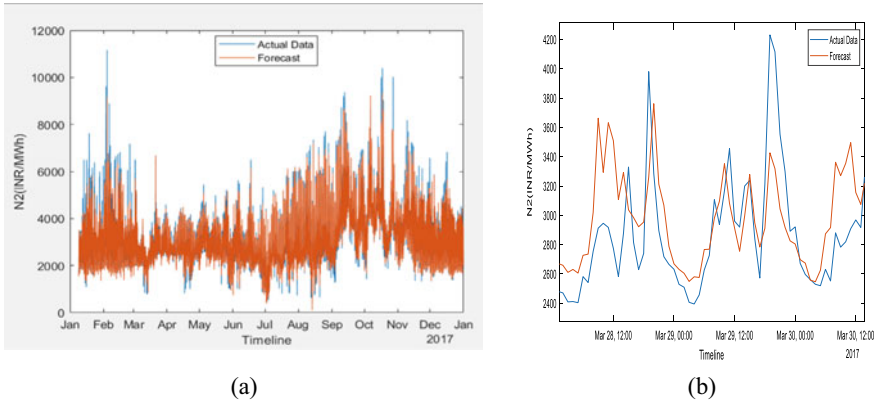


Fig. 17 Hourly N2 data of 2017. **a** Left panel—actual vs NARX forecast. **b** right panel—close view of actual vs forecast

Neural Net Fitting (NNF) is a two-layer feed-forward neural network with a hidden layer and output layer as a linear function. To achieve superiority over the ‘univariate approach’, NNF considers the ‘multivariate approach’ [28] for prediction. The model’s mathematical expression is demonstrated in Eqs. (7) and (8).

$$Z = \alpha(x) \triangleq f(WX + b) \tag{7}$$

where α and $f(\cdot)$ represent non-linear transformation function and activation function (e.g., sigmoid function), respectively; W represents parameter matrix indicating weight of individual inputs and b represents the biased element. The output, $y(t)$ is expressed as

$$y(t) = \beta(Z) = \tilde{f}(\tilde{W}Z + \tilde{b}) \tag{8}$$

where $y(t)$ is the output; β and $\tilde{f}(\cdot)$ represent non-linear transformation function and decoded activation function; \tilde{W} and \tilde{b} represent decoded weight matrix and bias element.

Our first NNF model is developed with the same price time series of hourly N2 dataset and thirteen external inputs as considered in the above NARX model. The NNF structure is made of thirteen inputs, a hidden layer with twenty-four neurons and sigmoid function, and the output layer with single-neuron and identity activation function. The results are given in Table 6.

The model, selected on the most optimized ‘ R ’ values, is shown in Fig. 18a–c for the corresponding network diagram, Actuals vs Forecast data, and its close view.

The second NNF model is developed using the hourly price dataset of MCP for the year 2017 with a data count of 8593. Eight exogenous variables, i.e., all India-energy

Table 6 NNF Models: Hourly MCP for one year

Model no. Model no.	Hidden layer neuron	R-training	R-testing
1	10	9.02147e-1	8.97634e-1
2	8	9.01543e-1	9.04732e-1
3	5	9.03994e-1	9.12896e-1

Note Model no. 3 is selected based on the highest 'R' values among the other experimented models

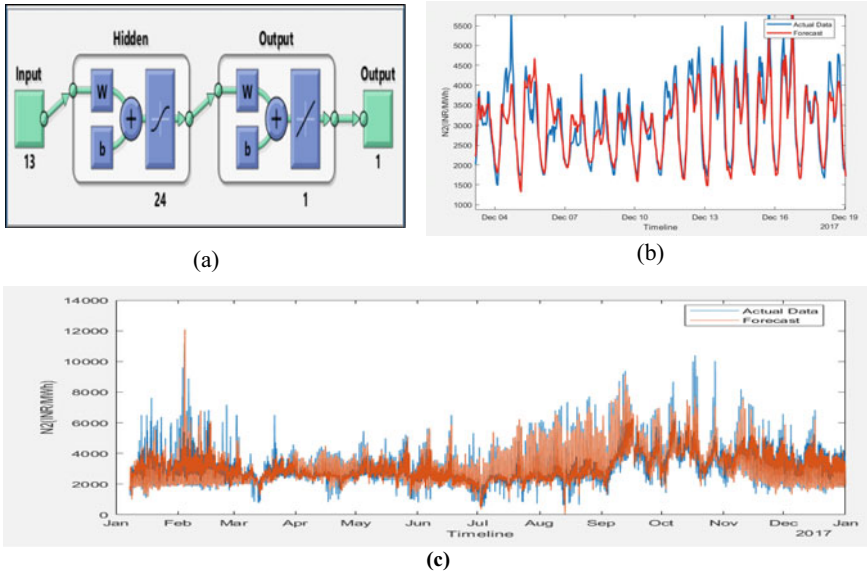


Fig. 18 N2 pricing of 2017. **a** upper panel: left-NNF configuration. **b** upper panel: right-close view of actual vs forecast. **c** lower panel: actual vs NNF forecast

not supplied (MU), peak demand not met (MW), GDP data, Maximum and Minimum temperature (degree Celsius), price of week same hour, previous day same hour, and the average price of previous 24-h, are considered as input datasets. The total dataset is divided into three components, viz. training (70%), validation (15%), and testing (15%). The optimum number of hidden layers has experimented on trial and error. The best three models using Bayesian Regularization are exhibited in Table 7. The network configuration and the graph showing Actuals vs Forecast data are shown in Fig. 19 and 20a, b.

Table 7 NNF models: hourly DAM price data of N2 bid area for one year

Model no.	Hidden layer neuron	R-training	R-testing
1	10	8.88149e-1	8.69807e-1
2	8	8.85765e-1	8.61483e-1
3	5	8.75463e-1	8.75136e-1
4	20	9.01970e-1	8.71001e-1
5	24 (3rd time retrained)	9.04436e-1	8.61845e-1

Note Model no. 5 is selected based on the highest ‘R’ values among the other experimented models

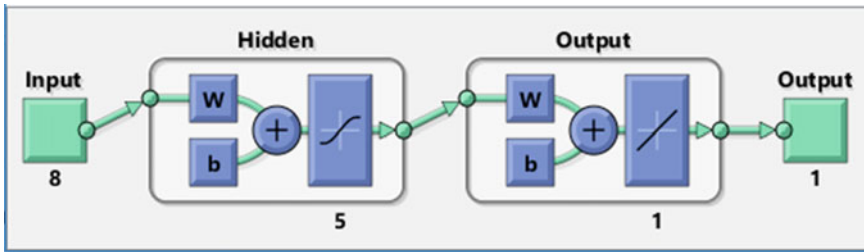


Fig. 19 NNF network configuration: Hourly MCP data (1 year)

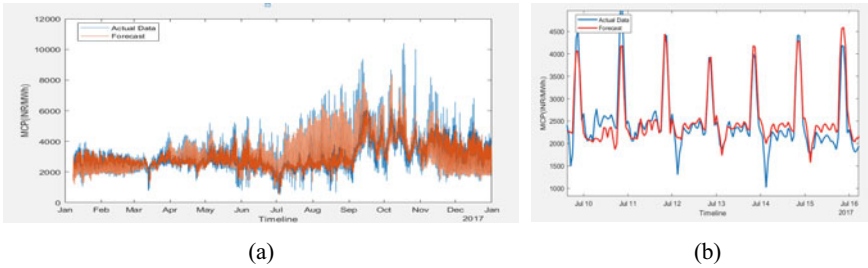


Fig. 20 Hourly MCP of 2017. **a** right: actual vs NNF forecast **b** left: close view

3 Results and Discussions

The performances of the individual categories of models are demonstrated in Tables 1–7 for different price datasets. Table 8 shows the comparison of forecasting performances among all the best-fit models under individual categories.

Table 8 Performance comparison of forecasting errors

Ref	Model name	Dataset	MAE	MAPE
Table 1	SARIMA	Hourly MCP for one year	278.70	9.67
Table 2	SARIMA	Hourly MCP for five years	344.94	13.90
Table 3	NAR	Daily Peak Price for five years	246.30	6.99
Table 4	NAR	Hourly Price of N2 bid area for one year	178.29	5.74
Table 5	NARX	Hourly Price of N2 bid area for one year	218.17	7.31
Table 6	NNF	Hourly MCP for one year	298.11	10.18
Table 7	NNF	Hourly Price of N2 bid area for one year	316.14	10.50

Note For the similar model, MAPE indices are reduced for a shorter period for the same price dataset. Also, MAPE indices are better for the same dataset with univariate input when compared to the same with multivariate inputs. Forecasting performances of ANN are generally better than the same compared to the stochastic models

3.1 Performance Indices

The forecasting accuracies of all the models mentioned above are computed using two types of indices, viz. Absolute Error, i.e., Mean Absolute Error (MAE) [9, 11], and Percentage Error, i.e., Mean Absolute Percentage Error (MAPE) [24, 25]. The Absolute Error can be defined in Eq. (9).

$$e_t = (y_t - y'_t) \tag{9}$$

where y_t and y'_t are observed value and predicted value at time t and n is the total number of data points. Hence, the mathematical expression of MAE [26, 29] is written in Eq. (10).

$$MAE = \text{Mean}_{i=1,n} |e_i| \tag{10}$$

The Percentage Error can be defined in Eq. (11).

$$p_t = [(y_t - y'_t) / y_t] * 100\% \tag{11}$$

Hence, the mathematical expression of MAPE [26] can be written in Eq. (12).

$$MAPE = \text{Mean}_{i=1,n} |p_i| \tag{12}$$

3.2 Discussions

While both the SARIMA models demonstrate a considerably good accuracy percentage for price forecasting of MCP over two different time frames, viz. one and five years, obviously SARIMA for shorter duration gives better accuracy. For enhancing forecasting accuracy, neural network models are applied. NAR models using daily peak price data for five years and hourly price data of N2 bid area for one year give accuracy to the tune of 94%. However, NARX and NNF models are developed with exogenous inputs to provide a further edge to forecasting. Such NARX and NNF models for hourly MCP and N2 price data of one year with inputs of weather, economy, power and energy of the corresponding period give lower forecasting accuracy than the same for univariate models. One of the causes could be the non-usage of weather and economic data in high resolution due to the apparent non-availability of such data from authentic sources free of cost. Had the weather data been daily/hourly, the forecasting performance could have been better than the calculated monthly values applied to NARX and NNF models. Further, the contribution of other major factors on DAM price variation, e.g., growth of transmission and distribution infrastructure, renewable energy generation, increase of transmission charges, trading margin, government and other related taxes, etc., could have been considered as additional exogenous inputs to improve forecasting accuracy.

During visualization of price data, it is observed that higher pricing of southern region during the year 2015 has come down gradually. In the year 2019 (after addition of new bid Area, S3 in 2017), it is almost at par with mean MCP covering all thirteen bidding areas across India.

Forecasting performances are further studied by breaking down the errors in periodicity of the month, week, and hour. For illustration, the result is shown in Fig. 21a–c on the NARX model using hourly price dataset for bid area N2 for the year 2017.

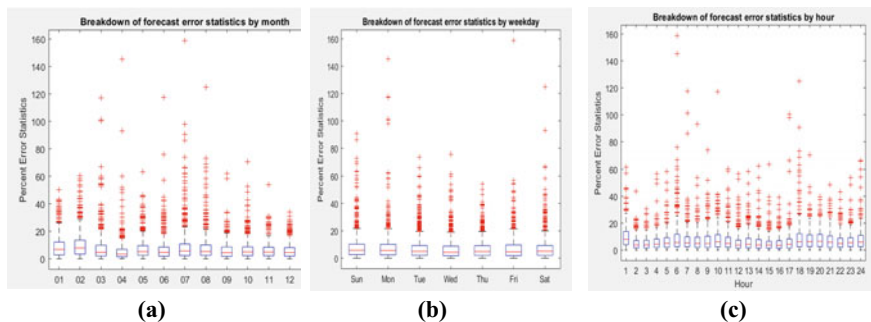


Fig. 21 NARX model-forecast error of N2 data for the year 2017. **a** Left panel: by months of year-July is the most error-prone month. **b** middle pane: by days of week-more errors on Mondays and Saturdays. **c** right panel: by hours of day-most of the errors occur at 6 am and 6 pm

4 Conclusions

In this paper, attempts are made to forecast the price of hourly electricity day-ahead market of IEX. The primary approach of visualizing historical data at different time zones (peak, Non-Peak, Day, Morning, Night, etc.) and thirteen bid areas and MCP covering different years give leverages to gauge volatility and the presence of strong seasonality in the market pricing mechanism. It also indicates the occurrence of very high as well as low pricing due to several reasons, including shortage of coal, low hydel power generation, increased demand due to festivity, growth in transmission and distribution capacity, renewable energy generation, application of charges/Govt. and other taxes, etc. For fitting the historical price data into suitable models, efforts are taken to reduce error in forecasting by application of conventional statistical method and use of the concept of neural network models. It starts with the ARIMA and SARIMA models and then, to achieve higher forecasting accuracy, ANN is used for modeling by training the existing dataset. A few of such external factors like weather data, economic data, demand–supply position, and coal stock/shortage are considered as inputs to make suitable model for time series prediction. For all models, the accuracy for forecasting is measured against actual data (test data), and MAE and MAPE are calculated. MAPE as low as 5.74% in price forecasting is achieved. However, the weather and economic data at higher resolutions and effect of other influencing factors as mentioned above could have been considered for further accuracy in forecasting DAM prices using ANN with external inputs. The findings of the work can lead to considerable accuracy in forecasting the price of one of the most essential Indian power exchange, IEX, which may assist Indian power market players in taking correct decisions.

As the future scope, a GUI (graphical user interface) can be prepared using MATLAB commands with ANN to forecast Indian power exchange hourly prices on a day-ahead basis at all its bid areas and MCP. The major exogenous inputs apart from those already considered in the work like growth in transmission and distribution capacity, renewable energy generation, application of charges/taxes, etc. can be considered as future scope of work to improve forecasting efficiency. The models used in our work may be tested for price forecasting of European markets, which are quite matured and thus competitive in nature.

References

1. Shukla UK, Thampy A (2011) Analysis of competition and market power in the wholesale electricity market in India. *Energy Policy* 39:2699–2710. <https://doi.org/10.1016/j.enpol.2011.02.039>
2. Report on Short-term Power Market in India: 2018–19. <https://cercind.gov.in/>
3. Annual Report: 2018–2019. <https://cea.nic.in/>
4. Indian Energy Exchange (IEX). <https://www.iexindia.com/>
5. Girish GP, Rath BN, Akram V (2018) Spot electricity price discovery in Indian electricity market. *Renew Sustain Energy Rev* 82:73–79. <https://doi.org/10.1016/j.rser.2017.09.009>

6. Liebl D (2013) Modeling and forecasting electricity spot prices: A functional data perspective. *The Annals of Applied Statistics*. 7(3):1562–1592. <https://doi.org/10.1214/13-AOAS652>
7. Benth FE, Kiesel R, Nazarova A (2012) A critical empirical study of three electricity spot price models. *Energy Econ* 34:1589–1616. <https://doi.org/10.1016/j.eneco.2011.11.012>
8. Ghosh D, Dutta S, Chakraborty S (2015) Multifractal detrended cross-correlation analysis of market clearing price of electricity and SENSEX in India. *Physica A*, 52–59. <https://doi.org/10.1016/j.physa.2015.03.082>
9. Girish GP (2016) Spot electricity price forecasting in Indian electricity market using auto-regressive-GARCH models. *Energy Strategy Rev*, 52–57. <https://doi.org/10.1016/j.esr.2016.06.005>
10. Pandey SN, Tapaswi S, Srivastava L (2008) Nodal congestion price estimation in spot power market using artificial neural network. *IET Generation, Trans & Distri* 2(2):280–290. <https://doi.org/10.1049/iet-gtd:20070309>
11. Peter SE, Raglend IJ (2017) Sequential wavelet-ANN with embedded ANN-PSO hybrid electricity price forecasting model for Indian energy exchange. *Neural Comput & Applic* 28:2277–2292. <https://doi.org/10.1007/s00521-015-2141-3>
12. Pany PK, Ghoshal SP (2015) Dynamic electricity price forecasting using local linear wavelet neural network. *Neural Comput & Applic* 26:2039–2047. <https://doi.org/10.1007/s00521-015-1867-2>
13. Rani RHJ, Victoire TAA (2019) A hybrid Elman recurrent neural network, group search optimization, and refined VMD-based framework for multi-step ahead electricity price forecasting. *Soft Comput* 23:8413–8434. <https://doi.org/10.1007/s00500-019-04161-6>
14. Deepa SN, Gobu B, Jaikumar S, Arulmozhi N, Kanimozhi P, Victoire TAA (2018) Adaptive regularized ELM and improved VMD method for multi-step ahead electricity price forecasting grid. 17th IEEE International Conference on Machine Learning and Applications. 2018. <https://doi.org/10.1109/ICMLA.2018.00204>
15. Girish GP, Vijaylakshmi S (2013) Determinants of electricity price in competitive power market. *Int J Business Management*. 8(21):70–75. <https://doi.org/10.5539/ijbm.v8n21p70>
16. Girish GP, Tiwari AK (2016) A comparison of different univariate forecasting models for Spot Electricity Price in India. *Econ Bull* 36(2):1039–1057
17. Data: all India mean temperature. <https://www.data.gov.in/>
18. Data: gross domestic product of India. <https://www.mospi.gov.in/>
19. Data: report archives. <https://www.cea.noc.in/>
20. Siddiqui Z, d'Aertrycke GM, Smeers Y (2012) Demand response in Indian electricity market. *Energy Policy* 50:207–216. <https://doi.org/10.1016/j.enpol.2012.06.030>
21. Aggarwal SK, Saini LM, Kumar A (2009) Electricity price forecasting in deregulated markets: A review and evaluation. *Electr Power Energy Syst* 21:13–22. <https://doi.org/10.1016/j.ijepes.2008.09.003>
22. Dawn S, Tiwari PK, Goswami AK (2017) An approach for efficient assessment of the performance of double auction competitive power market under variable imbalance cost due to high uncertain wind penetration. *Renew Energy* 108:230–243. <https://doi.org/10.1016/j.renene.2017.02.061>
23. Contreras J, Espinola R, Nogales FJ, Conejo AJ (2003) ARIMA models to predict next-day electricity prices. *IEEE Trans Power Syst* 18(3):1014–1020. <https://doi.org/10.1109/TPWRS.2002.804943>
24. Yadav A, Peesapati R, Kumar N (2017) Electricity price forecasting and classification through wavelet–dynamic weighted PSO–FFNN approach. *IEEE Syst* 12(4):3075–3084. <https://doi.org/10.1109/JSYST.2017.2717446>
25. Agarwal A, Ojha A, Tewari SC, Tripathi MM (2014) Hourly load and price forecasting using ANN and Fourier Analysis. 6th IEEE PIICON. 5–7 Dec. <https://doi.org/10.1109/POWERI.2014.7117736>
26. Panapakidis IP, Dagoumas AS (2016) Day-ahead electricity price forecasting via the application of artificial neural network based models. *Appl Energy* 172:132–151. <https://doi.org/10.1016/j.apenergy.2016.03.089>

27. de Martin MS, Bracco S, Asensio ER, Piazza G, Delfino F, Giribone PG (2020) Electricity spot prices forecasting for MIBEL by using deep learning: a comparison between NAR, NARX and LSTM networks. IEEEIC/I&CPS Europe. Date of Conference 9–12 June 2020. <https://doi.org/10.1109/EEEIC/ICPSEurope49358.2020.9160587>
28. Ezzeldin R, Hatata A (2018) Application of NARX neural network model for discharge prediction through lateral orifices. *Alex Eng J* 57(4):2991–2998. <https://doi.org/10.1016/j.aej.2018.04.001>
29. Uniejewski B, Marcjasz G, Weron R (2019) On the importance of the long-term seasonal component in day-ahead electricity price forecasting; Part II-probabilistic forecasting. *Energy Econ* 79:171–182. <https://doi.org/10.1016/j.eneco.2018.02.007>
30. Mandal P, Senjyu T, Uezato K, Funabasi T (2005) Several-hours-ahead electricity price and load forecasting using neural networks. *IEEE Power Eng Soc Gen Meet* 3:2146–2153. <https://doi.org/10.1109/PES.2005.1489530>

Traffic Density Classification for Multiclass Vehicles Using Customized Convolutional Neural Network for Smart City



Deepak Mane, Ranjeet Bidwe, Bhusan Zope, and Nihar Ranjan

Abstract Building a traffic monitoring system for intelligent transportation systems (ITS) in the developing smart cities has drawn in a mass of consideration in the latest past. Since the majority of cities in the world are observing the increasing number of vehicles on the road, they are tending to accept an intelligent transportation system for resolving tedious issues like traffic density, count of traffic lines and their length, the standard speed of the traffic, and increase in the number of vehicles during weekends or for a particular time span in a day. Smart transportation systems are provided with traffic pictures and recordings by installed cameras on roads or signals. Also, different types of sensor-actuator pairs help check and deal with traffic issues. This paper proposed a method that uses customized convolution neural network (CCNN) on traffic images to classify images as per the traffic density and thereby provide driving assistance. The proposed system in the paper can monitor traffic using videos captured by installed cameras and then classify the current traffic situation into categories of high, medium, and low categories. The aim is to use this as a model to provide traffic density information from various places to expert systems and take other important decisions regarding traffic control. NVIDIA graphics processing unit (GPU) is used to parallelize the training process and implement complex deep neural networks to obtain better accuracy. Performance evaluation of the proposed system is done on a real-time dataset containing recorded footage of traffic from Pune city (India) and recorded footage by highway CCTVs at Seattle, WA, obtained from the Washington State Department of Transport. Experiment results classify traffic density into high, medium, and low based on existing traffic and predict it correctly up to 99.6%. Obtained testing accuracy is much better than the results given by existing algorithms.

Keywords Deep learning · Intelligent traffic system · Convolutional neural network · Pattern recognition · Computer vision · Traffic image analysis

D. Mane (✉) · N. Ranjan
JSPM's Rajarshi Shahu College of Engineering, Pune, Maharashtra 411033, India
e-mail: dtmane@gmail.com

R. Bidwe · B. Zope
Pune Institute of Computer Technology, Pune, Maharashtra 411043, India

1 Introduction

Machine learning is the study of teaching computers to mimic like a human without programming them explicitly. Machine learning aims to make computers find the solution by observing many similar sample examples, a process very similar to how we teach toddlers. One of the vast areas of research under machine learning is deep learning. In deep learning, many learning layers are used to find the maximum available information. For feature extraction, many explicit methods have been developed. However, the main advantage of deep learning is that it does not require any external information and can extract the features on its own. Deep learning's ability to learn by comprehension makes it a very powerful tool. The motivation behind deep learning is twofold: (1) in order to mimic the human brain and its deep architecture; (2) the more depth, the better the efficiency. A few examples of popular deep learning algorithms are deep neural networks, convolutional neural networks [17], deep belief network [18], recursive neural network [19], recursive Boltzmann machine [20], etc. Since traffic congestion and roadblocks are serious problems, especially in India, it would help if drivers were informed of traffic situations with the help of systems that have learned to predict traffic density accurately.

Currently, GPS tracking estimates the time required for traveling starting point A to point B. This drains the smartphone battery. Therefore, the main intend of this effort is to reduce this burden on a smartphone. The other few objectives of this study are

- To predict the traffic congestion that will re-route the traffic and prevent further emergencies that may be generated because of traffic congestion.
- To make the traffic monitoring system more elastic to predict and avoid traffic congestions in advance and the existing system can adapt according to the traffic situation.
- To contribute to the smart city concept by predicting traffic congestions in advance and avoiding it.

Organization of the paper flow as follows: In Sect. 2, it represented the literature survey for traffic density. Proposed CCNN architecture explained in Sect. 3. An experiment result on the proposed model is represented in the Sect. 4. Conclusion with future direction is explained in the last Sect. 5

2 Related Work

The survey shows the variety of applications of CNN which are majorly in areas of computer vision and image classification. Use of CNN in vehicle type identification and vehicle logo detection is well demonstrated in [1] and [2], respectively. Appearance-based vehicle type classification method is proposed in [1] which uses frontal view images of vehicles. CNN can also be used to identify the vehicle logos

which then can be verified using combination of SVM and pyramid of histogram of oriented gradients (PHOG) [2]. Other popular applications of CNN are forest species recognition [5], handwriting character recognition of Telugu language [6] and for handwritten digits. It has applications in large-scale image classification [4] and a lot of scope for research and improvement [3]. Thus, we have decided to utilize the power of automatic feature extraction of hierarchical features, of convolutional neural networks for a different-related traffic issue, to predict the traffic density by image analysis.

In [7], a classifier is designed to distinguish between various types of flows. Here, the motion of objects like traffic motion is a challenging visual process. Visual processes have been classified by using a KL kernel for dynamic textures, using a variety of datasets, including traffic [8]. Hence, we are using the same traffic flow dataset but the CNN technique to predict the traffic density from the extracted CCTV frames. System from reference [9] has used automatic detection traditional asset called intelligent transport systems (ITS). This system internally uses legacy camera systems and does a manual surveillance of vehicles. Authors have integrated computer vision techniques on videos captured by installed cameras to extract traffic information. The system uses state-of-the-art objects detectors to classify pedestrians, cyclists, vehicles. Also proposed system is integrated with CNN to classify various kinds of vehicles. The system can estimate speed of vehicle (also predicted by the system in [21]) with vehicle count. The approach used in reference [10] proposes a vehicle detection system for video surveillance. The system calculates results from busy streets in the city, including different kinds of vehicles. The proposed system in the paper classifies vehicles in the traffic in three categories, such as light (includes motorbikes, bikes, tricycles), medium (includes cars, sedans, SUVs), and heavy vehicles (trucks, buses) and able to modify conclusion with motorbikes' sudden changes in direction. Authors claim accuracy of 95.3% for detecting traffic density at place. Reference [11] has identified potential issue of traffic in major city and proposed a system that is capturing the video data from the installed cameras.

Captured data will be provided to edge computing techniques for vehicle classification and counting. Further data are uploaded to the database, based on the number of vehicles and density; the system will automatically decide green light duration. Reference [12–14] summarizes the sensors and technologies used in vehicle detection and traffic estimation. It reviews various traffic detection mechanisms using intrusive and non-intrusive sensors. Pair of sensors and actuators has a vital application in intelligent transportation systems. These modern devices can determine the vehicle count, location, speed, traffic density, traffic estimation. This paper motivates to use fusion of sensors and computer vision techniques for prediction of traffic density and better accuracy. In [15], paper has explained how deep learning algorithms used to detect moving objects from satellite images. Also, it explains various issues like over fitting and low performance which current deep learning algorithms are facing. This paper insists to use deep neural-based architectures instead of classifiers with features, and this paper claims to achieve promising accuracy using such methods. Proposed system also takes help of existing methods like YOLO and faster region-based convolutional neural networks (faster R-CNNs) to detect vehicles using

Table 1 WSTD sequential model

Layer (type)	Output shape	Parameter #
Conv2D	(none, 64, 64, 32)	896
Conv2D_1	(none, 62, 62, 32)	9248
Max_pooling2D	(none, 31, 31, 32)	0
Dropout	(none, 31, 31, 32)	0
Conv2D_2	(none, 31, 31, 128)	36,992
Max_pooling2D_1	(none, 15, 15, 128)	0
Dropout_1	(none, 15, 15, 128)	0
Flatten	(none, 28,800)	0
Dense	(none, 512)	14,746,112
Dropout_2	(none, 512)	0
Dense_1	(none, 3)	1539
Total params: 14,794,787		
Trainable params: 14,974,787		
Non-trainable parameter: 0		

different special locations. Results received in this paper are exceptional since it can identify smaller objects that current latest similar approaches cannot do. The system proposed in [16] ensembles various classifiers to solve problem of traffic congestion. This system is state of the art since it can reduce traffic congestion, travel time, fuel consumption, and CO2 emission. There are many methods have been proposed for traffic density classification and management, and they are producing promising results (refer Table 1). Few noteworthy techniques used in them are SVM for classification [28], KNN for two step classification is proposed in [29]. Proposed system in [30] uses two stages CNN and used softmax classifier. Method from [31] does automatic feature extraction using deep neural networks. Deep CNN is used for feature extraction, and SVM is used for classification in [32]. Data augmentation and normalization are effectively being used in [33, 34] and producing very promising results. Many systems use combination of different types of sensors with ML to produce efficient results [35] and also [36, 37] are using DNN to give promising results. A very descriptive survey of all the classification methods is explained in [38]. It provides details of classification using different kind of sensors which are ensemble with modern machine learning techniques.

3 Proposed Architecture

Aside from the design of the traditional CNN architecture [22], several distinct examination has proposed varieties in different parts of the model like increases the depth of convolution layers, average pooling, variations in filter size etc. [23–26]. In presented

model, customized CNN architecture in a way that that it recognizes and classify the traffic density images and give up the better results over all the previous methods. Presented customized CNN model does not put any limitation on the quantity of different layers; rather, it enhances the number to fulfill the need of the mentioned issue. Moreover, a different kernel size like 5*5, 3*3 has been utilized for different convolutional layers. Proposed model can monitor traffic using videos captured by installed cameras and then classify the current traffic block. The aim is to use this as a model to provide traffic density information from various places to expert systems and take further important decisions regarding traffic controlling. NVIDIA graphics processing unit (GPU) is used to parallelize the training process and implement complex deep neural networks to obtain better accuracy. Figure 1 shows the general framework design of proposed model.

CCNN consists of convolutional layers which are followed by subsampling or pooling layers which is then attached to a fully connected multi-layered perceptron. It exploits the 2D structure of images. The main aim of CCNN is to provide automatic extraction of hierarchical features from images. Random initialized filters are used for performing the convolution to reduce input size without loss in features which is possible by increased number of feature maps. The range of the values of the filter is a function of the number of input feature maps to the next layer. Usually, after convolution, pooling is performed. We have used max pooling in which the maximum value of pixels is considered and represented as a single pixel value. The feature leads toward the maximum number of features having the similar values. The finally extracted features are passed on to the fully connected multi-layered perceptron (MLP) to obtain a classification of the test input. In the proposed system, the CCNN classifier is used to detect traffic density. We changed the different tuning parameters in such way that it is produced the better testing accuracy than the existing. After acquiring the data, the traffic images are rescaled to pass to layers of CNN. Each input is passed to convolutional layer and max pooling layer for feature extraction. The added extra convolutional layers and max pooling layers helped to achieve the higher accuracy than existing systems. Figure 2 represents the CCNN architecture of proposed system. CNN provides high accuracy for image classification and

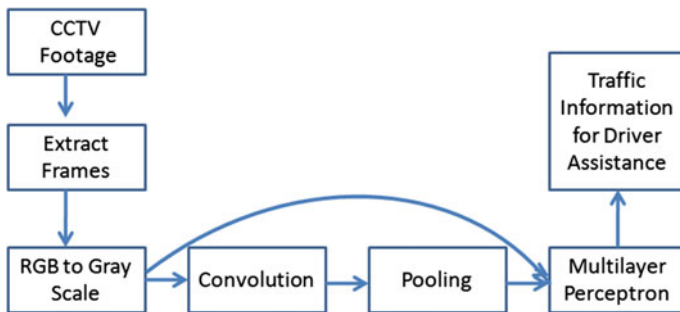


Fig. 1 General framework architecture for traffic density classification

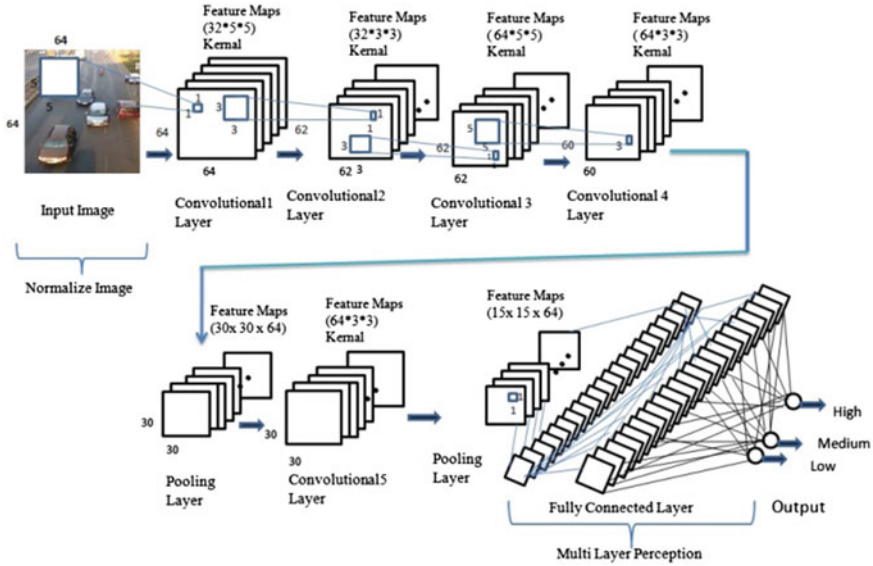


Fig. 2 Customized CNN model

recognition as compared to other algorithms [27]. CCNN model is represented in Fig. 2.

3.1 CCNN Learning Algorithm

This section will discuss the mathematics behind all the operations discussed above.

1. *Convolution operation:*

The convolution layer takes the output of the previous layer, $\alpha^{[l-1]}$ with size $(h^{[l-1]}, w^{[l-1]})$ as an input, $\alpha^{[0]}$ being the input image. Then, convolution operator \odot is used on this input using filter/kernel \mathcal{K} .

$$(\alpha^{[l-1]} \odot \mathcal{K})_{x,y} = \sum_1^{h^{[l-1]}} \sum_1^{w^{[l-1]}} \mathcal{K}_{i,j} * \alpha_{(x+i-1,y+j-1)}^{[l-1]} \tag{1}$$

2. *Activation function:*

To bring the non-linearity in the transformation, after convolution operation, activation function ψ is used. So the output of the layer will be

$$\psi((\alpha^{[l-1]} \odot \mathcal{K})_{x,y}) = \psi\left(\sum_1^{h^{[l-1]}} \sum_1^{w^{[l-1]}} \mathcal{K}_{i,j} * \alpha_{(x+i-1,y+j-1)}^{[l-1]}\right) \tag{2}$$

The most popular choices for activation function ψ are sigmoid, tanh, ReLU, etc. Since other functions suffer from vanishing gradient problem, the ReLU function is used as an activation function. Hence in this case, output will be

$$\text{Max}\left(0, \psi\left(\sum_1^{h^{[l-1]}} \sum_1^{w^{[l-1]}} \mathcal{K}_{i,j} * \alpha_{(x+i-1,y+j-1)}^{[l-1]}\right)\right) \tag{3}$$

3. *Pooling Layer:*

The exact position of the feature in the image is not important. However, sometimes, this spatial information is also recorded. One way to avoid this is to downsample it. Pooling layer helps us to achieve it. Max pooling technique is used in our approach. Given a matrix A of size (n1, n2) and pooling filter of size k, pool operator will find the maximum value of k × k block.

4. *Fully connected layer:*

The output of the convolution layer or pooling layer will be a 2D matrix due to the single channel present. To input this to a fully connected layer, we need to flatten the 2D (3D in general) array and produce the vector. This vector is then given to a fully connected layer.

Forward propagation: In *i*th layer, in the forward pass, output vector is calculated as

$$A^{[i]} = \psi^{[i]}(Z^{[i]}) = \psi^{[i]}(W^{[i]T} A^{[i-1]} + b^{[i]}) \tag{4}$$

where $W^{[i]}$: is weight vector for *i*th layer.

$b^{[i]}$: Bias term for *i*th layer.

$\psi^{[i]}$: Activation functions for *i*th layer; from this term, it is evident that one can have different activation functions for different layer. In the proposed system, ReLU is used for all but one layer, and sigmoid is used for the last layer.

Back propagation: In this phase, error committed in the prediction is inserted into the network to update the parameters accordingly.

4 Experimental Results

Performance measurement of the presented model is done on real-time dataset containing recorded footage of traffic from Pune city (India) and recorded footage by highway CCTVs at Seattle, WA, which is obtained from the Washington State Department of Transport. Experiment results classify traffic density into high, medium, and

low based on existing traffic and predicting it correctly up to 99% and 99.6%, respectively. The architecture chosen by us is five-layer architecture with varying filter sizes and number of input feature maps. The number of feature maps for the pair of convolution and max pooling layers is 32, 32, 64, 64, and 64, and the filter sizes for each of the five pairs of convolution and max pooling layers are 5×5 , 3×3 , 5×5 , 3×3 , and 3×3 . The last layer is a fully connected multi-layered perceptron with two layers which are fully connected and is used to obtain the final classification of the image into three classes: zero—for low traffic, one- for medium traffic, and two- for high traffic. Convolution and max pooling process were repeatedly carried out 5 times and 2 times, respectively, to extract the features from the image automatically. The activation function used was rectified linear unit (ReLU) which is known to give a better output for complex problems, and the dropout used for the fully connected MLP is 0.5 to discard some neurons in the training process. The number of output neurons in our case is 3, for three classes of traffic density.

4.1 The Learning Steps for CCNN Technique Are

- Obtain frames from the CCTV videos using FFmpeg utility.
- Obtain pre-processed input. (extracted pixels which are converted to grayscale image pixels)
- Normalize the images: 0–1 normalization.
- Perform convolution in convolution layer
- Perform max pooling with pool size (2, 2) to obtain one average from 4 pixels.
- Connect to output neurons through a fully connected MLP.
- Train the CCNN on the training data of traffic images.
- The usual back propagation algorithm is carried out on the convolution neural network to propagate errors backward.
- Required parameters of the model are obtained after training.

4.2 Dataset

There are two datasets are used to check the performance of the proposed CCNN approach. In first, WSDT dataset, there are total 13,998 images of size 64×64 along with their classes (0, 1, or 2) and out of that 70% used for training and 30% used for testing. Class wise samples of each class are represented in Fig. 3. The data were converted to the corresponding grayscale pixels for that image and stored in .CSV files which were provided as an input for training. The dataset was obtained from the Washington State Department of Traffic Transportation [8] and used for the classification of auto-regressive visual processes [7] and consists of videos of real-time traffic from Seattle, Washington. The frames were extracted from these CCTV videos by using the FFmpeg Linux utility by specifying the number of images to be



Fig. 3 Class wise samples of WSDT dataset

captured from the video. The images originally were of size 320* 240 and then were cropped to 64*64 before converting them to .CSV files. From the presented CNN architecture, we have obtained an accuracy of 99% on a dataset of 13,998 images. Further, we have trained the CNN on GPUs for 100 epochs.

In second dataset, recorded footage of traffic from Pune city (India) is used which then converted into images. Total of 920 images of sized 64×64 along with their classes (0, 1, and 2) and out of that 70% used for training and 30% used for testing. Class wise samples of each class are represented in Fig. 4. The data were converted to the corresponding grayscale pixels for that image and stored in .CSV files which were provided as an input for training. From the presented CNN architecture, we have obtained an accuracy of 99.6% on a dataset of 920 images.

Figure 5 shows total no. of samples in each category in the WSDT dataset. Class low label 0 has 4399 images in data, class medium label 1 has 4492 images in data, and class high label 2 has 4507 images in data. Figure 6 shows Indian dataset the total no. of samples in each category. Class low label 0 has 298 images, class medium DR label 1 has 320 images in data, and class high label 2 has 302 images in data.



Fig. 4 Class wise samples of Indian dataset

Fig. 5 Category wise WSDT dataset

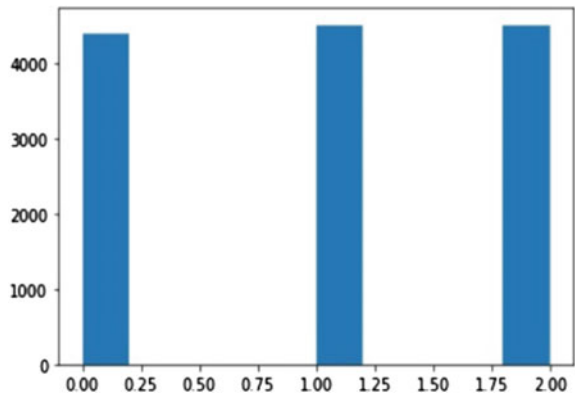
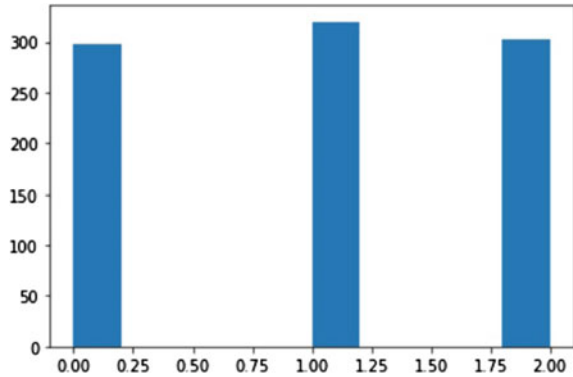


Fig. 6 Category wise Indian dataset



CCNN sequential model WSTD dataset and Indian dataset are represented in following Tables 1 and 2. Figures 7 and 8 represent the different loss graph at the time of training. Classification reports of WSTD and Indian dataset are represented in Tables 3 and 4. Classification report values are obtained after finding true positives (TP, presence of object classified correctly), true negative (TN, absence of object classified correctly), false positive (FP, presence of object classified incorrectly), and false negative (FN, absence of object classified incorrectly).

Confusion matrix will evaluate proposed classifier based on the obtained TP, TN, FP, FN values. Precision value explains correct classification of the objects by the

Table 2 Indian traffic sequential model

Layer (type)	Output shape	Parameter #
Conv2D	(none, 64, 64, 32)	2432
Conv2D_1	(none, 62, 62, 32)	9248
Conv2D_2	(none, 62, 62, 64)	51,264
Conv2D_3	(none, 60, 60, 64)	36,928
Max_pooling2D	(none, 30, 30, 64)	0
Dropout	(none, 30, 30, 64)	0
Conv2D_4	(none, 30, 30, 64)	36,928
Max_pooling2D_1	(none, 15, 15, 64)	0
Dropout_1	(none, 15, 15, 64)	0
Flatten	(none, 14,400)	0
Dense	(none, 512)	7,373,312
Dropout_2	(none, 512)	0
Dense_1	(none, 3)	1539
Total params: 7,511,651		
Trainable params: 7,511,651		
Non-trainable parameter: 0		

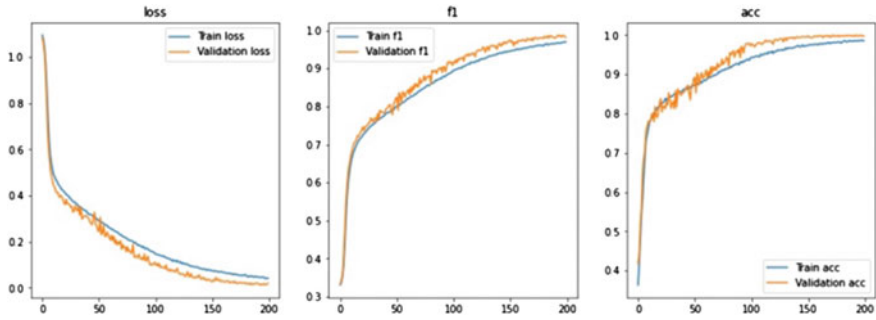


Fig. 7 WSTD dataset training versus validation loss

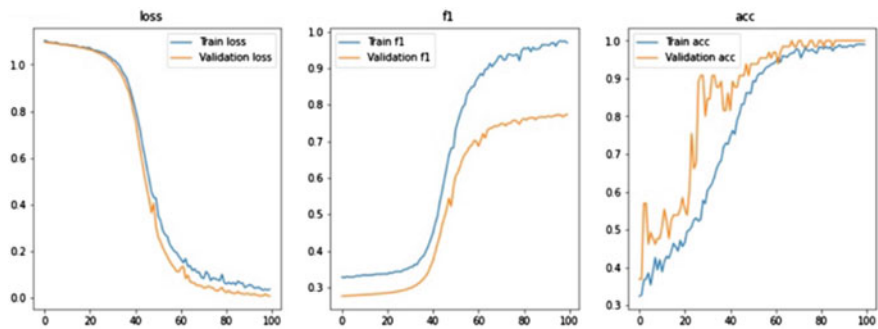


Fig. 8 Indian dataset training versus validation loss

Table 3 WSTD classification report

	Precision	Recall	F1-Score	Support
Class 0 (low)	0.98	1.00	0.99	1341
Class 1 (medium)	1.00	0.98	0.99	1345
Class 2 (high)	1.00	1.00	1.00	1354
Accuracy			0.99	4020
Micro avg	0.99	0.99	0.99	4020
Weighted avg	0.99	0.99	0.99	4020

Table 4 Indian dataset classification report

	Precision	Recall	F1-Score	Support
Class 0 (low)	1.00	1.00	1.00	84
Class 1 (medium)	1.00	0.99	0.99	100
Class 2 (high)	0.99	1.00	0.99	92
Accuracy			1.00	276
Micro avg	1.00	1.00	1.00	276
Weighted avg	1.00	1.00	1.00	276

classifier; recall is a measure of correct classification of the object from all positive objects. F1-score is harmonic mean of precision and recall. Support is representing total sample size. Accuracy of the model is calculated by drawing confusion matrix and finding count of correct predictions made by model from all possible predictions.

Prediction of actual vs predicted images is represented by using the confusion matrix in Figs. 9 and 10. Table 5 helps to understand the existing accuracies of algorithms, and it can be stated that ensemble methods and R-CNN are the two algorithms that give accuracy closer to CCNN algorithm. Result comparison of proposed system with existing/proposed algorithms is represented in Table 5.

Here, the dataset used for simply density classification. The existing technique for predicting the travel time involves the use of GPS of users to indicate how much time a user takes to move from one place to another. Algorithms like blob have been used in the past, and they simply indicate the number of vehicles that passed, but location specific information needs to be provided in advance. This paper aims to eliminate the need for human intervention in traffic density prediction.

Fig. 9 WSTD confusion matrix

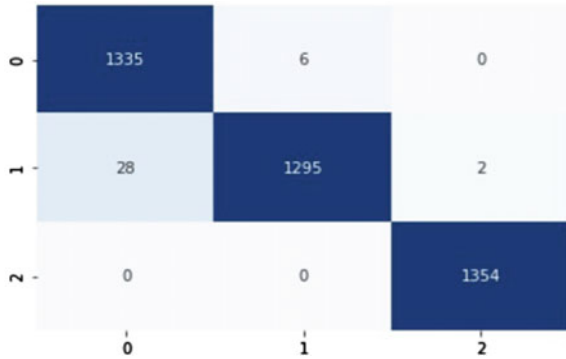


Fig. 10 Indian dataset confusion matrix

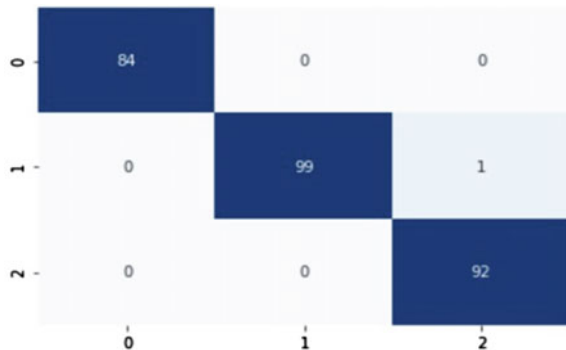


Table 5 Result comparison of proposed system with existing algorithms

Year and Ref	Method	Accuracy (%)
(2012) [28]	SVM for classification	94.6
(2012) [29]	Two step KNN	92.6
(2012) [30]	Automatic feature extraction using two-stage CNN	89.4
(2016) [31]	Automatic feature extraction by DNN	97
(2017) [32]	Feature extraction using DCNN, SVM for classification	89
(2017) [33]	Data augmentation, ensemble CNN	97.8
(2017) [34]	Data normalization and augmentation	80
(2017) [35]	Uses both infrared sensors and ultrasonic sensors; classification using Bayesian network and neural network	Up to 99
(2018) [36]	Advanced DNN	95.46
(2018) [37]	ML-based classification	97
(2020) [10]	Convolutional neural network	95.3
(2020) [16]	Ensemble (fuzzy, KNN, ANN-MLP)	95
(2021) [15]	YOLO, faster R-CNN	96.26
Proposed system	CCNN	99 and 99.6

5 Conclusion

In this paper, CCNN model is presented for better pattern recognition on traffic images, to classify images as per the traffic multiclass vehicles situation into high, medium, and low categories and thereby provide driving assistance. Presented model does not put any limitation on the quantity of different layers; rather, it enhances the number to fulfill the need of the mentioned issue. GPU along with CPU is used to reduce the training time. Performance evaluation of the proposed system is done on real-time dataset containing recorded footage of traffic from Pune city (India) and recorded footage by highway CCTVs at Seattle, WA, which is obtained from the Washington State Department of Transport. Experiment results classify traffic density into high, medium, and low based on existing traffic and predicting it correctly up to 99.6% on Indian traffic dataset and 99% WA traffic dataset. Due to uproarious and vulnerability in the comparability a type of vehicle, the presented model sometimes shows improper results which can be more improved in future.

References

1. Dong Z, Wu Y, Pei M, Jia Y (2015) Vehicle type classification using a semisupervised convolutional neural network. *IEEE Trans Intell Transp Syst* 16:2247–2256
2. Thubsang W, Kawewong A, Patanukhom K (2014) Vehicle logo detection using convolutional neural network and pyramid of histogram of oriented gradients. In: 11th International Joint Conference on Computer Science and Software Engineering (JCSSE), pp 34–39

3. Yim J, Ju J, Jung H, Kim J (2015) Image classification using convolutional neural networks with multi-stage feature. In: Kim JH, Yang W, Jo J, Sincak P, Myung H (Eds) Robot intelligence technology and applications 3. *Advances in Intelligent Systems and Computing*, vol 345. Springer, Cham
4. Xiao T, Zhang J, Yang K, Peng Y, Zhang Z (2014) Error-driven incremental learning in deep convolutional neural network for large-scale image classification. In: proceedings of the 22nd ACM international conference on Multimedia
5. Hafemann LG, Oliveira LE, Cavalin PR (2014) Forest species recognition using deep convolutional neural networks. In: 22nd International Conference on Pattern Recognition, 1103–1107
6. Soman ST, Nandigam A, Chakravarthy VS (2013) An efficient multiclassifier system based on convolutional neural network for offline handwritten Telugu character recognition. In: 2013 National Conference on Communications (NCC), 1–5
7. Chan AB, Vasconcelos N (2005) Probabilistic kernels for the classification of auto-regressive visual processes. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), pp 846–851 vol. 1
8. Source: <http://www.wsdot.wa.gov/> last accessed in 25/7/2015
9. Liu C, Huynh DQ, Sun Y, Reynolds M, Atkinson S (2021) A vision-based pipeline for vehicle counting, speed estimation, and classification. In: *IEEE Transactions on Intelligent Transportation Systems*, vol 22, no 12, pp 7547–7560
10. Pham LH, Phan HN, Chung NM, Vu T, Ha SV (2020) A robust multiclass vehicle detection and classification algorithm for traffic surveillance system. In: 2020 RIVF International Conference on Computing and Communication Technologies (RIVF), pp 1–6
11. Deshmukh P, Gupta D, Das SK, Sahoo UK (2020) Design of a traffic density management and control system for smart city applications. In: Mallick P, Balas V, Bhoi A, Chae GS (Eds), *Cognitive informatics and soft computing. Advances in Intelligent Systems and Computing*, vol 1040. Springer, Singapore
12. Tasgaonkar PP, Garg RD, Garg PK (2020) Vehicle detection and traffic estimation with sensors technologies for intelligent transportation systems. *Sens Imaging*, 21–29
13. Frniak M, Markovic M, Kamencay P, Dubovan J, Benco M, Dado M (2020) Vehicle classification based on FBG sensor arrays using neural networks. *Sensors* 20(16):4472
14. Shokravi H, Shokravi H, Bakhary N, Heidarrezaei M, RahimianKoloor SS, Petrů M (2020) A review on vehicle classification and potential use of smart vehicle-assisted techniques. *Sensors* 20:3274
15. Vasavi S, Priyadarshini NK, Harshavaradhan K (2021) Invariant feature-based Darknet architecture for moving object classification. *IEEE Sensors J* 21(10):11417–11426
16. Rocha Filho GP, Meneguette RI, Torres Neto JR, Valejo A, Weigang L, Ueyama J, Pessin G, Villas LA (2020) Enhancing intelligence in traffic management systems to aid in vehicle traffic congestion problems in smart cities. *Ad Hoc Networks* 107:102265
17. Albawi S, Mohammed TA, Al-Zawi S (2017) Understanding of a convolutional neural network. In: International Conference on Engineering and Technology (ICET), pp 1–6
18. Geoffrey E (2009) Hinton: deep belief networks. *Scholarpedia* 4(5):5947
19. Setiono R, Baesens B, Mues C (2008) Recursive neural network rule extraction for data with mixed attributes. *IEEE Trans Neural Networks* 19(2):299–307
20. Zhang J (2011) Deep transfer learning via restricted Boltzmann machine for document classification. In: 10th International Conference on Machine Learning and Applications and Workshops, pp 323–326
21. Yang H, Liu C, Zhu M, Ban X, Wang Y (2021) How fast you will drive? Predicting speed of customized paths by deep neural network. In: *IEEE Transactions on Intelligent Transportation Systems*, pp 1–11
22. LeCun Y, Boser B, Denker JS, Henderson D, Howard RE, Hubbard W, Jackel LD (1989) Handwritten digit recognition with a back-propagation network. *Advances in Neural Info Processing Systems, NIPS* 2:396–404

23. Mane D, Kulkarni U (2017) A survey on supervised convolutional neural network and its major applications. *Int J Rough Sets Data Anal* 4:71–82
24. Mane D, Kulkarni U (2018) Visualizing and understanding customized convolutional neural network for recognition of handwritten Marathi numerals. *Procedia Computer Science* 132:1123–1137
25. Mane DT, Kumbharkar PB, Dhotre PS, Borde S (2021) Vehicle-type classification using customized fuzzy convolutional neural network. In: *Data Engineering and Intelligent Computing. Advances in Intelligent Systems and Computing*, vol 1. Springer, Singapore, pp 419–429
26. Mane DT, Tapdiya R, Shinde SV (2021) Handwritten Marathi numeral recognition using stacked ensemble neural network. *Int J. Inf Technol* 13:1993–1999
27. Shinde S, Kulkarni U, Mane D, Sapkal A (2021) Deep learning-based medical image analysis using transfer learning. In: *A Computational Perspective in Healthcare. Studies in Computational Intelligence*, vol 932. Springer, Singapore, pp 19–42
28. Chen Z, Ellis T, Velastin SA (2012) Vehicle detection, tracking and classification in urban traffic. In: *International IEEE Conference on Intelligent Transportation Systems*, pp 951–956
29. Mithun NC, Rashid NU, Rahman SM (2012) Detection and classification of vehicles from video using multiple time-spatial images. *IEEE Trans Intelligent Transportation Syst* 13(3):1215–1225
30. Unzueta L, Nieto M, Cortés A, Barandiaran J, Otaegui O, Sánchez P (2012) Adaptive multicue background subtraction for robust vehicle counting and classification. *IEEE Transactions Intelligent Transportation Syst* 13(2):527–540
31. Huttunen H, Yancheshmeh FS, Chen K (2016) Car type recognition with deep neural networks. In: *IEEE Intelligent Vehicles Symposium (IV)*, pp 1115–1120
32. Adu-Gyamfi YO, Asare SK, Sharma A, Titus T (2017) Automated vehicle recognition with deep convolutional neural networks. *Transportation Res Record* 2645(1):113–122
33. Liu W, Zhang M, Luo Z, Cai Y (2017) An ensemble deep learning method for vehicle type classification on visual traffic surveillance sensors. *IEEE Access* 5:24417–24425
34. Audebert N, Le Saux B, Lefèvre S (2017) Segment-before-detect: Vehicle detection and classification through semantic segmentation of aerial images. *Remote Sens* 9(4):368
35. Odat E, Shamma JS, Claudel C (2018) Vehicle classification and speed estimation using combined passive infrared/ultrasonic sensors. *IEEE Trans Intelligent Transportation Syst* 19(5):1593–1606
36. Xu C, Wang Y, Bao X, Li F (2018) Vehicle classification using an imbalanced dataset based on a single magnetic sensor. *Sensors* 18(6):1690
37. Balid W, Tafish H, Refai HH (2018) Intelligent vehicle counting and classification sensor for real-time traffic surveillance. *IEEE Trans Intelligent Transportation Syst* 19(6):1784–1794
38. Won M (2020) Intelligent traffic monitoring systems for vehicle classification: A survey. *IEEE Access*, vol 8, pp 73340–73358

U-shaped Transformer for Enhancing Low-Dose CT Images



Aswin Unnikrishnan, Amal Pavithran, Arpith G. Naik, Abhishek P. Jiju, and P. V. Sudeep 

Abstract Image restoration is an predominant pre-processing step in computer vision and medical applications. In literature, different image restoration methods using deep learning have been reported. As a latest development in deep learning, transformer has proved its practicality in tasks such as machine translation, text generation, etc. In this paper, we study the performance of transformer-based deep image restoration for suppressing noise in low-dose CT images. To substantiate, we perform experiments on publicly available LDCT datasets and evaluated the results qualitatively and quantitatively.

Keywords Deep learning · Image denoising · Low dose CT · TED-net · Transformers · Uformer

1 Introduction

Imaging plays an important role in modern medicine. Although imaging modalities such as X-rays, computed tomography (CT) scans, MRI, and ultrasound are popular in medical diagnosis and treatment, these techniques are often corrupted by unwanted noise. Image restoration methods eliminate the noise and create high-quality images. Such images are vital for the visual inspection of the physicians in identifying diseases and for the post-processing such as image registration, segmentation and object detection from the images.

Although CT scanning plays a vital role in diagnosis, the ionizing radiation due to X-rays damages the tissue and DNA and causes cancer [8, 11] to the patient. One way to tackle this issue is by decreasing the radiation dose. However, the quality of CT image produced using reduced radiation dose of X-rays is poor when compared to normal dose CT (NDCT) images. Decreasing the radiation dose or the number of projection may lead to greater noise and artifacts, which can negatively affect the radiologists decision and trust.

A. Unnikrishnan · A. Pavithran · A. G. Naik · A. P. Jiju · P. V. Sudeep (✉)
National Institute of Technology Calicut, Kozhikode, Kerala 673601, India
e-mail: sudeep.pv@nitc.ac.in

In literature, the prior-driven algorithms [13] are reported by using the prior image data available from NDCT images for reconstructing the clean image. However, prior based techniques can be unreliable if the prior image and the current Low Dose CT (LDCT) image show different structures due to variations over time. Another widely explored LDCT denoising technique is the Iterative Reconstruction (IR) algorithms. It merges the statistical properties and prior information from the image data. IR reconstruction techniques have replaced the classical filtered back-projection (FBP) in most modern CT scanners. However, IR techniques are computationally costly and the reconstruction is slow.

In the past, deep neural networks (DNNs) have demonstrated promising improvements in image processing and computer vision tasks. DNNs such as auto-encoders (AEs) [10], convolution neural networks (CNNs) [3] and generative adversarial networks (GANs) [18] are utilized to recover clean medical images from its noisy versions. Medical images such as LDCT and MRI are usually affected by noise distributions other than Gaussian during its acquisition which can be removed by using deep neural networks.

Recently, transformers [14] are proposed in the field of Natural Language Processing (NLP) and is still being explored in the field of Computer Vision (CV). Transformer uses the mechanism of attention. Vision transformer [5] based architectures have shown superior performance in removing noise from natural images and RGB images. Inspired by this, we employed U-shaped transformers such as Uformer [16] and TED-net [15] for LDCT image restoration and compared with other methods such as RED-CNN [3] and deep image prior (DIP) [13].

The main contributions of the paper are listed below:

- Uformer model is used for LDCT image denoising.
- The Uformer model is compared with convolutional network models like TED-net [15] and RED-CNN [3] along with DIP [13].
- DIP model is introduced for the first time for LDCT image denoising.
- Uformer model is modified to use single-channel images.

The remainder of this paper is structured as follows: Sect. 2 includes the materials and methods used in the research work. The popular models used for LDCT image denoising are RED-CNN [3], TED-net [15] and DIP [13]. The concepts of CNN and transformer based models is further detailed in this section. Section 3 consists of experiments done in the research work. The architecture of both the Uformer [16] and TED-net [15] is displayed in this section. Information of the dataset and the performance metrics used is given in this section. Further the findings of qualitative and quantitative evaluation is displayed in this section.

2 Materials and Methods

LDCT images are popular medical imaging technique as it reduces the risk of X-ray radiation to patients when compared with NDCT images. Most of the deep

learning based LDCT image restoration uses CNNs, and Residual Neural Networks (RNNs) [3]. Denoising of LDCT images using CNN based architecture [4, 20–22] has been successful in achieving monumental results, but showed restriction in attaining long-range dependencies. In order to overcome this problem, few recent works have explored the use of image transformers in image denoising and have obtained competitive results with CNN based models.

U-shaped structures with skip-connection are becoming a popular solution for capturing multiscale information hierarchically for various image restoration tasks, including image denoising, deblurring and demoiereing. This has mostly been followed in the case of RGB images and now is being adopted for LDCT images in models such as RED-CNN [3], TED-net [15] and Uformer [16].

One of the earliest convolutional neural network used for LDCT image denoising is residual encoder-decoder convolutional neural network (RED-CNN) [3]. RED-CNN [3] incorporates the notion of deconvolution network [23] and short connections [6, 12] into the CNN model which assists in sustaining structural details and helps in training deeper networks. It further replaced the conventional encoder–decoder structure with residual learning which facilitates the working of the convolutional and its corresponding deconvolutional layer.

Deep image prior (DIP) [13] method uses a randomly initialized convolutional neural networks to increasing the spatial resolution an image, using its structure as image prior. DIP method doesn't require learning, but produces clean images and gives close results to state-of-the-art methods.

2.1 Uformer

In [16], Wang et al. proposed a transformer based model for RGB smart phone image denoising and trained on the SIDD dataset [1]. Uformer [16] uses the hierarchical encoder-decoder structure and skip connections as shown in Fig. 1a. It replaces the convolutional blocks with transformer blocks in a U-shaped network. To achieve a better capture of local dependencies, Uformer uses a modified transformer block called Locally enhanced Window (LeWin) transformer shown in Fig. 1b [16]. It uses a non-overlapping window-based multi-head self-attention which reduces the computational cost compared to global self-attention. To capture local dependencies, a depth-wise convolutional layer is added between the two feed-forward layers in the transformer block. The architecture also proposes two variations of the skip-connection between encoder and decoder by including a self-attention block which has given competitive results.

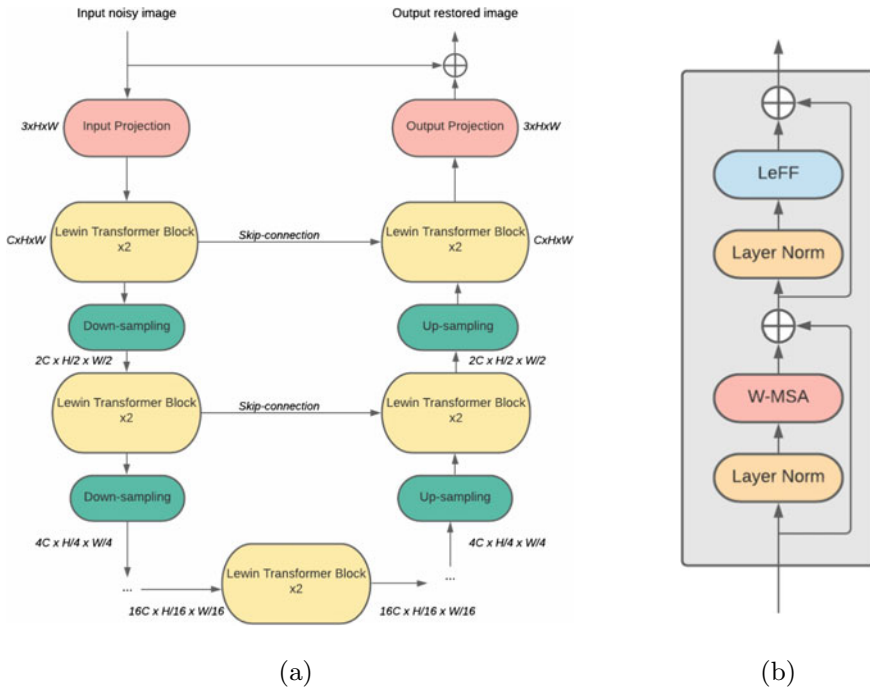


Fig. 1 Uformer [16] **a** the model **b** LeWin transformer block of the Uformer model in (a). Note: input image with height H and width W

2.2 TED-net

TED-net [15] is a convolution free transformer architecture as shown in Fig. 2 developed for LDCT denoising. It uses an encoder–decoder structure, Token-to-token (T2T) blocks [19], and the cyclic shift block [7] to improve the perception of contextual information in the tokens. It further modified the T2T block by adding dilation to the tokenization process. Thus, it can extract the relation over a larger region and refine the contextual information fusion.

3 Experiments

To conduct our experiments, we implemented different deep learning models using Pytorch and trained them on a public medical image dataset. The performance of different methods are evaluated quantitatively as well as qualitatively. The dicom images were pre-processed to convert the pixels into Hounsfield unit and make the slices of uniform thickness.

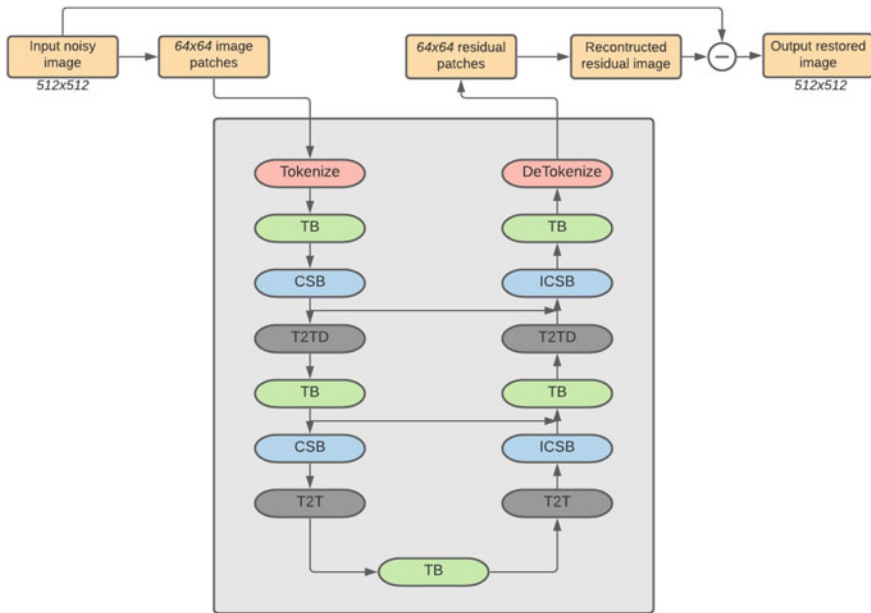


Fig. 2 Block diagram of TED-net [15] model

3.1 Experimental Setup

The 2016 NIH-AAPM-Mayo Clinic LDCT Grand Challenge dataset¹ which is a publicly available dataset provided by The Cancer Imaging Archive(TCIA) has been used for model training and testing. The dataset contains full dose and quarter dose abdominal CT images of 3mm slice thickness of 10 patients in the DICOM format with dimensions of 512x512. We randomly extract image patches of size 32x32, 64x64, 128x128, 256x265 from each original image of size 512x512 in every epoch. Data from patient L506 is used for evaluation and the data which contains 211 images, the other nine patients data is used for the training of the model which contains 2167 images.

We have used Kaggle² with NVIDIA TESLA P100 GPU and a local workstation equipped with NVIDIA QUADRO RTX-8000 GPU to train and test these models. We have trained the model using an embedded dimension of 32 with a window size of 8, the token embedding scheme used is linear. The loss function used is Charbonnier Loss [2, 21] with a small hyper-parameter of 10⁻³. For the estimation of the quantitative metrics, the images have been truncated to the window of -160 to 240 Hounsfield Units (HU).

¹ <https://www.aapm.org/grandchallenge/lowdosect/>.

² <https://www.kaggle.com>.

For comparison, five quality measures were used and they are root mean square error (RMSE), mean structural similarity (Mean SSIM), feature similarity index matrix (FSIM) [24], multi-scale structural similarity (MS-SSIM) [17] and Blind/Referenceless Image Spatial Quality Evaluator (BRISQUE) [9]. RMSE is a measure of change in the pixel quantity of the original image and the reconstructed image. Lower RMSE value indicates that the quality of the reconstructed images are superior. Mean SSIM is a perceptual metric used for computing the similarity of two images. The visual quality of one image is compared with the other. BRISQUE is a no-reference image quality assessment technique, which compares the original image with the default model. Smaller values of BRISQUE indicates a better quality reconstructed image.

3.2 *Qualitative Evaluation*

For the qualitative evaluation of the results, we have chosen an random image from the abdominal scans of patient L506. We have compared it with results from DIP [13], RED-CNN [3], TED-net [15] and Uformer [16]. The Uformer model was modified to work for single-channel dicom images. Also, the model was trained with multiple patch sizes to find the optimal patch size.

The quantitative measures for the predictions on the particular image in Fig. 3 are given in the order (RMSE/SSIM/MS-SSIM/FSIM/BRISQUE). By visually inspecting the residual images given in Fig. 5, it is seen that Uformer has the least amount of error from the ground truth when compared to DIP, RED-CNN and TED-net.

Figure 4 illustrates the comparison of deep image prior, RED-CNN, TED-net and Uformer on a random region of interest highlighted by the red square in Fig. 3a. Also, we displayed the residual images computed for each of the output images with reference to the Ground Truth (GT). Our qualitative evaluation indicates the superior performance of the transformer model over the CNN based methods such as DIP and REDCNN.

3.3 *Quantitative Evaluation*

In this section, we report the results obtained via quantitative evaluation. For that, we compared Uformer [16] model with deep image prior [13], RED-CNN [3], TED-net [15]. The performance metrics used for quantitative analysis are RMSE, BRISQUE, SSIM and FSIM. The mean measurements on the testing dataset for different LDCT denoising algorithms are given in Table 1. We can see that from the results obtained that Uformer [16] performs the best among all the methods under comparison.

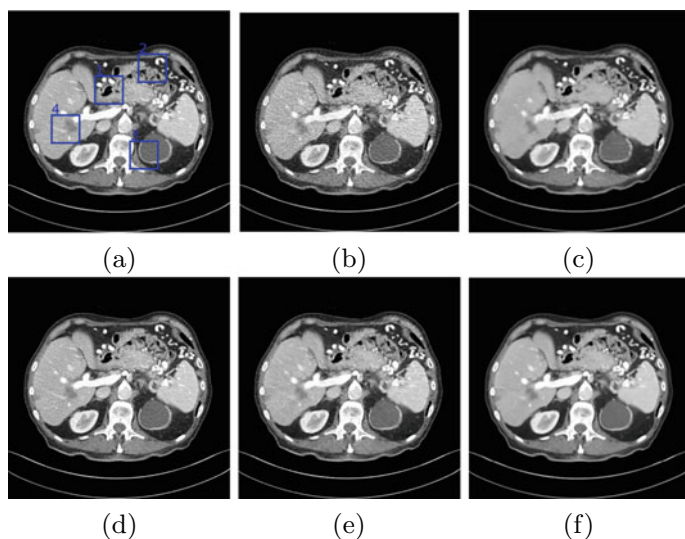


Fig. 3 Qualitative comparison of different image restoration methods on LDCT image. **a** GT, **b** LDCT (23.988/0.788/0.940/0.917/31.916), image restored using **c** DIP [13] (19.613/0.770/0.935/0.908/80.300), **d** RED-CNN [3] (15.073/0.860/0.960/0.944/69.948), **e** TED-net [15] (14.544/0.848/0.963/0.945/82.023), **f** Uformer [16] (13.677/0.850/0.965/0.945/76.580)

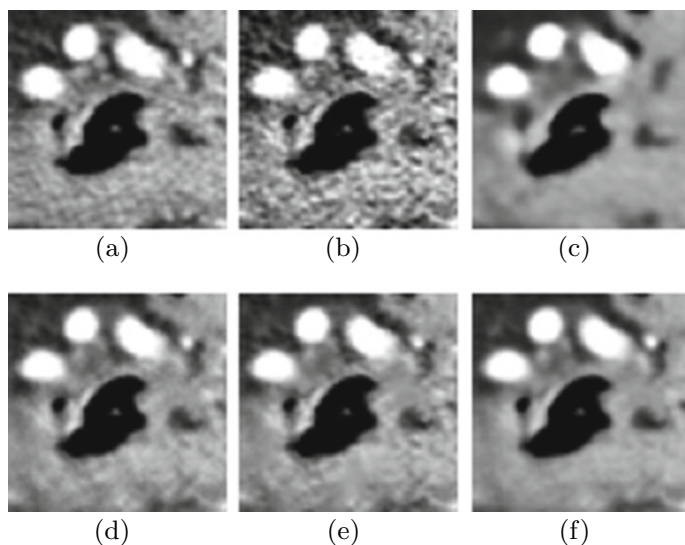


Fig. 4 Comparison of ROI labeled 1 in Fig. 3, **a** GT, **b** LDCT, **c** DIP, **d** TED-net, **e** RED-CNN, **f** Uformer

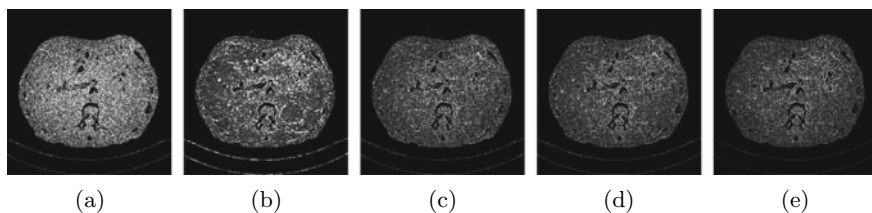


Fig. 5 Residual images with respect to ground truth in Fig. 3a using **a** LDCT, **b** DIP, **c** RED-CNN, **d** TED-net, **e** Uformer

Table 1 Quantitative measures associated with different LDCT denoising algorithms

Methods	RMSE↓	Mean SSIM↑	FSIM↑	BRISQUE↓
LDCT	14.2416	0.8759	0.9548	129.1468
DIP	15.7356	0.8475	0.9174	88.0782
REDCNN	9.8630	0.9053	0.9667	86.7339
TED-net	8.7681	0.9144	0.9687	90.7287
Uformer	8.3226	0.9207	0.9701	92.2257

Also, we computed metrics to measure the performance of the algorithms over the regions of interest marked in Fig. 3a. Uformer had the lowest RMSE and BRISQUE values and the highest SSIM AND FSIM values. Thus, the quantitative analysis in Fig. 7 shows that Uformer is the best performing algorithm compared to TED-net, RED-CNN and DIP.

3.4 Discussion

As discussed above, the results in Figs. 4 and 5 indicate that Uformer [16] performs better for enhancing the output of low-dose CT images. This is supported by the result of the quantitative evaluation in Table 1 too. To validate our results, we have plotted histograms of output images obtained with different methods in Fig. 6. It can be observed from the histograms that the histogram plotted for Uformer [16] (see Fig. 6f) has the best match with the histogram of the GT as shown in Fig. 6a.

Figure 8 shows the results obtained with Uformer [16] on the abdominal image for different patch sizes. Figure 8a, b shows the presence of artifacts for small patch sizes of 32×32 , 64×64 patches. Artifacts are much lower in the case of 128×128 and 256×256 patch sizes as illustrated in Fig. 8c, d, due to less number of patches.

Besides, it can be noticed in Table 2 that the largest patch size has scored the best result in 4 out of the 5 metrics used for evaluation. Moreover, it can be inferred from Fig. 9 that the training converges much faster when a larger patch size is used. This shows that using a larger patch size clearly provides an advantage in the training time as well as for enhancing the overall result.

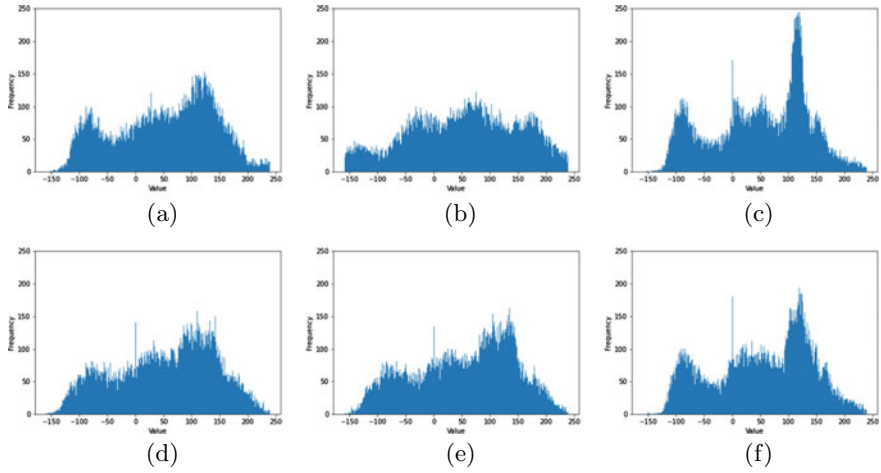


Fig. 6 Histogram of image **a** ground truth, **b** LDCT, **c** DIP [13], **d** RED-CNN [3], **e** TED-net [15], **f** Uformer [16]

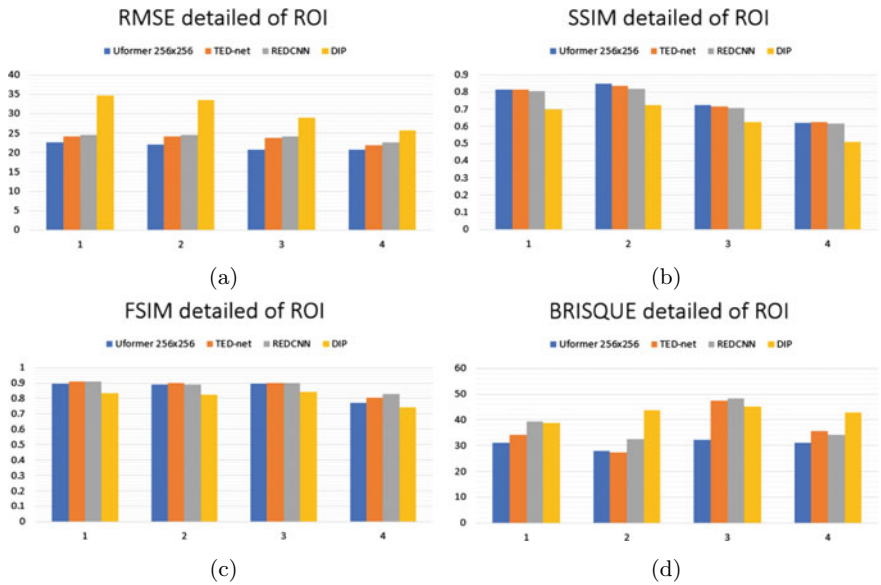


Fig. 7 Comparison of ROIs mentioned in Fig. 3a in terms of **a** RMSE, **b** SSIM, **c** FSIM, **d** BRISQUE

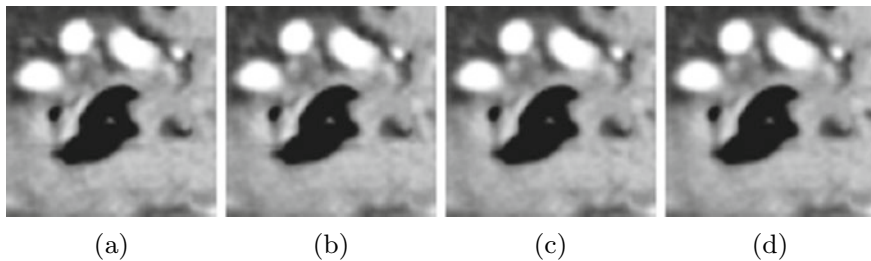


Fig. 8 Zoomed image of the ROI marked in Fig. 3a abdominal image for different input patch sizes in Uformer, **a** 32×32 , **b** 64×64 , **c** 128×128 , **d** 256×256

Fig. 9 Variation of PSNR with epoch during network training for different patch sizes

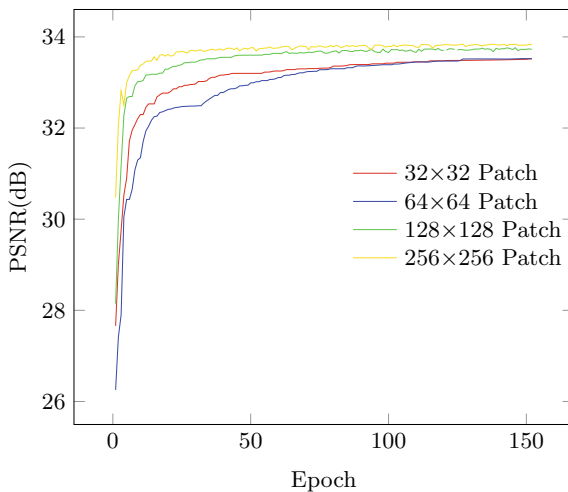


Table 2 Quantitative measures obtained for Uformer with different patch sizes

Patch size	RMSE↓	Mean SSIM ↑	MS-SSIM↑	FSIM↑	BRISQUE↓
LDCT	14.2417	0.87594	0.9694	0.9548	129.1469
32	8.6049	0.9175	0.9821	0.9691	91.3320
64	8.5757	0.9173	0.9822	0.9692	89.8556
128	8.3910	0.9195	0.9828	0.9693	89.6947
256	8.3226	0.9207	0.9830	0.9701	92.2258

4 Conclusion

Uformer [16] is able to capture the image features more accurately with the help of the depth-wise convolution blocks. This leads to faster convergence and fewer epochs for training than TED-net [15]. With the same training time, Uformer [16] has achieved better results than TED-net [15]. It is noted that U-shaped deep learning models are seen to perform well for LDCT denoising tasks. The training time required for Uformer [16] to train the model is very less and is able to outperform TED-net. When compared to DIP, RED-CNN [3] and TED-net [15], Uformer [16] offers the best result. The use of transformer-based models in medical image processing is a field that has not been explored to its full potential. The recent works in the field such as that of Uformer [16] and TED-net [15] show that they have great potential in enhancing LDCT images and its application needs to be explored further.

Acknowledgements This work was supported by the Science and Engineering Research Board (Department of Science and Technology, India) through project funding *SRG/2019/001586*. Also, authors would like to thank Aneta Christopher, National Institute of Technology Calicut, for assistance with the experiments.

Conflict of interest

None of the authors has any financial or personal relationships that could inappropriately influence or bias the content of the paper.

References

1. Abdelhamed A, Lin S, Brown MS (2018) A high-quality denoising dataset for smartphone cameras. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1692–1700
2. Charbonnier P, Blanc-Feraud L, Aubert G, Barlaud M (1994) Two deterministic half-quadratic regularization algorithms for computed imaging. In: Proceedings of 1st international conference on image processing, vol 2. IEEE, pp 168–172
3. Chen H, Zhang Y, Kalra MK, Lin F, Chen Y, Liao P, Zhou J, Wang G (2017) Low-dose ct with a residual encoder-decoder convolutional neural network. *IEEE Trans Med Imag* 36(12):2524–2535
4. Cheng S, Wang Y, Huang H, Liu D, Fan H, Liu S (2021) Nbnnet: noise basis learning for image denoising with subspace projection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 4896–4906
5. Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S, Uszkoreit J, Houlsby N (2021) An image is worth 16×16 words: transformers for image recognition at scale. In: International conference on learning representations
6. He K, Zhang X, Ren S, Sun X (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
7. Liu Z, Lin Y, Cao Y, Hu H, Wei Y, Zhang Z, Lin S, Guo B (2021) Swin transformer: hierarchical vision transformer using shifted windows. [arXiv:2103.14030](https://arxiv.org/abs/2103.14030)
8. Mathews JD, Forsythe AV, Brady Z, Butler MW, Goergen SK, Byrnes GB, Giles GB, Wallace AB, Anderson PR, Guiver TA et al (2013) Cancer risk in 680 000 people exposed to computed tomography scans in childhood or adolescence: data linkage study of 11 million Australians. *Bmj* 346

9. Mittal A, Moorthy AK, Bovik AC (2011) Blind/referenceless image spatial quality evaluator. In: 2011 conference record of the forty fifth Asilomar conference on signals, systems and computers (ASILOMAR). IEEE, pp 723–727
10. Nishio M, Nagashima C, Hirabayashi S, Ohnishi A, Sasaki K, Sagawa T, Hamada M, Yamashita T (2017) Convolutional auto-encoder for image denoising of ultra-low-dose ct. *Heliyon* 3(8):e00393
11. Smith-Bindman R, Lipson J, Marcus R, Kim K-P, Mahesh M, Gould R, González ABD, Miglioretti DL (2009) Radiation dose associated with common computed tomography examinations and the associated lifetime attributable risk of cancer. *Archives Internal Med* 169(22):2078–2086
12. Srivastava RK, Greff K, Schmidhuber J (2015) Training very deep networks. In: NIPS
13. Ulyanov D, Vedaldi A, Lempitsky V (2018) Deep image prior. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 9446–9454
14. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I (2017) Attention is all you need. In: Advances in neural information processing systems, pp 5998–6008
15. Wang D, Wu Z, Yu H (2021) Ted-net: convolution-free t2t vision transformer-based encoder-decoder dilation network for low-dose ct denoising
16. Wang Z, Cun X, Bao J, Liu J (2021) Uformer: a general u-shaped transformer for image restoration. [arXiv:2106.03106](https://arxiv.org/abs/2106.03106)
17. Wang Z, Simoncelli EP, Bovik AC (2003) Multiscale structural similarity for image quality assessment. In: The thirty-seventh Asilomar conference on signals, systems & computers, vol 2, pp 1398–1402. IEEE
18. Yang Q, Yan P, Zhang Y, Yu H, Shi Y, Mou X, Kalra MK, Zhang Y, Sun L, Wang G (2018) Low-dose ct image denoising using a generative adversarial network with Wasserstein distance and perceptual loss. *IEEE Trans Med Imag* 37(6):1348–1357
19. Yuan L, Chen Y, Wang T, Yu W, Shi Y, Jiang Z, Tay FEH, Feng J, Yan S Appendix for “tokens-to-token vit: training vision transformers from scratch on imagenet”
20. Yue Z, Zhao Q, Zhang L, Meng D (2019) Dual adversarial network: toward real-world noise removal and noise generation. In: European conference on computer vision. Springer, pp 41–58
21. Zamir MW, Arora A, Khan S, Hayat M, Khan FS, Yang M-H, Shao L (2020) Learning enriched features for real image restoration and enhancement. In: Computer vision—ECCV 2020: 16th European conference, Glasgow, UK, August 23–28, 2020, proceedings, Part XXV 16. Springer, pp 492–511
22. Zamir SW, Arora A, Khan SH, Hayat M, Khan FS, Yang M-H, Shao L (2021) Multi-stage progressive image restoration. [arXiv:2102.02808](https://arxiv.org/abs/2102.02808)
23. Zeiler MD, Taylor GW, Fergus R (2011) Adaptive deconvolutional networks for mid and high level feature learning. In: 2011 international conference on computer vision. IEEE, pp 2018–2025
24. Zhang L, Zhang L, Mou X, Zhang D (2011) Fsim: a feature similarity index for image quality assessment. *IEEE Trans Image Process* 20(8):2378–2386, e00393

Vehicle-Type Classification Using Capsule Neural Network



Deepak Mane, Chaitanya Kharche, Shweta Bankar, Swati V. Shinde, and Suraksha Suryawanshi

Abstract Vehicles include cars, bikes, buses, aeroplanes, space shuttles, cycles and many more. Vehicle detection and vehicle-type recognition is a practical application of machine learning concepts and is straight away applicable for different tasks in a surveillance system of the traffic contributing to a smart surveillance. We will broach the functioning of instinctive vehicle-type identification using static image datasets. We have put forward Capsule Neural Networks for the identification of vehicle types by consolidating capsules to overcome drawbacks posed by CNN and its counterparts where CNN pooling loses significant data while modern processes can overcome this still image reconstruction is a tedious task. Capsule Network takes into different consideration delineations of object orientation, possie and relates its approach with inverse graphics making object recognition more efficient and accurate and significantly increasing classification performance.

Keywords Capsule network · Intelligent traffic system · Pattern recognition · Vehicle-type classification

1 Introduction

Vehicle-type Classification has been one of the emerging subjects in the field of Intelligent Transport System. It has an extensive variety of implementations ranging from surveillance monitoring control systems to computing vehicle circulation to intelligent parking administration. This allows concerned administrations to identify and manage several means of transit. Conventional systems employ cameras and

D. Mane (✉) · C. Kharche · S. Bankar
JSPM's Rajarshi Shahu College of Engineering, Pune, Maharashtra 411033, India
e-mail: dtmane@gmail.com

S. V. Shinde
Pimpri Chinchwad College of Engineering, Pune, Maharashtra 411044, India
e-mail: swati.shinde@pccoepune.org

S. Suryawanshi
JSPM's Rajarshi Shahu College of Engineering, Polytechnic, Pune, Maharashtra 411033, India

sensors [1], but as these systems employ camera's we are given the opportunity to employ image-based techniques with garner more attention nowadays. Various techniques have been suggested and employed successfully, namely prototype-build and aspect-build techniques. Prototype-build approach computes the length, height and width specifications and utilizes that to redeem a 3D view of the vehicle's image. Also, aspect-build techniques are based on withdrawing attributes constituting the vehicle aspects to identify the vehicles. While the preponderance of identification process requires side-view vehicle images, we will use images consisting of a frontal view image of a vehicle. A mechanism is put forward that recognizes its correct type. Since several surveillance cameras are fixed at numerous places, it is possible to get front view images of the vehicles. These types of images are not difficult to classify [2]. Several algorithms have been developed such as Oriented Contour-Based Algorithm which is a shape matching algorithm that makes use of arrays of prototype images which are flipped versions of original images [3]. Another being Capsule Networks for Dynamic Routing where the goal was relation extraction in a multi-instance learning environment for entity relations and recognition [4]. In 2018, a SVM-based vehicle recognition model is proposed on single framework, which got the 91% accuracy on BIT vehicle dataset [5]. Initially, deep learning CNN algorithms produced state-of-the-art results in many image-based applications [6–10]. In another case, an IoT-based Sensor was developed for Accident Detection using CNN for detecting Driver Drowsiness, Driver Distraction and Pedestrian Detection [11]. Previously, CFCNN method was used combining the advantages of FHSNN and CCNN [12]. But here we made use of Capsule Networks which aptly overcomes the shortcomings of CNN. Capsule Networks depends largely on modelling the relationships which are hierarchical in apprehension of an image to imitate how man learns. This one is a new perspective embraced by conventional neural networks. One major field where Capsule Networks come out on top is they can distinguish and identify rotational invariance, which CNN could not [1].

From the Fig. 1 images, we can say that our brain can see the image of the car from single angle and still we can evaluate and conclude that these different angled images of cars are actually the same car. Here, Capsule Networks have an advantage over the CNNs. In this paper, BIT vehicle standard dataset considered and its classified vehicles into 5 different categories (Bus, Minivan, SOV, Truck and Sedan) which is represented in Fig. 2.

Organization of the paper flow is as follows: Sect. 2 represented the proposed capsule architecture. An experiment result on the proposed model is represented in Sect. 3. Conclusion with future direction is explained in the last Sect. 4.



Fig. 1 Multiple angle images of a car



Fig. 2 Different vehicle types into which we will classify the data from BIT dataset

2 Capsule Neural Network Architecture

Capsule Network schematic is represented in Fig. 3 and layer-by-layer flowchart of proposed Architecture is represented in Fig. 4. The architecture of the Capsule Neural Network mainly consists of the encoder and the decoder. In it, firstly an image of the vehicle is taken as an input by the encoder and then it graphs to represent the same image into a 16 dimensional vector. This vector has all important information which is crucial to manifest the image. Then, the features are diagnosed by the ConvNet Layer which are later analysed by or capsule network. The lowest capsule layer which is the primary layer contains the 32 distinct capsules each applying results of the preceding one to generate 4D vector as an output. The higher capsule layer which is routed by the primary one gives as a result 16D vectors that has substantiation parameters necessary for object reconstruction. Secondly, the decoder takes this vector from the higher capsule and learns to decipher the parameters included in the vehicle image given to it. Euclidian distance vector is calculated by the decoder. It indicates how close rebuilt feature is in comparison with original one which is trained since beginning. Capsule has the only information which is useful to identify the car. It is an easy-to-understand neural network. It has 3–4 Fully Connected Layers. These networks train to rebuild the given image by reducing squared variance among rebuilt and the image provided earlier.

Capsule Networks are unique in the sense that they overcome rotational invariance, which is an inherent difficulty faced by CNN’s making CNN unable to identify features relative to the original source. The Novelty of Capsule Network lies in that we group the neurons together and train them as a single unit while in a CNN we

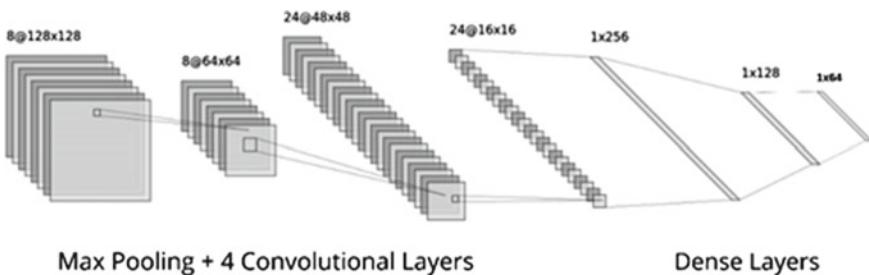


Fig. 3 Capsule network architecture schematic

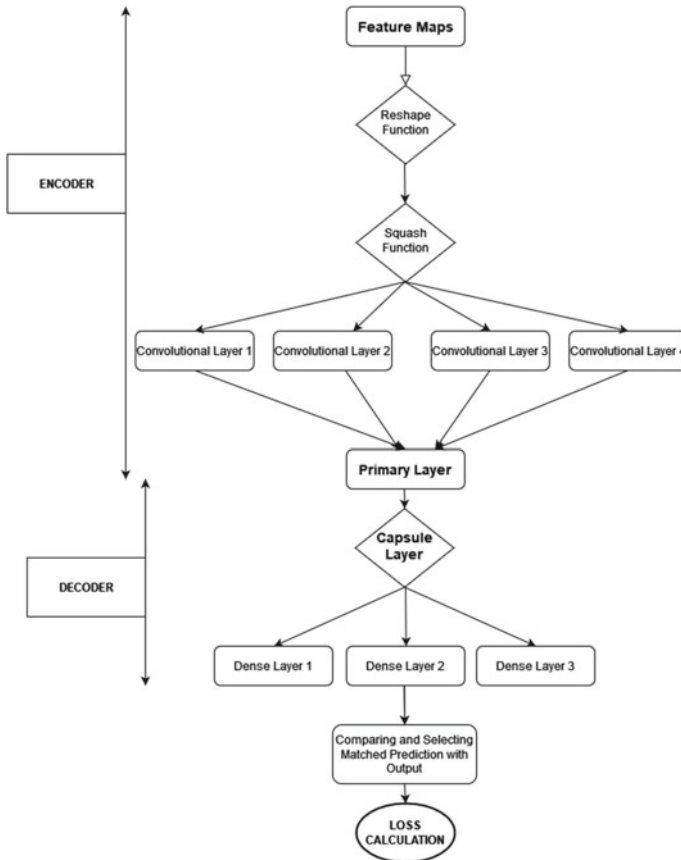


Fig. 4 Layer-by-layer flowchart of proposed architecture

focus on training the neurons individually. Capsule Network are better because it takes into account the features based on their shape and descriptive “vectors” such as position, size and orientation relative to the image they are extracted from, while CNN isolates the features after extraction and computes “individual probabilities” and then adds them for classifying the full image.

2.1 Learning Algorithm for Capsule Neural Network

Step-by-step description of the proposed model is described in following steps.

Step 1: Matrix Multiplication is carried out between the two consecutive layers. The information of the non-linear relationships is ciphered. This information shows possibility of label identifications.

Step II: Scalar Weights is a stage where capsules of the low level modify their weights in accordance to the capsules of high level in order to match with their weights. The capsule of the higher level forms the graph of the distribution of the weight accepting greatest allotment. These convey to one another by dynamic routing.

Step III: Dynamic Routing is the process in which capsule of low level transfers its data to main capsule. All the data is transferred to the most fitting capsule in their according and the one which receives majority data is considered as a parent or main capsule. Weights are then assigned by this parent capsule in same manner. To acknowledge dynamic routing more properly, consider giving capsule network an image of a vehicle. The difficulty may arise in recognizing the car’s roof. Hence, the capsules inspect the image, particularly its constant portion. They arrange the car frame, doors and roof of the car. The conclusion of whether the object is a car or not is done first. Further, it is sent to high level capsules.

Step IV: Vector to vector non-linearity occurs dynamic routing comes to an end. Then, the information is squashed, i.e. information is compressed. It lets us know the possibility of capsule identifying the vehicle.

3 Experimental Results

Performance: We have used the BIT vehicle dataset for our vehicle classification using Capsule Neural Network. This dataset consists of total samples of 9850 which includes bus, truck, SUV, sedan and minivan. The Capsule Neural Network gives fit here best because of encoding the information about components of vectors done by capsules. We have obtained this primary capsule by implementing four convolutional layers. The dataset is categorized as 558 of buses, 822 trucks, 1392 SUVs, 5719 sedans and 1359 minivans. From this entire dataset, 7883 samples were chosen for training model and the remaining 1967 samples were used for testing. The confusion matrix is shown below in the Fig. 3. You can also see the classification report in Table

Table 1 Classification report

	Precision	Recall	f1-score	Support
0	0.92	0.91	0.91	111
1	0.91	0.88	0.89	164
2	0.91	0.95	0.93	278
3	0.98	1.00	0.99	1143
4	0.93	0.86	0.89	271
Accuracy			0.96	1967
Macro avg	0.93	0.92	0.92	1967
Weighted avg	0.96	0.96	0.96	1967

Table 2 Comparative analysis with existing algorithms

Year and reference	Employed architecture	Accuracy (%)
2004 [13]	Rigid structure approach	82.7
2011 [14]	Probabilistic neural network	78.3
2012 [15]	PCA with self-clustering	87.6
2014 [16]	Convolutional neural network (CNN)	88.8
2015 [17]	Laplacian filter and CNN	89.4
2017 [1]	Support vector machine	91.08
2018 [18]	CNN	93.6
2020 [7]	CCNN and FHSNN	94.26
2021	Capsule network architecture	99.4

1 given below. The testing accuracy achieved was 99.4 which is comparatively much better than all previous algorithms mentioned in Table 2.

Step I: Firstly, the libraries which are necessary should be defined. Below given is the snippet of importing dataset and libraries. We have imported BIT vehicle dataset which consists of 9850 samples. It has various images of vehicles of 5 different types.

Step II: Keep a holder of channel as 1. Now start defining the capsule by defining two ConvNet layers from the start and reshape the result from the second ConvNet layer for achieving output of the primary capsule. Do same with the second capsule.

Step III: Define a squash function which will squash vectors.

$$V_j = \frac{\|S_j\|^2}{1 + \|S_j\|^2} \frac{S_j}{\|S_j\|} \tag{1}$$

Step IV: Now, we will define the capsule. Calculate the resultant vectors as both layers one and two are in connection. Using primary capsule project result of first capsule.

Step V: Now perform routing. We perform 2 round routing and define the placeholder for the label. Below is the routing algorithm.

- 1: ROUTING ($\hat{u}_{j|i}$, r, l)
- 2: For all capsule i in layer m and capsule c in layer (m+1): $b_{i,j}$
- 3: for n iterations do
- 4: for all capsule i in layer m: $c_i \leftarrow \text{softmax}(b_i)$ softmax computes Eq. 3
- 5: for all capsule c in layer (m + 1): $s_j \leftarrow \sum_i c_{ij} \hat{u}_{j|i}$
- 6: for all capsule c in layer (m + 1): $v_j \leftarrow \text{squash}(s_j)$ Dsquash computes Eq. 1

- 7: for all capsule i in layer 1 and capsule c in layer $(1 + 1)$: $b_{ij} \leftarrow b_{ij} + \hat{u}_{j|i} \cdot v_j$
- 8: return v_j

Step VI: 2 or more different images can be compared now that we have computed the routing and placeholder for the said label and we will now define the margin loss.

Step VII: Here, we define the values and encode the labels in addition to the output capsule.

We get the matrix of shape after we have defined the norm and reshaped it. As stated earlier, we have computed the loss and the final loss is calculated by taking the means of these

$$L_k = T_k \max(0, m^+ - \|V_k\|)^2 + \lambda(1 - T_k) \max(0, \|V_k\| - m^-)^2 \quad (2)$$

Step VIII: Defining the decoder is the next step after margin loss calculation; here, we have 3 fully connected layers using feature maps and other information for reconstructing the image which will be passed as a vector by the encoder.

Step IX: After the defining the decoder and completing the reconstructing part as well as reconstructed loss. Final loss will be calculated at last by combining Margin loss and Reconstruction loss.

Step X: Now the final step is to train the capsule network and compute the accuracy and loss.

Step XI: Perform Step I to XI for all image.

Figures 5 and 6 represent the results of the accuracy graph for both the testing and training sets. The lighter line is for the training dataset while the darker line is for the testing dataset. Final testing dataset evident from the Graph is shown in Fig. 5. In Fig. 6, the loss graph represents data in a similar manner the lighter line for the training dataset and the darker for the testing dataset. The Loss is minimized further. This indicates the Model works as intended, and the network has optimal fitting.

Confusion Matrix represents the comparison between vehicle actual sample class verses predicted output vehicle class values is presented in Fig. 7. The Data/Images has been classified into 5 classes, and the above Confusion Matrix represents how well the Model has classified the images. It also indicates how some data was misclassified,

Fig. 5 Training vs testing accuracy graph

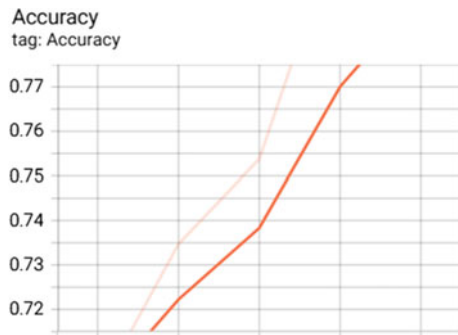


Fig. 6 Training vs testing loss graph

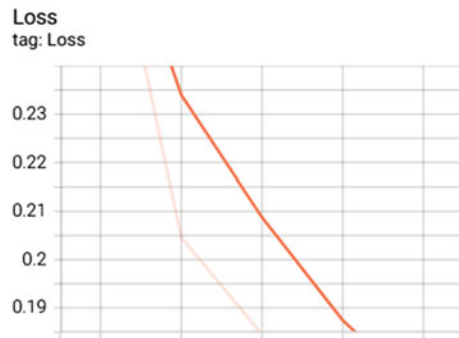


Fig. 7 Confusion matrix

		Predicted Vehicle Class				
		Bus	Minivan	SOV	Sedan	Truck
Actual Vehicle Class	Bus	[101	5	1	0	4]
	Minivan	[6	144	2	3	9]
	SOV	[0	0	265	8	5]
	Sedan	[0	0	3	1140	0]
	Truck	[3	9	19	8	232]

although such errors were kept to a minimum and did not affect the accuracy of the Model.

Classification report shows us the precision and recall and different parameters as shown in Tables 1 and 2 and compares the results achieved by using capsule network with previously proposed architecture along with each of their accuracies.

4 Conclusion

In this paper, presented a Capsule Network which proves to be a proficient object detection technique besting all previously employed techniques while retaining important data and improving efficiency. Overcoming rotational invariance has been one of the biggest positives of capsule network. Due to their complex nature, this is still a relatively newer approach but owing to its accuracy it has shown great promise and will continue to get better. One key difference lies where it differs in its approach from neural network. They will get faster and better and become the standard for image detection and classification problems. BIT Dataset was used here and we achieved an accuracy of 99.4% which is outstandingly better than previous approaches and while other approaches do produce inaccurate results due to unwanted noise, this problem has been overcome delivering superior results.

References

1. Rajasegaran J et al (2019) DeepCaps: going deeper with capsule networks. *IEEE/CVF*
2. Zhang B (2013) Reliable classification of vehicle types based on cascade classifier ensembles. *IEEE Trans Intell Transp Syst* 14(1):322–332
3. Negri P, Clady X, Milgram M, Poulencard R (2006) An oriented-contour point based voting algorithm for vehicle type classification. In: 18th International Conference on Pattern Recognition (ICPR'06), Hong Kong, pp 574–577
4. Zhang N, Deng S, Sun Z, Chen X, Zhang W, Chen H (2018) Attention-based capsule networks with dynamic routing for relation extraction EMNLP
5. Bai S et al (2017) Classify vehicles in traffic scene images with deformable part-based models *Mach. Vis Appl* 29:393–403
6. Mane DT, Kulkarni UV (2017) A survey on supervised convolutional neural network and its major applications. *Int J Rough Sets Data Anal* 4:71–82
7. Mane DT, Kumbharkar PB, Dhotre PS, Borde S (2021) Vehicle-type classification using customized fuzzy convolutional neural network. In: *Data engineering and intelligent computing. Advances in intelligent systems and computing*, vol 1407. Springer, Singapore, pp 419–429
8. Mane DT, Tapdiya R, Shinde SV (2021) Handwritten Marathi numeral recognition using stacked ensemble neural network. *Int J Inf Technol* 13:1993–1999
9. Shinde S, Kulkarni U, Mane D, Sapkal A (2021) Deep learning-based medical image analysis using transfer learning. In: *A computational perspective in healthcare. Studies in computational intelligence*, vol 932. Springer, Singapore pp 19–42
10. Mane DT, Kulkarni UV (2019) A novel fuzzy convolutional neural network for recognition of handwritten Marathi numerals. *Int J High Perform Comput Netw* 15:158–169
11. Alvi U, Khattak MA, Shabir B, Malik AW, Muhammad SR (2020) A comprehensive study on IoT based accident detection systems for smart vehicles. *IEEE Access* 8:122480–122497
12. Mane DT, Kulkarni UV (2018) Modified fuzzy hypersphere neural network for pattern classification using supervised clustering. *Procedia Computer Sci* 143:295–302
13. Petrovic V, Cootes T (2004) Analysis of features for rigid structure vehicle type recognition. *BMVC*, pp 1–10
14. Psyllos A, Anagnostopoulos CN, Kayafas E (2011) Vehicle model recognition from frontal view image measurements. *Comput Stand Interfaces* 33(2):142–151
15. Peng Y, Jin JS, Luo S, Xu M, Cu Y (2012) Vehicle type classification using PCA with selfclustering. In: *IEEE International Conference on Multimedia and Expo Workshops*, pp 384–389
16. Dong Z, Pei M, He Y, Liu T, Dong Y, Jia Y (2014) Vehicle type classification using unsupervised convolutional neural network. In: *22nd International Conference on Pattern Recognition, Stockholm*, pp 172–177
17. Dong Z, Wu Y, Pei M, Jia Y (2015) Vehicle type classification using a semisupervised convolutional neural network. *IEEE Trans Intell Transp Syst* 16(4):2247–2256
18. Roecker MN, Costa YMG, Almeida JLR, Mat-sushita GHG (2018) Automatic vehicle type classification with convolutional neural networks. In: *IEEE 25th International Conference on Systems, Signals and Image Processing (IWSSIP)*, pp 1–5

Trend Prediction of Power Transformers from DGA Data Using Artificial Intelligence Techniques



A. S. Kunju Lekshmi, Deepa S. Kumar, and K. Sabeena Beevi

Abstract Oil-immersed power transformer being a vital component in power system, any type of faults in the transformer can cause severe outage. Therefore, continuous monitoring of its in-service behaviour should be done to improve the performance of the power system by avoiding catastrophic failures. Predicting the dissolved gas level in transformer oil is essential for detecting a developing fault in the transformer. Artificial neural network (ANN) is a powerful tool for application of establishing the power transformers' fault prediction. In this paper, the long short-term memory (LSTM) model was established to predict the future values of DGA data based on the historical gas concentration rates. Then, the accuracy and factors influencing the LSTM model are analysed with classical regression models and machine learning techniques. This selected machine learning approach using LSTM yields a satisfactory result. It can make a reliable prediction and detection of transformer health conditions with respect to generated combustible gas rate.

Keywords Dissolved gas analysis (DGA) · Long short-term memory (LSTM) · Artificial neural network (ANN) · Fault diagnosis · Neural network · Power transformer testing

1 Introduction

Power transformers are one of the most critical equipment in the transmission and distribution sectors of a power system. Therefore, a safe and stable operation is necessary for the regular and reliable operation of the whole power system [1]. Transformers that are constant in operation are affected by various types of stresses, such as electrical and thermal stresses; as a result, it will affect the normal operation

A. S. K. Lekshmi · K. S. Beevi (✉)
Department of EEE, TKM College of Engineering Kollam, Kollam, Kerala, India
e-mail: sabeena3000@tkmce.ac.in

D. S. Kumar
Power Networks Demonstration Centre, University of Strathclyde, Glasgow, Scotland
e-mail: deepa.kumar@ieec.org

of the transformers. These stresses lead to the breakdown in the power system, which might produce financial losses and social harm to the power industry, and inconvenience to the consumers. The liquid insulation in the form of mineral oil is used as a cooling agent [2]. The solid insulation is in the form of impregnated cellulose or paper [3]. The stresses inside the transformer will result in the decomposition of the insulation materials and lead to the generation of specific gaseous products. The liquid insulation, i.e. transformer oil, when heated up due to the faults, decomposes and generates gases like hydrogen (H_2), methane (CH_4), acetylene (C_2H_2), ethylene (C_2H_4), and ethane (C_2H_6) and decomposition of solid insulation, i.e. cellulose will generate carbon dioxide (CO_2) and carbon monoxide (CO) [4].

The generation of these gases would reduce the quality of the transformer oil; as a result, its properties as coolant and insulator get affected, which may result in transformer failures. This issue can be prevented by measuring the amount of gas dissolved in the transformer's oil at regular intervals during its operation. Dissolved gas analysis (DGA) has become a widely established diagnostic approach for detecting internal problems in transformers [5]. By evaluating the production of gas concentration in the transformer oil, the approach is particularly sensitive and gives an early sign of impending defects. The composition and rate of the gases formed depend on the types and severity of the faults. Different types of conventional DGA [6] are key gases, other ratios, and graphical representation methods. So by the gas analysis, it will provide information for the evaluation of any fault and increase the chance of finding a necessary maintenance schedule.

Traditional methods are not always easy to carry out because of the variable nature of the gas concentration and the operational nature of the power transformer. Even though the reliability of the DGA methods is limited due to several factors such as lack of expert knowledge about the diffusion process of gases and sampling of service oil and also due to several complex procedural steps in oil chromatography, there is great chance where the gases may be escaped during sampling, which will lead to reduced accuracy of the DGA method to detect the correct type of fault [7–9]. Thus, forecasting the gas concentration rate is vital for early detection of the incipient failure in the transformer by sending early warning signals to maintainers about the status of the transformer.

In earlier times, a traditional regression method, ARIMA is used to predict time series data due to its high reliability and simplicity structure. But the performance of the model has been limited in forecasting due to certain drawbacks such as less accuracy for short-term forecasting and high error values. Recently, various types of artificial intelligence techniques have been used to implement a time series prediction model and acquired good results, such as artificial neural network (ANN) [10], Recurrent Neural Network (RNN) and long short-term memory (LSTM). However, all these classical regressions and AI methods possess certain drawbacks, i.e. training speed of ANN is slow, prediction accuracy is low, it is easy to fall into the local minimum point, and a large number of training samples are needed in training [11]. A Recurrent Neural Network (RNN) could be a form of Neural Network within which the output of every layer from the previous time step is given as an input to the present time step. LSTM stands for long-term memory (LSTM); it is a suitable

method for dealing with statistical prediction and time series data. It is a specific generic form of RNN, where RNN is a standard cyclic neural network.

The essential theory of the use of dissolved gas analysis is introduced in Section II followed by general introduction. The Duval triangle technique is depicted in Section III. The various time series forecasting techniques are discussed in Section IV. The proposed solution is presented in Section V. The experiment and its results are discussed in Section VI. Its findings were confirmed by comparing them to other models, and Section VII provides the conclusion of the work.

2 Application of Dissolved Gas Analysis

Some of the potential faults in a transformer are over-heating, partial discharge, and arcing. They produce a range of gases. The concentrations and composition rates of these gases can be used to identify and estimate the type and the severity of the faults.

Table 1 shows the fault indication with respect to generated gas concentration due to localized material breakdown of the insulation system under high voltage stresses. D1 is the low-energy arcing discharge that induces turn perforation or formation of carbon particles in the oil, D2 is the discharges in paper or oil insulation, leading to the massive formation of carbon particles, and in some cases, this can result in instrument tripping. DT is the intermediate fault, with a mix of thermal and low-energy electrical faults. T1 is the thermal faults in oil or paper below 300 °C that turns the paper brown in colour. T2 is the thermal faults in oil or paper insulation between 300 °C and 700 °C, leading to carbonization of paper. T3 is the thermal faults in oil, and in paper, above 700 °C leads to carbonization of oil. The catastrophic failure will lead to a power failure, and the power system will be damaged, which will bring huge economic loss and social harm.

The traditional method is not always an easy task due to the variability of gas data and the operational nature of the transformer. Due to the complex operation of

Table 1 Type of fault indication with respect to generated gas rate

Fault	Gas concentration rate in percentage value					
	H ₂	CH ₄	C ₂ H ₄	C ₂ H ₆	C ₂ H ₂	CO/CO ₂
PD	85	13	1	1	0.1	0.2
D1	> 30	6	< 20	3	80	< 0.01
D2	60	5	3	2	30	< 0.01
T1	2	16	19	63	0.01	< 0.01
T2	> 2	16	63	19	0.01	< 0.01
T3	70	8	20	10	0.2	< 0.01
DT	6.7	1.2	< 0.01	< 0.01	< 0.01	92

the oil chromatography, the gases may be escaped during sampling. Prediction of gas concentration in oil is vital for early incipient fault detection of a transformer by sending early warning signals.

3 Duval Triangle Method

The Duval triangle method was advanced in the early 1970s which is demonstrated in Fig. 1. This method is based on the usage of three gas concentration, such as methane (CH₄), ethane (C₂H₄), and acetylene (C₂H₂), with respect to the growing energy levels of gas generation rate. Generally, there are two major types of incipient or an inner fault occurs in the power transformers which are thermal and electrical.

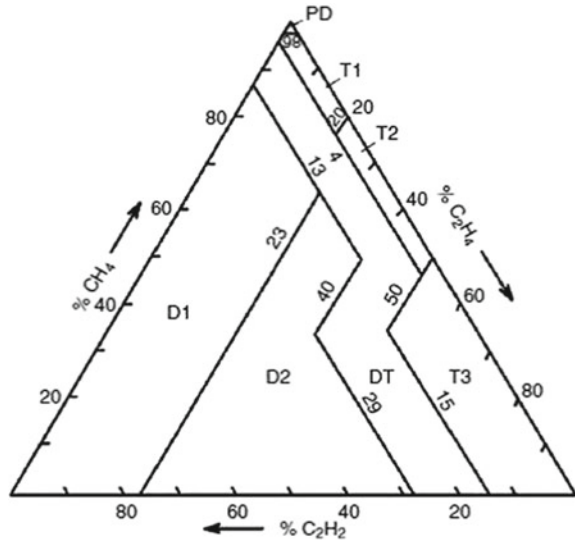
$$\%CH_4 = \frac{100 * x}{(x + y + z)}, \text{ where } x = [CH_4] \text{ in ppm} \tag{1}$$

$$\%C_2H_4 = \frac{100 * y}{(x + y + z)}, \text{ where } y = [C_2H_4] \text{ in ppm} \tag{2}$$

$$\%C_2H_2 = \frac{100 * z}{(x + y + z)}, \text{ where } z = [C_2H_2] \text{ in ppm} \tag{3}$$

The three sides of the triangle illustrated in Fig. 1 are denoted in triangular coordinates as x, y, and z from 0 to 100% for each gas act for the relative proportions of CH₄, C₂H₄, and C₂H₂ concentration in ppm, respectively. To find the transformer state or

Fig. 1 Duval triangle method



diagnose which kind of fault was materialized from Duval Triangle method, firstly calculate the percentage rate of each three gas from its concentration in ppm using Eqs. 1–3. The value of parameters such as x , y , and z should always be in between 0 and 100%, and the sum total of these values should be equal to 100%. Then take out the lines for %CH₄ value aligned to the C₂H₂ line, %C₂H₄ value aligned to the CH₄ line and %C₂H₂ value aligned to the CH₄ line on the specially designed graphical plot. Then take out intersection of all three lines for its percentage rate would inform the type of fault responsible for the DGA data results in the transformer.

4 Model Description

One of the most generally used predictive analytics models is the forecast model which compact the estimating numeric value for new data depend on training from historical data. This section demonstrated the structures of classical regression model, ARIMA, and artificial intelligence techniques such as ANN, RNN, and LSTM model. All these commonly used prediction models such as the classical regression model and advanced model Artificial Intelligence techniques possess their own advantages and disadvantages. And depending on the kind of problems to be solved and by proper selection of appropriate models, we can efficiently use these models effectively.

4.1 ARIMA

ARIMA is one of the foremost and generally used time series models, and it stands for Autoregressive Integrated Moving Average [12] which is applied to forecast data that happens over a period of time. This model is used to forecast future data in a time series with better understanding of the past data where a time series is a pattern where a metric is denoted over intermittent time intervals. The prediction for a given time series data is based on its past values, i.e. its own lags and the lagged forecast misimpression. An ARIMA model is generally featured by using three terms such as p , d , and q . The term ‘ p ’ is denoted as order of the AR term, ‘ q ’ is denoted as the order of the MA term and ‘ d ’ is represented as the number of differencing that required to make the time series stationary.

The term ‘AR’ represented for auto-regression, which shows an attenuation that involves regression on its own lagged values, i.e. it predicts future values based on a linear combination of past values based on below equation:

$$Y_t = \alpha + \beta_1 * Y_{t-1} + \beta_2 * Y_{t-2} + \dots + \beta_p * Y_{t-p} + \varepsilon_t \quad (4)$$

In above Eq. (1), Y_t represents function of the lags of forecasted values, α is a constant value or the intercept term, β_1 is the coefficient of first lag value, and Y_{t-1} is the first lag value of the time series data and is the error term. The term ‘I’ denotes

integrated part, which means it gives the difference between current static data values and its previous values. And the term ‘MA’ denotes the moving average part, which shows the correlation between an observed value and a residual error of lagged values from a moving average applied to previous observations values.

$$Y_t = \alpha + \varepsilon_t + \vartheta_1 * \varepsilon_{t-1} + \vartheta_2 * \varepsilon_{t-2} + \dots + \vartheta_p * \varepsilon_{t-p} \tag{5}$$

In short, an ARIMA model is a type of regression method where the time series data was firstly differenced at once or twice to make it stationary for further processing and combine the AR and the MA using above Eqs. (4) and (5) terms to generate the forecasted values to give the overall mathematical equation.

So, in general form we can denote ARIMA model using equation as:

Forecasted (Y_t) = Constant + Linear combination Lags of Y (for lags) + Linear Combination of Lagged forecast errors (for q lags).

4.2 Neural Network

A Neural Network is a network of neurons which are in layers as shown in Fig. 2. Multilayer feed-forward back-propagation algorithm is selected as the network configuration. The inputs form the input layer, outputs form the output layer, and an in-between layer of hidden layer that have hidden neurons where every type of calculations and character extractions carried out. Outputs are taken by a linear combination of the inputs that we give to the input layer. This topology structure of network has a fundamental role in the performance of the neural network. The essential idea of neural network depend on fault diagnosis is nonlinear mapping. It is surmise that the link between the input vector ‘X’ and the output vector ‘Y’ are

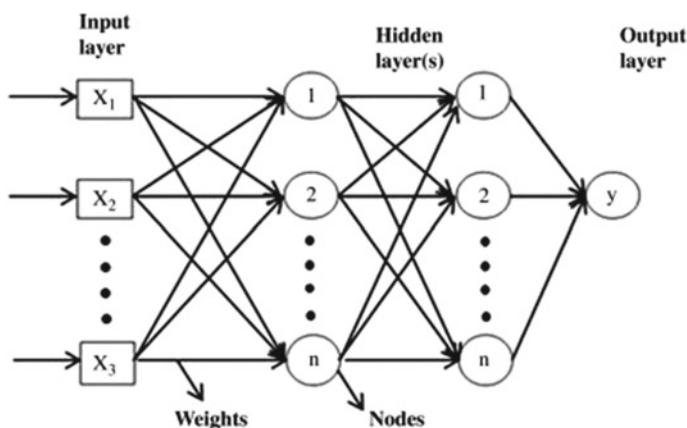


Fig. 2 NN architecture

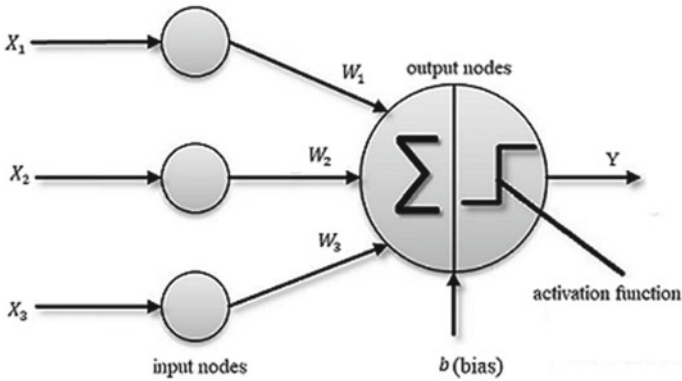


Fig. 3 Neuron representation in a neural network

defined by the physical nature of the problem and this link can be represented by a number of input–output pairs to the input and output layers, respectively.

Each neuron on each layer in the neural network can be represented as shown in Fig. 3, and the equation that relate the mathematical behaviour of the neuron is given by:

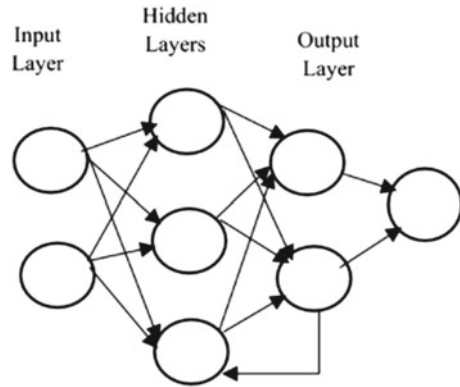
$$y = g\left(\sum_{i=1}^n w_i * x_i + b\right) \tag{6}$$

where ‘*n*’ represents the number of inputs of the neuron in input layer, ‘*W_i*’ represent the weight associated with the *i*th input, ‘*b*’ represents the threshold or bias value associated to each of the neuron, *x_i* denotes the *i*th input of the neuron, *g(x)* denotes the activation function the neuron, and ‘*Y*’ is the output of the neuron.

4.3 Recurrent Neural Network (RNN)

Recurrent Neural Network (RNN) is a type of Neural Network in which the output of each layer from the previous step is fed as an input to the current step, as shown in Fig. 4 [13]. In conventional NNs, all the inputs and outputs are independent of each other, but in cases like when it is required to predict the subsequent data of particular sequences, the previous value of knowledge is required, and hence, there is a desire to recollect the previous data values [14]. Thus, RNN came into existence, which solved this issue with the assistance of an intermittent hidden layer, i.e. RNN provides a really useful way of addressing statistic sequential data that have the correlations between data points that are drawn in the sequence with respect to time intervals. The most important feature of RNN is its hidden state, which remembers

Fig. 4 Recurrent neural network architecture



necessary information from the previous sequence data. The outstanding structural characteristic of RNN is its feedback circuit with time delay function.

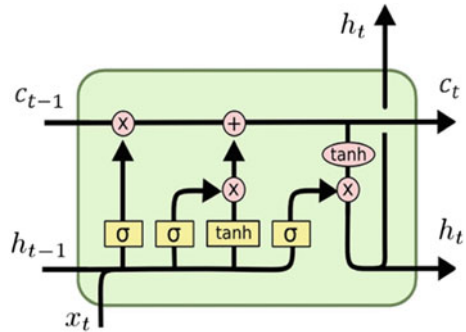
The RNN structure consists of multiple fixed activation function units, one for every time step. Each unit contains an internal state which is named because of the hidden state of the unit. This hidden state provides the past knowledge of the network's information at a given time step. This hidden state is updated in steps in the learning process.

5 Methodology

LSTM stands for long short-term memory, and it is a very good method for dealing the time series prediction [15]. It is a specific generic form of Recurrent Neural Network (RNN), where RNN is a common cyclic neural network as we discussed in the above section.

Thus, the most important feature of RNN structure is its hidden state, which remembers some information or past states about the sequence. LSTM, in its core function, retains the input data information that has already been passed through it at the time of training using the hidden state [16]. Structurally, it is a connection of many RNN units or memory blocks and each block consist of one recurrently connected memory cell and its three gate units as shown in Fig. 5, where these gated cells can act on the received input data by throwing or keeping the information based on the importance of the data element to solve the vanishing gradient problem, which commonly occurred in basic RNN models. Each RNN network module contains one or more self-correlated memory cells and three-door units such as input gate, forget gate, and output gate. The forget gate decides which information should be forgotten or needs attention. The current input $x(t)$ and hidden state $h(t-1)$ values were passed through the sigmoid function along with their respective weight and bias matrix values. The Sigmoid function produces output values in between 0 and 1. The output from the forget gate is given by Eq. (7), and it concludes whether the part of the old

Fig. 5 LSTM-RNN cell architecture



output is necessary or not (by giving the output closer to 1). This value of $f(t)$ will later be used by the cell for further multiplication with the previous cell state values as in Eq. (8).

$$f_t = \sigma(W_f * [h_{t-1}, x_t] + b_f) \tag{7}$$

$$C_t^0 = f_t * C_{t-1} \tag{8}$$

Input gate $i(t)$ determines which cell should be updated and the cell state at current time node as in Eq. (9). Cell state $C(t)$ gives what to keep or thrown from memory with help of both forget and input gate, the cell state at current time node as in Eq. (9). Cell state $C(t)$ gives what to keep or thrown from memory with help of both forget and input gate.

$$i_t = \sigma(W_i * [h_{t-1}, x_t] + b_i) \tag{9}$$

$$C_t = \tanh(W_f * [h_{t-1}, x_t] + b_c) \tag{10}$$

$$C_t = C_t^0 + i_t * C_t \tag{11}$$

where $f(t)$ represents forget gate, it represents an input gate, $O(t)$ represent output gate, $h(t)$ represents hidden state, $C(t)$ represents cell state, $h(t - 1)$ represents previous hidden state, $C(t - 1)$ represent previous cell state and $x(t)$ represent the current input. $W_f, W_i, W_o,$ and W_c represents the weights matrix corresponding to different gates, and $b_f, b_i, b_o,$ and b_c represents the bias of different state gates.

Finally, the new cell state $C(t)$ and new hidden state $h(t)$ are calculated using Eqs. (11) and (13). The overall working of the LSTM-RNN can be concluded such as the forget gate determines which necessary information from the previous steps is needed. The input gate decides what information should be added or kept from the current step, and the output gates finalize the next hidden state.

6 Results and Discussions

This paper takes the historical oil chromatography dataset collected by 220 kV transformer oil for a period from 31/05/2011 to 20/05/2020 from Power Grid Corporation of India Ltd (PGCIL), Thiruvananthapuram, to perform this experiment. The sampling interval is each month. A total number of 3285 data samples are used for each gas. To evaluate the accuracy and validation of the prediction model proposed in this paper.

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (y_i^* - y_i)^2}{n}} \tag{12}$$

where y_i^* is the predicted or forecasted values, y_i is the actual or real values, and n is the number of data samples used for forecasting.

6.1 Prediction Results

In Fig. 6, it shows the time series plot for ethylene gas concentration with actual and forecasted concentration values over months. Here, we forecasted the next five days' gas concentration rate with the available past data values, and by the analysis of this plot, it is clear that the predicted trend of concentration rate has an increasing

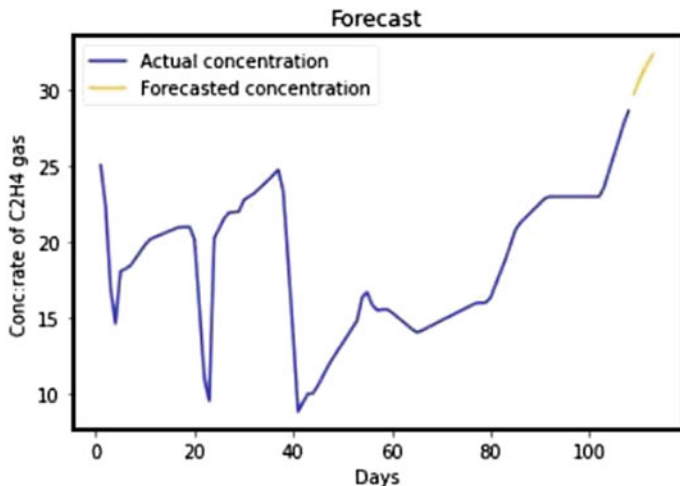


Fig. 6 Time series plot for C₂H₄ gas concentration

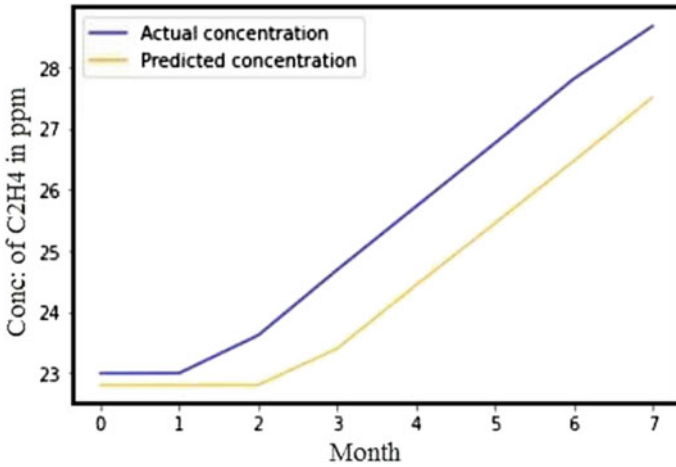


Fig. 7 Predicted versus actual concentration

Table 2 Comparison of actual data with forecasted data

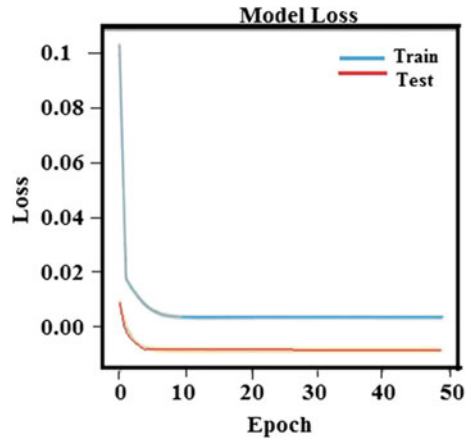
Sl. No.	Type of dataset	Forecasted value
1	Data from oil chromatography	30
2	Forecasted data using ARIMA	26.1
3	Forecasted data using LSTM	26

nature over the period of monthly interval. The comparison graph for the predicted and actual occurred data for ethylene gas is shown below in Fig. 7.

In Table 2, it shows a comparison table of the forecasted data values for unseen or upcoming values using LSTM model with the actual measured data using oil chromatography test. It is clear that the forecasted data using LSTM model is very closer to the data that found out from the DGA method than classical regression method or other machine learning technique used here. In other words, we can say that the deviation level of value between the actual and predicted values is smaller in using LSTM.

In Fig. 8, the curve for learning looks like a good fitting and the learning curve for validation loss shows some noisy behaviour. The root mean square error is 1.05% for ethylene gas. From Table 2, it is clear that the root mean square error value of the LSTM method is lower than the classic regression ARIMA model and Neural Network (NN) model. Therefore, it can be concluded that the use of LSTM to predict power transformers' gas concentration has high stability and reliability.

Fig. 8 Validation curve



7 Conclusion

DGA is a popular and efficient tool for identifying fault types in an oil-filled power transformer. The present work uses deep learning techniques to forecast the DGA data to diagnose the internal abnormalities of the oil-filled power transformer. In this work, we evaluated and compared the performance of LSTM deep learning methodology with classical forecasting methods such as ARIMA and Neural Network models for time series DGA data prediction. The performance is evaluated on a benchmark dataset collected from PGCIL, Thiruvananthapuram, through an oil chromatography test using accuracy and error measures. The results showed that unidirectional LSTM networks produced a better performance for predicting DGA data. The results also showed the superiority of deep learning methodology over shallow neural networks. Overall, LSTM networks attained better performance and convergence for both short- and long-term predictions. This proposed method could be applied to identify the inner faults in power transformers. The experimental analysis and results show the potential of the proposed model to forecast and detect the incipient fault of the power transformer.

References

1. IEC Standard 61181 (2012) Electrical mineralization—the use of gas analysis (DGA) in factory testing of electrical equipment and AMD1
2. IEEE Standard C57.130 (2015) IEEE guide to the use of disposable gas system used in factory temperature testing in testing transformers in blood and oil transformers
3. Xu H (2012) Study on transformer oil dissolves gas online monitoring and fault diagnosis method. In: IEEE International Conference on Condition Monitoring and Diagnosis, pp 593–596

4. Ding H, Heywood R, Lapworth J, Josebury R, Roxborough A, McCulloch E (2017) The experience of making molten gas in transformer oil to detect incoming errors. In: IEEE 19th International Conference on Dielectric Liquids (ICDL), pp 1–5
5. Golarz J (2016) Understanding dissolve gas analysis (DGA) techniques and definitions. In: IEEE/PES Transmission and Distribution Conference and Exposition (T&D), pp 1–5
6. Haema J, Phadunghin R (2012) Power transformer condition evaluation by the analysis of DGA methods. In: Asia-Pacific Power and Energy Engineering Conference, pp 1–4
7. Han S, Yi D, Kang KK (2006) Development of oil immersed transformer management technologies by using dissolve gas analysis. In: IEEE 8th International Conference on Properties & applications of Dielectric Materials, pp 683–686
8. Hettiwatte SN, Fonseka HA (2012) Analysis and interpretation of dissolved gases in transformer oil: A case study. In: IEEE International Conference on Condition Monitoring and Diagnosis, pp 35–38
9. Londo L, Çelo M, Bualoti R (2015) Assessment of transformer condition using the improve key gas methods. *Int J Engineering Res Techn* 4(5):385–391
10. Iyswaryaa Annapoorani K, Umamaheswari DB (2012) Fault prediction based on dissolving gas concentration from insulating oil in power transformer using Neural Network. In: IEEE 10th International Conference on the Properties and Applications of Dielectric Materials, pp 1–4, July
11. da Silva N, Imamura MM, de Souza AN (2013) The use of neural networks in the analysis of dissolved gases in the protection of oil used in transformers. In: IEEE International Conference on Systems, Man and Cybernetics, vol 4, pp 2643–2648
12. Hirata T, Kuremoto T, Obayashi M, Mabu S, Kobayashi K (2015) Time series prediction using DBN and ARIMA. In: International Conference on Computer Application Technologies, pp 24–29
13. Su L, Yang Y, Xing H, Wei B, Ling P, Lu W (2019) On machine learning approaches towards dissolve gases analysis of power transformer oil chromatography. In: IEEE Symposium Series on Computational Intelligence (11). *SSCI*, pp 1743–1750
14. Mnyanghwalo D, Kundaeli H, Kalinga E, Ndyetabura H (2020) Faults detection and classification in electrical secondary distribution network using recurrent neural network. In: 6th IEEE International Energy Conference (ENERGYCon), pp 958–966
15. Aparna A, Sabeena Beevi K, Benson S, Asif Ali A, Dilshad A (2020) A modified CNN for detection of faults during power swing in transmission lines. *DSK 6th Biennial IEEE International Conference on Emerging Trends in Engineering, Science and Technology (ICETEST)*
16. Ghanbari R, Borna K (2021) Multivariate time-series prediction using LSTM neural networks. In: 26th International Conference, Computer Society of Iran (CSICC), pp 1–5

Artificial Intelligence and Machine Learning in the Context of E-commerce: A Literature Review



Richard Fedorko , Štefan Kráľ , and Igor Fedorko 

Abstract Artificial intelligence is now becoming increasingly popular as the development of information and communication technologies continues to advance. The goal of companies operating in e-commerce is to influence customer behavior and prompt customers to purchase selected products. It is the use of innovative tools of artificial intelligence in the field of e-commerce that may seem like the right step forward. The paper aims to provide an overview of the issues of e-commerce, machine learning, artificial intelligence, and their benefits. The aim is also to appraise the importance of artificial intelligence, machine learning, and their purpose in the context of e-commerce based on available studies on this issue.

Keywords Artificial intelligence · Machine learning · E-commerce · Electronic commerce

1 E-commerce, Artificial Intelligence, and Machine Learning

The modern information age has brought along new possibilities, software and technological innovations that could be used in marketing and shopping. New technologies lead companies to think creatively and help raise the quality, efficiency, and cost-effectiveness of services offered by organizations [1, 2]. Current trends in creativity are mainly based on the development of information and communication technologies, which extensively affect the advancement of the business environment [3]. One of the sectors where the digital transition and the significance of electronic commerce

R. Fedorko (✉) · Š. Kráľ · I. Fedorko
Faculty of Management and Business, University of Presov, Prešov, Slovakia
e-mail: richard.fedorko@unipo.sk

Š. Kráľ
e-mail: stefan.kral@smail.unipo.sk

I. Fedorko
e-mail: igor.fedorko@unipo.sk

is principally pronounced is retail, where digital tools, for instance, websites largely replace or complement traditional commerce [4].

1.1 Meaning of E-commerce

The Internet advancement and the progress of digital technologies have led to modification of consumer habits. Nowadays, people are more and more using e-commerce to shop instead of visiting brick-and-mortar stores in person [5].

Electronic commerce/e-commerce takes place in the online environment of the Internet, with the Internet being considered a single platform that connects the seller and the buyer [6]. According to Ullman [7], e-commerce includes all commercial transactions carried out online. This can be described as digitally empowered business transactions between companies and their customers [8].

Online purchasing is thus a type of electronic commerce in which consumers browse the Internet and buy products or services directly from a retailer’s website [9]. Online shopping is becoming a regular, everyday activity. In recent years, this way of shopping has become very popular, especially because people appreciate the convenience and ease of making a purchase from the comfort of their own home [10]. B2C e-commerce is the most common type of e-commerce. Within it, merchants try to attract and win individual customers. Over the next 5 years, B2C e-commerce is projected to grow by 20% annually [11]. Online retailing is currently the fastest-growing retail channel with the fastest-growing customer base and an ever-expanding range of goods [12].

The advantages e-commerce brings to online retailers and customers are shown in Fig. 1 [13, 14].

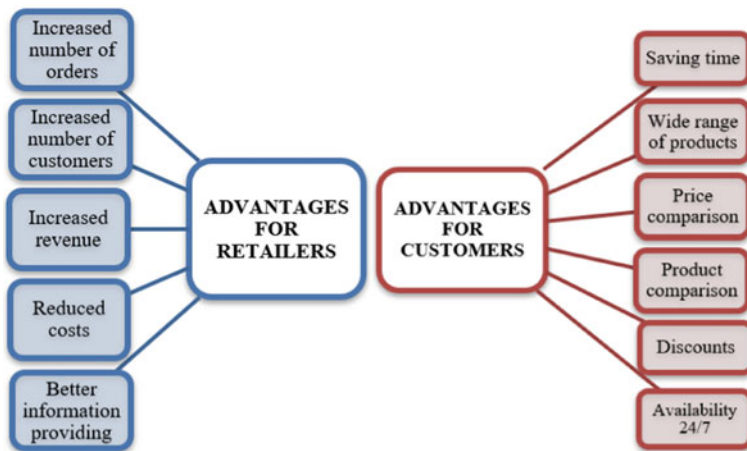


Fig. 1 Advantages of e-commerce for online retailers and customers

E-commerce is the result of economic, scientific, technological, and cultural development. The progress of e-commerce is changing the way businesses do business as well as consumer behavior and is making a significant contribution to the progress of the global economy [15]. As the importance of the Internet grows today, the adoption of online commerce is a source of competitive advantage. It is necessary to monitor trends in online shopping and changes in shopping behavior and preferences of online shoppers in a rapidly evolving Internet environment. As a result, online retailers can tailor their range of products and services to consumers' shopping habits and meet their needs [16]. The implementation of e-commerce in the company requires a fundamental modification of current business models and business activities, as well as the implementation and use of sophisticated digital technologies [17].

1.2 Meaning of Artificial Intelligence and Machine Learning

Innovations and more efficient application of technologies have led to the creation of intelligent systems that can manage and control business models with limited human involvement [18]. The advancement of artificial intelligence has generated tremendous economic benefits to humanity, to almost all aspects of life, greatly promoted social development, and taken it into a new era [19].

Artificial intelligence is considered a new interdisciplinary technological branch of science that advances theoretical approaches, technologies, and applications for the simulation and expansion of human intelligence [20]. The use of artificial intelligence has been examined in sectors such as health care, business, education, manufacturing, marketing, and financial management [21–23].

Russell [24] summarized the different definitions of artificial intelligence systems within two dimensions. The first dimension is Behavior dimension = Reasoning, which speaks of artificial intelligence as a system that thinks and acts like human. The second dimension is Rationality dimension = Human performance, which speaks of artificial intelligence as a system that thinks and acts rationally.

Artificial intelligence can perform mental work by simulating and augmenting the intelligence of humans. Artificial intelligence systems can work autonomously and to some extent adapt their behavior based on the analysis of the previous steps [15, 18].

Key generated artificial intelligence capabilities include prediction, planning, and learning. More significantly, however, artificial intelligence abilities are not independent—they mutually cooperate with human skills to generate business values in view of efficiency and effectiveness [25]. Artificial intelligence has the potential to transcend human intellectual and physical abilities, offering opportunities to increase productivity and performance. In order for artificial intelligence to work best in the company, it is necessary to integrate it with existing business processes [26, 27]. Artificial intelligence can reliably perform computer tasks and automate repetitive learning. It is also able to analyze large amounts of data more accurately and get

the most out of them. Artificial intelligence adapts through progressive learning algorithms which become a classifier or predictor [28].

Artificial intelligence systems should have capabilities such as information processing for natural-sounding communication, the ability to store and present information, automatic reasoning—the use of stored data to respond to questions and reach new inferences, machine learning to cope with new conditions, and to discover new patterns of behavior [29].

At the heart of artificial intelligence is intelligent technology, which develops intelligent tools—a process similar to human intellectual work. These tools are able to react immediately upon receipt of control commands [15]. Other means of artificial intelligence include expert systems, decision support systems, or machine learning [30]. One of the means used by artificial intelligence is fuzzy logic. It is a tool that can display human action, process, and interpret information and knowledge as if it were performed by a human. The use of fuzzy logic in conjunction with artificial intelligence enables better planning, objective professional evaluation and risk assessment, rational decision-making and management. It can also help eliminate errors associated with human factor failure [31, 32].

Artificial intelligence has an irreplaceable role in the development of society and has brought revolutionary results in the form of the advantages as shown in Fig. 2 [27, 33].

The dynamic advancement of artificial intelligence in terms of its capability to decide in complex and unpredictable situations in the production in the last few years has been made possible mainly due to increased investments in innovative technologies and the accessibility of huge volumes of data [34]. Studies estimate that by 2030, it is likely that 70% of businesses will use some form of artificial intelligence technology in their business processes and activities. Adaptation and introduction of artificial intelligence in companies in line with advanced technologies are projected to gain in popularity [35].

Fig. 2 Advantages of artificial intelligence



1.3 Machine Learning as a Subcategory of Artificial Intelligence

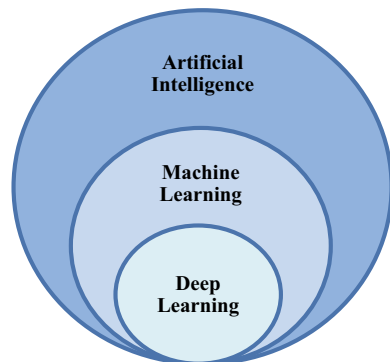
In order for artificial intelligence systems to advance, they must be able to learn. Machine learning helps a lot here. Machine learning is a combination of statistical and mathematical models that intelligent systems use to perform their tasks. They will do so without the need for a thorough input [36].

The history of machine learning dates way back. In 1952, a computer program was created that could play a game of draft. In 1958, the first artificial neural network was built. It was a machine learning algorithm which was based on the general neuron structure of the human brain. Machine learning evolved over the next decades. In 1997, IBM created the Deep Blue computer, which successfully defeated the world chess champion [37].

Machine learning is a subcategory of artificial intelligence (see Fig. 3). It can be described as a technique for proposing a series of sequential steps to solve a particular problem (algorithm) that is automatically optimized based on experience with reduced or no human involvement. These techniques can be used to find patterns in large data sets from as diverse and innovative sources as possible [38]. Machine learning algorithms are a group of computer algorithms that apply statistical tools to learn from instances, called “Training data,” in an effort to forecast new data, allowing the algorithm to better capture difficult patterns in the data [39, 40]. Machine learning algorithms are designed in a way the machine attempts to learn on its own without being explicitly coded for every individual instruction and relying on human experts [41, 43]. Algorithms “learn” primarily by recognizing patterns based on as much data as possible. They then analyze the information and find the trends that are involved in it. Finally, the algorithm is “intelligent” enough to use what it has learned in the new data sets [37].

While artificial intelligence allows machines to imitate behavior of people, machine learning is a subcategory that applies statistical tools to allow machines to enhance the experience. Deep learning is a subcategory of machine learning and allows the calculation of a multilayer neural network. It works on the basis of layers

Fig. 3 Artificial intelligence versus machine learning versus deep learning



and uses an artificial neural network, a design that is inspired by a biological neural network. The human brain typically processes the received information and attempts to recognize it based on the past information that the brain has recorded. The brain does this by tagging and classifying information into different categories in a fraction of a millisecond. Deep learning just takes data connections among all the artificial neurons and modifies them following a data pattern. If the data size is too large, more neurons are needed. It automatically offers multi-level abstraction learning, allowing the system to learn to map difficult features without dependence on any particular algorithm [41].

2 Role of Artificial Intelligence and Machine Learning in E-commerce

Today, e-commerce is one of the industries that use artificial intelligence to the best of its ability by creating a huge customer base, trying to understand the needs of customers, doing real-time research, coming up with final solutions to problems, and performing many other activities [35].

Table 1 outlines selected studies that focus on the role of artificial intelligence and machine learning in e-commerce. It contains authors of the study, objective of the study, study theme, and main findings.

2.1 Artificial Intelligence Use in E-commerce

Digital commerce has evolved over the years, and as customer demand and requirements continue to change, the involvement of artificial intelligence is now required. Artificial intelligence technology is one of the driving forces of innovation in e-commerce. By introducing artificial intelligence, online stores can not only streamline the shopping process itself, make it easier for customers and avoid complaints or returns, but also save the cost of running an e-shop. This technology offers a number of benefits that allow owners to increase their chances of success in the highly competitive environment of a growing number of e-shops [42].

Artificial intelligence can take several forms. When it comes to software artificial intelligence, AI can take the form of so-called virtual assistants, image analysis software, text analysis software, search engines, and speech and face recognition systems. When it comes to an artificial intelligence built into material devices, AI can take the form of robots, self-propelled cars, or drones [18].

Multicriterial Text Analysis and Machine Translators

Advanced text analysis is a tool that analyzes and processes the texts of all publicly available user reviews. In practice, this means that after entering the product name

Table 1 Selected studies about artificial intelligence and machine learning in e-commerce

Reference	Objective	Study theme	Findings
Song et al. [15]	To analyze the present situation of the use of artificial intelligence technology in the field of e-commerce	Artificial intelligence in e-commerce	Artificial intelligence will be an essential driver of e-commerce transformation
Soni [35]	To analyze the use of artificial intelligence in e-commerce business	Artificial intelligence in e-commerce	Artificial intelligence can help people excel in their high sales and customer relationships in the e-commerce business
Kar [44]	To analyze specific integration of chatbots in the IoT systems	Applying chatbots to the Internet of things	Chatbots can help consumers find products, check order status, compare products and connect them to the customer support service staff
Kumar [45]	To present impact of artificial intelligence in e-commerce	Artificial intelligence in e-commerce	Integration of artificial intelligence in e-commerce companies can help create sophisticated solutions and will impact transactions, customer retention, satisfaction, and efficiency
Pallathadka et al. [46]	To discuss machine learning and artificial intelligence applications in e-commerce, corporate management, and finance	Artificial intelligence in business management, e-commerce, and finance	The most common applications include sales growth, profit maximization, sales forecasting, inventory management, security, fraud detection, and portfolio management
Zhang et al. [55]	To explain how artificial intelligence technology, people, and processes should be managed to successfully generate value	Artificial intelligence in e-commerce	The key artificial intelligence capabilities generated include forecasting, planning, and learning. They are not independent—they cooperate with human skills to create business value

(continued)

Table 1 (continued)

Reference	Objective	Study theme	Findings
Shankar [56]	To present a framework for understanding artificial intelligence, outline applications of it in various aspects of retail and discuss the perspective of artificial intelligence in retailing	Artificial intelligence in retailing	Artificial intelligence will allow retailers to improve their understanding of consumer psychology by approximating to what the consumer brain is thinking at any given moment
Makridakis [64]	To present changes caused by artificial intelligence, contrasts with the industrial revolution, the effect on developed and developing countries, and the dominant firms in artificial intelligence	The potential effects of artificial intelligence on businesses, manufacturing, and commerce	Relationship between artificial intelligence and humans will strengthen, companies will focus on creating values through collaboration between the human workforce and artificial intelligence tools
Luo et al. [65]	To study the impact of artificial intelligence chatbot disclosure on customer purchases	Impact of artificial intelligence on customer purchases	Prior experience with AI induces more customer purchases and is helpful in reducing the negative disclosure effect
Loureiro et al. [68]	To give an insight into the current state of research in the field of artificial intelligence in the business context and propose an agenda for further research	Artificial intelligence in business	Future trends in Artificial intelligence are the development of advanced automated systems, integration of neurostimulators and nanochips into the brain, smart devices connected to artificial intelligence systems
Martínez et al. [51]	To develop machine learning framework for predicting future purchasing	Prediction of online consumer behavior by machine learning	Important characteristics are number of purchases, mean time between purchases, time since last purchase, mean value of purchase

(continued)

Table 1 (continued)

Reference	Objective	Study theme	Findings
Cui et al. [52]	To develop a good data-driven model to forecast the quantity of returned products in the future	Prediction by machine learning	Predicting variables were sales, time, product features, retailer, production process and resources, multi-product order and historical returns. Sales are the most significant variable
Orogun [53]	To develop predicting model of consumer behavior in digital market	Prediction of online consumer behavior by machine learning	Important predicting attributes are invoice number, product code, description of product (item) name, quantity of purchase of each item per transaction, invoice date and time of each transaction, unit price, customer ID, and country
Piramuthu [54]	To develop a dynamically configurable supply chain framework using machine learning	Prediction of supply chain problems in e-commerce	The suggested framework automatically shapes the supply chain dynamically according to incoming orders and the constraints from suppliers upstream

in the application, the user gets a summary of what they need to know about the matter without having to search for and process reviews themselves. Text analysis comes at hand in addressing customer requirements and their feedback. Advanced text analysis allows for easy deciphering of the essence of notifications, whether it is a query to the helpdesk, in the chat, or a response to the service in e-mails. Phrase translators are now being gradually replaced by solutions based on neural networks, with the best results being achieved by specialized models that are trained using machine learning for specific areas [42].

Artificial Intelligence Assistants—Chatbots

The e-commerce website is accessible to shoppers 24 h a day, 7 days a week, with 24-h customer support by virtual assistants/chatbots. The primary function of the chatbot is to automatically answer customer queries, react to simple voice commands, and give product recommendations using a natural language processing system. Chatbots can also be defined as a type of software application that uses artificial intelligence

to conduct an online conversation via text or voice media with consumers visiting a given website or application. Chatbots are able to respond to customer questions and requests, can help consumers find suitable products, check order status, compare different products, and help consumers pay for their orders. In case of complaints or questions, they will direct customers to the customer support service staff who will take over the issue [15, 43, 44].

Recommendations Tool

Using an artificial intelligence algorithm, it is possible to do statistical programming, forecasting, and analyzing the behavior of consumers and huge data sets to anticipate which products could interest potential customers. From recent searches by prospective customers, the algorithm can capture key information of the searched product. The recommendation tool then generates appropriate suggestions for the search engine and displays what will ultimately help customers find the product quicker [15, 45, 46].

Visual and Voice Search

With the help of artificial intelligence, it is possible to implement visual and audio searches on websites. Visual and voice search is based on image and sound processing algorithms. Customers do not need to enter keywords in the search, and they can search for the product using an image or voice [45, 46]. In a visual search, a potential customer uses an image or photo as input instead of a regular text search. The customer takes a certain object or text or uploads an image which is then recognized by the search engine and displayed in the search results [47]. Voice search allows users to use spoken language as an input and find results by transcribing a voice query. Voice search is based on intelligent natural speech recognition and processing technologies. At the output, the user has either a spoken answer or relevant results in the form of text or images [48].

Customer Relationship Management

Customers are an important part of an e-commerce business. In the past, companies used employees to manage customer relationships. Nowadays, artificial intelligence systems are becoming more and more popular. Artificial intelligence is able to predict how consumers will behave when shopping, what products certain groups of customers will choose, and what steps should the company take to build and maintain the best possible relationships with its customers. Using artificial intelligence, a company can gather information about customer satisfaction and carefully plan how to respond to customer needs and requirements, regardless of time and situation. Artificial intelligence helps people build a balanced environment in which man and machine work together to make a profit and sell [35, 46].

2.2 Machine Learning Use in E-commerce

Machine learning is used in the e-commerce business to optimize prices, segmentation, personalization and targeting of customers, search results, product recommendations, predictions of customer behavior, automatic completion of site search, A/B testing, inventory management, customer support, strengthening omnichannel marketing, fraud protection, and image and sound recognition and processing [49, 50].

Machine learning offers a number of benefits (Fig. 4), especially for online retailers. The capability of algorithms to understand a wide range of data is invaluable. Machine learning helps increase conversions, runs more relevant campaigns, improves internal operational efficiency, improves decision-making, helps with personalization, fraud detection, and improves customer service [37, 49].

Martinez et al. [51] in their study focused on the development of the machine learning framework, which focused on predicting customers' shopping behavior over the next month. The main attributes they used were the number of purchases, the average time between purchases, the time since the last purchase, and the average value of the purchase. Cui et al. [52] developed a model using machine learning, aimed at predicting future product returns. They examined the variables of sales, time, product characteristics, vendor, production process and resources, order of several products and historical revenues, with sales proving to be the most significant attribute. Orogun [53] also developed a model for predicting consumer behavior using machine learning. The predictive attributes they used included invoice number, invoice date, transaction time, product code, product description, price, customer ID, and country. In his research, Piramuthu [54] focused on the development of a model that would use machine learning to predict supply chain problems when placing an order, taking into account the supplier's options.

Fig. 4 Advantages of machine learning for retailers



2.3 *Perspective of Artificial Intelligence in E-commerce*

Human intelligence often seems to be limited in performing certain tasks in e-commerce. This concerns in particular the forecasting of demand and supply chain mechanisms. It is in these cases, which pose a serious challenge for businesses, that artificial intelligence appears to be an appropriate tool [55]. Shankar [56] states that artificial intelligence helps increase the profitability of e-commerce through all available tools and helps improve personalized referrals and payments. It also improves customer relationship management, logistics management, and inventory optimization.

Artificial intelligence technologies have been integrated into marketing and retail, where big data analysis is applied to create personalized profiles of customers and predict their shopping behavior. Understanding and forecasting the demand of shoppers through integrated supply chains are more important nowadays, and artificial intelligence technology is expected to be an essential indispensable feature [57]. The advancement and application of artificial intelligence require a high level of acceptance of this technology in the future. With it, traders can match product information with information that consumers are looking for in order to assure the efficient consumption of products or services [58, 59].

Juniper Research [60] reports that the demand forecast using artificial intelligence will increase between 2019 and 2023, and that chatbot interactions will reach 22 billion from the current level of 2.6 billion over the same period. This research underlines that organizations are investing extensively in artificial intelligence to enhance trend analysis, logistics planning, and inventory management. Innovations based on artificial intelligence, like virtual mirror and visual search, are tailored to boost interactions with customers and reduce the disparity between physical and virtual shopping experiences.

According to forecasts made by the consulting company Gartner [61], most organizations that use artificial intelligence on their e-commerce platforms are projected to achieve an increase in customer satisfaction of at least 25% by 2023. This will also be accompanied by an increase in revenues and a reduction in costs. By 2025, it is also expected that 70% of companies will be forced to focus on small data instead of big data.

Scientists say it is likely that the relationship between artificial intelligence will lead to cooperation with humans rather than to the replacement of humans. An important step to success is to create partnerships in which artificial intelligence works and predicts while humans clarify and decide on appropriate measures [62–64].

The great potential of artificial intelligence is unquestionable, but many challenges still need to be met in order for solution to be put into practice. Shankar [56] recommends researching the consequences of the unintentional impact of artificial intelligence on customers. Luo et al. [65] recommend that attention should be paid to research on improving bots and eliminating their shortcomings, as that could strengthen consumer confidence. They also state that businesses should focus on streamlining the use of artificial intelligence in social networking. According to

Moriarty [66], the connection between artificial intelligence and virtual reality applications should be researched as well. According to Tousignant [67], further research should also focus on examining the relationship between artificial intelligence and online evaluations, in particular the identification of false reviews.

Kumar [45] points to possible risks and challenges for e-commerce companies that constrain the efficiency and effectiveness of artificial intelligence in fulfilling business expectations. For this reason, it is necessary to explore the possibilities and opportunities in view of the changing requirements of consumers in e-commerce. In order to support progress in research on the application of artificial intelligence in the business sector, it is necessary to examine the effectiveness of artificial intelligence in a multidisciplinary context. Based on comprehensive knowledge, experts and researchers will be able to set priorities and tasks to manage effective investments in important aspects of artificial intelligence, including e-business [68].

3 Conclusion

The aim of the paper was to outline the essence of e-commerce, artificial intelligence, machine learning, and the benefits they bring. The paper also provides an insight into the evaluation of the importance of artificial intelligence, machine learning, and their use in the future in the context of e-commerce based on an overview of available studies on this issue.

In the current world of commerce and digital technology, people use the Internet every day and are willing to test new products and brands, but at the same time they are critical in this regard. In this case, e-commerce seems to be a suitable way to satisfy their requirements. The use of artificial intelligence and its tools in electronic commerce has become the subject of interest of many scientists and business experts. Previous research has highlighted the need for further research that would contribute to the development of knowledge and strategies in the application of artificial intelligence in e-commerce. It is expected that artificial intelligence will continue to be used in electronic commerce and will become an integral part of all e-commerce companies.

Acknowledgements This paper is one of the partial outputs under the scientific research grants VEGA 1/0609/19 “Research on the development of electronic and mobile commerce in the aspect of the impact of modern technologies and mobile communication platforms on consumer behavior and consumer preferences” and VEGA 1/0694/20 “Relational marketing research—perception of e-commerce aspects and its impact on purchasing behavior and consumer preferences.”

References

1. Gburová J (2019) Consumer shopping behavior in the e-commerce environment. *J Glob Sci* 4(2):1–6
2. Khrais L (2020) Role of artificial intelligence in shaping consumer demand in e-commerce. *Future Internet* 12(12)
3. Delina R, Vajda V (2006) Theory and practice of electronic commerce. Grafotlač, Prešov
4. Hagberg J, Sundstrom M, Egels-Zandén N (2016) The digitalization of retailing: an exploratory framework. *Int J Retail Distrib Manage* 44:694–712
5. Menaka B, Seethal K (2018) Recent trends in e-commerce. *Shanlax Int J Commer* 6(1):40–44
6. Tan S (2013) *Ecom hell: how to make money in ecommerce without getting burned*. Ecom Hell, San Francisco
7. Ullman L (2013) *Effortless e-commerce with PHP and MySQL*. New Riders, San Francisco
8. Laudon KC, Traver CG (2021) *E-commerce 2020–2021-business—technology—society*. Pearson education limited, United Kingdom
9. Rahman MA, Islam MA, Esha BH, Sultana N, Chakravorty S (2018) Consumer buying behavior towards online shopping: an empirical study on Dhaka city Bangladesh. *Cogent Bus Manage* 5(1):1–22
10. Prabha RJ, Karunanidhi M (2017) A study on consumers problems towards online shopping. *Int J Bus Manage Invention* 4(3):36–37
11. eMarketer (2021) Retail e-commerce sales, by Country. <https://www.emarketer.com/forecasts/5a53f4f2d8690c0d70ffaa06/5b2aab133808a50ff80464db>, Accessed 10 Oct 2021
12. Chaffey D, Hemphill T, Edmundson-Bird D (2019) *Digital business and e-commerce management*. Pearson education limited, UK
13. Diaz C (2021) What is E-commerce? definition, differences with other terms and first steps to follow if you want to launch your e-commerce. <https://www.doofinder.com/en/blog/what-is-e-commerce>, Accessed 10 July 2021
14. Ganapathi R (2015) A study on factors affecting online shopping behaviour of consumers in Chennai. *J Manage Res Anal* 2(2):123–126
15. Song X, Yang S, Huang Z, Huang T (2019) The application of artificial intelligence in electronic commerce. In: *The 4th annual international conference on information system and artificial intelligence*, pp 1–6. IOP Conference series, Hunan, China
16. Nisar TM, Prabhakar G (2017) What factors determine e-satisfaction and consumer spending in e-commerce retailing? *J Retail Consum Serv* 39:135–144
17. Kim TY, Dekker R, Heij C (2017) Cross-border electronic commerce: distance effects and express delivery in European union markets. *Int J Electron Commer* 21:184–218
18. Europarl: Artificial intelligence—definition and use. <https://www.europarl.europa.eu/news/sk/headlines/society/20200827STO85804/umela-inteligencia-definicia-a-vyuzitie>, Accessed 10 July 2021
19. Duan Y, Edwards JS, Dwivedi YK (2019) Artificial intelligence for decision making in the era of big data—evolution, challenges and research agenda. *Int J Inf Manage* 48:63–71
20. Lu Y, Xu LD (2018) Internet of things (IoT) cybersecurity research: a review of current research topics *IEEE. Internet Things J* 6(2):2103–2115
21. Pee L, Pan SL, Cui L (2019) Artificial intelligence in healthcare robots: a social informatics study of knowledge embodiment. *J Am Soc Inf Sci* 70:351–369
22. Yoon M, Baek J (2016) Paideia education for learners' competencies in the age of artificial intelligence—the google DeepMind challenge match. *Int J Multimedia Ubiquit Eng* 11:309–318
23. Ying W, Pee LG, Jia S (2018) Social informatics of intelligent manufacturing ecosystems: a case study of KuteSmart. *Int J Inf Manage* 42:102–105
24. Russell SJ, Norvig P (2016) *Artificial intelligence: a modern approach* (3rd ed.). Pearson, Essex
25. Dwivedi YK, Hughes L, Ismagilova E, Aarts G, Coombs C, Crick T et al (2021) Artificial intelligence (AI): multidisciplinary perspectives on emerging challenges, opportunities, and agenda for research, practice and policy. *Int J Inf Manage* 57

26. SAS: Artificial intelligence—what is it and how it works, https://www.sas.com/sk_sk/insights/analytics/what-is-artificial-intelligence.html, Accessed 10 July 2021
27. Duan N, Liu LZ, Yu XJ, Li Q, Yeh SC (2019) Classification of multichannel surface-electromyography signals based on convolutional neural networks. *J Industr Integr Manag* 15:201–206
28. Swathi B, Babu SS, Ayyavaraiah M (2019) Artificial intelligence: characteristics, subfields, techniques and future predictions. *J Mech Continua Math Sci* 14(6):127–135
29. Huang MH, Rust RT (2018) Artificial intelligence in service. *J Serv Res* 21(2):155–172
30. Zhang C, Yang L (2021) Study on artificial intelligence: the state of the art and future prospects. *J Ind Inf Integr* 23
31. Kelemen M, Polishchuk V, Gavurová B, Szabo S, Rozenberg R, Gera M, Kozuba J Andoga R, Divoková A, Blišťan P (2019) Fuzzy model for quantitative assessment of environmental start-up projects in air transport. *Int J Environ Res Public Health* 16(19)
32. Polishchuk V, Kelemen M, Gavurová B, Varotsos C, Andoga R, Gera M, Christodoulakis J, Soušek R, Kozuba J, Blišťan P, Szabo S (2019) A fuzzy model of risk assessment for environmental start-up projects in the air transport sector. *Int J Environ Res Public Health* 16(19)
33. Davenport TH, Ronanki R (2018) Artificial intelligence for the real world. *Harv Bus Rev* 96:108–116
34. Bughin J, Seong J, Manyika J, Chui M, Joshi R (2018) Notes from the AI frontier: modeling the impact of AI on the world economy. <https://www.mckinsey.com/featured-insights/artificial-intelligence/notes-from-the-ai-frontier-modeling-the-impact-of-ai-on-the-world-economy>, Accessed 10 July 2021
35. Soni VD (2020) Emerging roles of artificial intelligence in ecommerce. *Int J Trend Sci Res Dev* 4(5):223–225
36. Výchá D. What is machine learning? <https://itya.sk/machine-learning/>. Accessed 10 Oct 2021
37. Shaw N. Ecommerce machine learning: AI's role in the future of online shopping, <https://www.bigcommerce.com/blog/ecommerce-machine-learning/#differences-between-machine-learning-and-artificial-intelligence>. Accessed 10 Oct 2021
38. NBS. Artificial intelligence and “Big Data”. <https://www.nbs.sk/sk/dohlad-nad-financnym-trhom/fintech/umela-inteligencia-a-big-data>. Accessed 10 Oct 2021
39. Stieglitz S, Dang-Xuan L, Bruns A et al (2014) Social media analytics. *Bus Inf Syst Eng* 6:89–96
40. van Zoonen W, van der Meer TG (2016) Social media research: the application of supervised machine learning in organizational communication research. *Comput Hum Behav* 63:132–141
41. Davis Babu A. Artificial intelligence versus machine learning versus deep learning (AI versus ML versus DL). https://medium.com/@alanb_73111/artificial-intelligence-vs-machine-learning-vs-deep-learning-ai-vs-ml-vs-dl-e6afb7177436. Accessed 13 Oct 2021
42. TouchIt: E-commerce 4.0. Artificial intelligence, advanced text analysis and machine translations are emerging. <https://touchit.sk/e-commerce-4-0-nastupuje-umela-inteligencia-pokroci-la-analyza-textov-aj-strojove-preklady/233309>. Accessed 11 Oct 2021
43. Soni VD. Challenges and solution for artificial intelligence in cybersecurity of the USA, <https://ssrn.com/abstract=3624487>, Accessed 11 July 2021
44. Kar R, Haldar R (2016) Applying chatbots to the internet of things: opportunities and architectural elements. *Int J Adv Comput Sci Appl* 7(11):147–154
45. Kumar T, Trakru M (2019) The colossal impact of artificial intelligence in e-commerce: statistics and facts. *Int Res J Eng Technol* 6(5):570–572
46. Pallathadka H, Ramirez-Asis EH, Loli-Poma TP, Kaliyaperumal K, Ventayen RJ, Naved M (2021) Applications of artificial intelligence in business management, e-commerce and finance. *Materials Today: Proceedings* (Article in Press)
47. Klačko R. What is SEO waiting for in 2020? Trends and tips. <https://digicchef.cz/co-caka-seo-v-roku-2020-trendy-rady-a-tipy>. Accessed 12 July 2021
48. Pastierová M. Are we ready for voice search? <https://itilib.cvtisr.sk/wp-content/uploads/docs/5.pdf>, last accessed 2021/07/12

49. Vekony B. 12 Best machine learning strategies for eCommerce businesses. <https://www.prefixbox.com/blog/machine-learning-for-ecommerce/>. Accessed 12 Oct 2021
50. Haponik A. The best machine learning use cases in Eur commerce. <https://addepto.com/best-machine-learning-use-cases-ecommerce/>. Accessed 12 Oct 2021
51. Martínez A, Schmuck C, Pereverzyev Jr S, Pirker C, Haltmeier M (2018) A machine learning framework for customer purchase prediction in the non-contractual setting. *European J Oper Res*
52. Cui H, Rajagopalan S, Ward AR (2020) Predicting product return volume using machine learning methods. *Eur J Oper Res* 281(3):612–627
53. Orogun A, Onyekwelu B (2019) Predicting consumer behaviour in digital market: a machine learning Approach. *Int J Innov Res Sci Eng Technol* 8(8):8391–8402
54. Piramuthu S (2005) Machine learning for dynamic multi-product supply chain formation. *Expert Syst Appl* 29(4):985–990
55. Soni N, Sharma EK, Singh N, Kapoor. A impact of artificial intelligence on businesses: from research, innovation, market deployment to future shifts in business models. <https://arxiv.org/abs/1905.02092>. Accessed 12 July 2021
56. Shankar V (2018) How artificial intelligence (AI) is reshaping retailing. *J Retail* 94(4):6–11
57. Zhang D, Pee LG, Cui L (2021) Artificial intelligence in E-commerce fulfillment: a case study of resource orchestration at Alibaba’s smart warehouse. *Int J Inf Manage* vol 57
58. Sterne J (2017) *Artificial intelligence for marketing: practical applications*. John Wiley & Sons Inc., USA
59. Sunstein RC (2016) Fifty shades of manipulation. *J Mark Behav* 1(3–4):213–244
60. Juniper Research. AI in retail. segment analysis, vendor positioning and market forecasts 2019–2023. <https://www.juniperresearch.com/researchstore/fintech-payments/ai-in-retail>. Accessed 11 July 2021
61. Gartner. The 4 trends that prevail on the Gartner hype cycle for AI, 2021, <https://www.gartner.com/en/articles/the-4-trends-that-prevail-on-the-gartner-hype-cycle-for-ai-2021>. Accessed 11 Oct 2021
62. Wang L, Wang XV (2016) Outlook of cloud, CPS and IoT in manufacturing. *Cloud-based cyber-physical systems in manufacturing*. Springer, Cham
63. Kumar SL (2017) State of the art-intense review on artificial intelligence systems application in process planning and manufacturing. *Eng Appl Artif Intell* 65:294–329
64. Makridakis S (2018) Forecasting the impact of artificial intelligence, Part 3 of 4: the potential effects of AI on businesses, manufacturing, and commerce. *Foresight Int J Appl Forecast* 49:18–27
65. Luo X, Tong S, Fang Z, Qu Z (2019) Frontiers: machines versus humans: the impact of artificial intelligence chatbot disclosure on customer purchases. *Mark Sci* 38(6):937–947
66. Moriarty E. How artificial intelligence and augmented reality can put a dent in return rates, <https://www.digitalcommerce360.com/2020/06/08/how-artificial-intelligence-and-augmented-reality-can-put-a-dent-in-return-rates/>. Accessed 11 July 2021
67. Tousignant L. Robots learned how to write fake yelp reviews like a human. <https://nypost.com/2017/08/31/robots-learned-how-to-write-fake-yelp-reviews-like-a-human/>. Accessed 11 July 2021
68. Loureiro SMC, Guerreiro J, Tussyadiah I (2021) Artificial intelligence in business: state of the art and future research agenda. *J Bus Res* 129:911–926

Improved Sliding Mode Control for Glucose Regulation of Type 1 Diabetics Patients Considering Delayed Nonlinear Model



Hamed Khodadadi, Hamid Ghadiri, and Ali Dehghani

Abstract Diabetes has been introduced as the sixth leading cause of death in the world. Type 1 diabetic patients should be injected insulin to keep their blood glucose at a safe level range. Due to an excessive increase in insulin injection can cause human death, it should be completely closed-looped controlled. On the other hand, the delayed nonlinear model of the glucose-insulin system makes some challenges in its control. In this paper, an improved sliding mode control (SMC) technique is proposed to regulate patients' blood glucose levels and suffer type 1 diabetes. The proposed improved SMC (ISMC) determines the pumping rate of insulin as a control signal in the closed-loop control. Simulation results demonstrate that the patient's blood glucose is regulated by optimal insulin injection rate. Compared to classical SMC, the proposed ISMC has higher accuracy and a more efficient control signal.

Keywords Type 1 diabetes · Sliding mode control · Delayed nonlinear model · Glucose-insulin system

H. Khodadadi

Department of Electrical Engineering, Khomeinishahr Branch, Islamic Azad University, Isfahan, Iran

e-mail: khodadadi@iaukhsh.ac.ir

H. Ghadiri (✉)

Department of Electrical Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran

e-mail: h.ghadiri@qiau.ac.ir; h.ghadiri@iitiii@gmail.com

A. Dehghani

National Laboratory of Pattern Recognition, Institute of Automation of Chinese Academy of Sciences, Beijing 100190, China

e-mail: ali.dehghani82@mails.ucas.ac.cn

University of Chinese Academy of Sciences, Beijing, China

1 Introduction

Diabetes is a metabolic disorder in which blood glucose levels rise above the normal range of 70–110 mg/dl in the fasting glucose test. In diabetics, the beta cells that are the source of insulin production in the body are destroyed, and the body cannot control blood glucose levels alone [1]. In this case, blood sugar should be regulated using insulin injection. The closed-loop control acts as an artificial pancreas. In this case, the patient's blood glucose concentration (BGC) is measured immediately by a glucose sensor, and the amount of insulin injected is determined by an appropriate control algorithm [2]. Finally, the amount of insulin is injected continuously into the patient using an insulin injection pump. Blood glucose sensor, insulin injection pump, and controller are the main components of the closed-loop control strategy. Figure 1 shows the outline of a closed-loop control system for a diabetic patient using an insulin injection pump.

Generally, finding an appropriate model for describing the hemodynamic variable and biological tissues has a great importance [3, 4]. There are several mathematical models for diabetes based on the interaction of insulin and glucose in the body, including the most common models for diabetes, which are Bergman model [5], Dalla Man's model [6], Hovorka's model [7], and Sorenson's model [8]. In some models, to obtain fluctuations in insulin secretion, insulin must be divided into plasma and intercellular insulin components, which is one of the disadvantages of the proposed model. In this paper, the Palumbo delayed nonlinear model is used due to the nonlinear behavior of insulin and glucose interaction in the body of type 1 diabetic patients. One of the most important advantages of this model is compliance with the behavior of the patient population according to the IVGTT intravenous glucose injection test [9, 10]. Researchers have proposed several control methods to control the BGC in type 1 diabetes mellitus (T1DM) patients, such as fuzzy control [11], H_{∞} control [12], SMC [13], model predictive control (MPC) [14], and observer-based nonlinear control [15].

The control approach based on nonlinear models is based on more knowledge of the physiological behavior of the patient community and naturally provides the possibility of presenting different control theory methods in regulating blood glucose. Obviously, the closer the model's behavior is to the nature of the patient's body, and

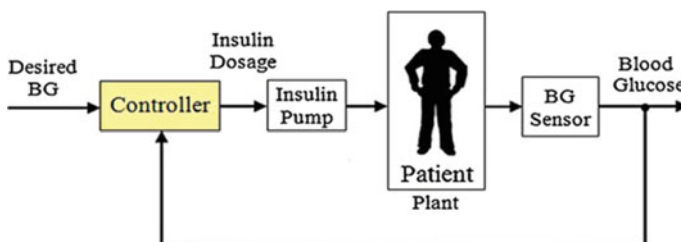


Fig. 1 Closed-loop control system for diabetic patients [15]

the more precise control law will be obtained. Providing appropriate solutions for controlling delayed nonlinear models of diabetes is still an important issue. Lack of sensitivity to internal and external disturbances, ultimate accuracy and pervasiveness, as well as limited-time convergence [16, 17], which are the main features of SMC, make it a good choice for control algorithms related to the human body. The SMC approach is a simple, robust control method that is the best option to maintain stability and consistent performance in the face of uncertainty in modeling. SMC's main idea is developed based on defining a sliding surface that converges the system to the surface using the proper control law and ensuring system stability [18, 19]. High sensitivity and change of model parameters in different patients, use of a high-performance structure, and robustness to these changes. On the other hand, the parameters of the model have many variations among different patients. Therefore, the motivation to use ISMC is explained in the structure of the blood glucose regulation system.

2 Glucose–Insulin Model

Delayed nonlinear models have received more attention in recent years due to their consistency with the results of diagnostic tests on the patient data. The presence of delays in nonlinear equations has reduced the model's order, albeit making their analysis more complex. A delayed nonlinear model was introduced in [7, 8], which describes the relation between glucose G^* [mM] and insulin $I(t)$ [pM] in the body regulatory system (1).

$$\begin{aligned} \frac{dG(t)}{dt} &= -K_{xgi}I(t)G(t) + \frac{T_{gh}}{V_g} \\ \frac{dI(t)}{dt} &= -K_{xi}I(t) + \frac{T_{ig\max}}{V_i}f(G(t - \tau_g)) + u(t) \\ \frac{dq(t)}{dt} &= G_{ref} - G(t) \\ G(\tau) &= G_0(\tau), I(\tau) = I_0(\tau), \tau \in [-\tau_g, 0] \end{aligned} \quad (1)$$

where $G_0(\tau)$ and $I_0(\tau)$ describe the initial values of patient glucose and plasma insulin can be considered based on the $G(0) = G_b$ and $I(0) = I_b$. In addition, γ is a positive constant parameter for circulating the plasma glucose, $u(t)$ is the subcutaneous insulin delivery rate ($pM\min^{-1}$), τ_g indicates the delay of the pancreas in insulin secretion when the BGC is increased (min), V_g demonstrates the glucose apparent distribution volume (L/kgBW), V_i illustrates the apparent distribution volume for insulin (L/kgBW), K_{xgi} shows the glucose rate uptake by tissues per pM of plasma insulin concentration ($pM^{-1}\min^{-1}$). Besides, T_{gh} and $T_{ig\max}$ represent the related indicator between hepatic glucose and the glucose tissue uptake ($\min^{-1}(\text{mmol}, \text{kgBW})$) and the maximum rate of second-phase insulin release ($\min^{-1}(\text{pmol}, \text{kgBW})$), respectively.

The nonlinear function $f(G)$ indicates the released insulin rate and presented as Eq. (2).

$$\begin{aligned} f(G) &= \frac{\left(\frac{G}{G^*}\right)^\gamma}{1 + \left(\frac{G}{G^*}\right)^\gamma} \\ T_{gh} &= K_{xgi} I_b G_b V_g \\ T_{igmax} &= K_{xi} I_b V_i \left[1 + \left(\frac{G_b}{G^*}\right)^\gamma \right] / \left(\frac{G_b}{G^*}\right)^\gamma \end{aligned} \quad (2)$$

G^* represents the amount of glucose in the blood when insulin secretion is less than half of its maximum [10].

3 Controller Architecture

The sliding mode controller is designed for type 1 diabetic patients' glucose regulation using the Bergman basic model in several studies. Insensitivity to internal and external disturbances, acceptable accuracy, robustness, as well as a limited convergence time are the main specifications of the SMC that make it a suitable choice for algorithms related to the human body [7].

Traditional SMCs have several flaws, including discontinuity in the control signal [20]. The higher-order sliding approach is used to solve these problems and achieve higher accuracy. The sliding mode variable is stabilized at the origin using k th order HOSM. HOSM is a suitable choice for controller design because of two fundamental characteristics:

1. This strategy improves the efficiency and accuracy of the design.
2. This strategy leads to a continuous control signal.

3.1 HOSM Controller Design

The state-space equations for the system introduced by Bergmann's minimal model are as follows:

$$\begin{aligned} \dot{x}_1 &= -p_1[x_1 - G_b] - x_1 x_2 + D(t) \\ \dot{x}_2 &= -p_1 x_2 + p_3[x_3 - I_b] \\ \dot{x}_3 &= -n[x_3 - I_b] - \gamma[x_1 - h] + t + u(t) \end{aligned} \quad (3)$$

where x_1 , x_2 , and x_3 are concentration of glucose in the blood plasma, Insulin's impact on glucose loss in networks G_b (1/min), and Plasma insulin concentration ($\mu U/ml$). Stabilizing a diabetic patient's blood glucose at a level of G_b is primarily an

output tracking issue. As a result, tracking error is defined as the difference between a diabetic patient’s blood concentration level and its critical amount, as shown in Eq. (4).

$$e = G_b - G(t) = G_b - x_1 \tag{4}$$

The controller $u(t)$ for the system presented by Eq. (3) should be designed in such a manner that the error tends to zero in the presence of uncertainties, parameter changes, disturbances, and food absorption indicated by $D(t)$.

Firstly, the system’s relative degree should be calculated. The relative degree is defined as the number of consecutive derivatives before the control signal comes in the equation. As a result of the r relative degree, the controller signal appears first in r th derivative of the output. The control signal appears in the equations after three derivations using Eq. (3), i.e.,

$$x_1^{(3)} = \phi(x, t) - p_3x_1u(t) \tag{5}$$

where

$$\begin{aligned} \phi(x, t) = & x_1[-p_1(p_1^2 + 3p_3I_b) - p_3I_b(p_2 + n) - p_3\gamma(x_1 - h) + t] \\ & + x_2[-p_1^2(1 + G_b) + p_1p_2(2G_b - 1) + 2D(p_1 + p_2)] \\ & + x_3[-2p_3(p_1 + D)] + x_1x_2[-(p_1 + p_2)^2 - 3p_3I_b] \\ & + x_1x_3[p_3(3p_1 + p_2 + n)] + x_1x_2^2[-3(p_1 + p_2)] \\ & + x_2^2(p_1G_b + D) + 3p_3x_1x_2x_3 - x_1x_2^3 + \ddot{D} \\ & + (p_1G_b + D)(p_1^2 + 2p_3I_b) \end{aligned} \tag{6}$$

Since $p_3 \neq 0$, $x_1 \neq 0$, and $p_3x_1 \in [1.2 \times 10^{-4}, 3 \times 10^{-2}]$, Eq. (3) has a well-defined relative degree $p = 3$, causes the controller satisfy $e \rightarrow 0$ condition. For designing the controller, the sliding variable is represented by (7).

$$\sigma = e = G_b - G(t) = G_b - x_1 \tag{7}$$

Equation (8) can be found by derivation of the (7).

$$\begin{aligned} \sigma^{(3)} &= -\phi(x, t) + p_3x_1u(t) \\ \dot{\sigma}(t) &= v \\ \sigma^{(4)} &= -\phi_1(x, t) + p_3x_1v \end{aligned} \tag{8}$$

where

$$\begin{aligned} \phi_1(x, t) &= -\dot{\phi}(x, t) + p_3\dot{x}_1u(t) \\ &= -\dot{\phi}(x, t) + p_3u(t)[-p_1(x_1 - G_b) - x_1x_2 + D(t)] \end{aligned} \tag{9}$$

Equation (9) is assumed to be finite during the designing process, means $|\emptyset_1| \leq L$. The HOSM, stabilizes σ in a short time at the origin, can be considered as Eq. (10).

$$x_1^{(3)} = \phi(x, t) - p_3x_1u(t) \tag{10}$$

It is obvious that the inclusion of virtual control adds a derivative, resulting in an increase in the system’s relative degree from three to four. For calculating $\sigma^k, k = 1, 2, 3, \dots$ the HOSM derivative is employed that the n order derivative of a uniform function $f(t)$ has the following general form:

$$\begin{aligned} \dot{z}_0 &= v_0 \\ v_0 &= -\lambda_0|z_0 - f(t)|^{(n/n+1)}\text{sign}(z_0 - f(t)) + z_1 \\ \dot{z}_1 &= v_1 \\ v_1 &= -\lambda_1|z_1 - v_0|^{(n-1/n)}\text{sign}(z_1 - v_0) + z_2 \\ &\vdots \\ \dot{z}_{n-1} &= v_{n-1} \\ v_{n-1} &= -\lambda_{n-1}|z_{n-1} - v_{n-2}|^{(1/2)}\text{sign}(z_{n-1} - v_{n-2}) + z_n \\ \dot{z}_0 &= v_0 \end{aligned} \tag{11}$$

Increasing the relative degree from three to four causes the virtual control to switch at a high frequency, while the local control u remains continuous, resulting in $u = f v d\tau$. In addition, instead of high order control in Eq. (10), fourth-order pseudo-continues control in Eq. (3) can be applied [6].

Dynamic system can be defined as:

$$S = s(t, x)x^r = a(t, x) + (t, x)u \tag{12}$$

where r stands for the relative degree, and the equation S is as follows:

$$\begin{aligned} S^r &= h(t, x) + g(t, x)u \\ 0 < K &\leq \frac{\partial}{\partial u} s^r \leq K_m \end{aligned} \tag{13}$$

If = 1:

$$\begin{aligned} S &= (G_{\text{ref}} - \dot{G}) + \lambda * (G_{\text{ref}} - G(t)) \\ U &= -K * \text{sign}(s) \end{aligned} \tag{14}$$

In this case, the value of K from the input function changes in an interval of $[-50, 50]$. The changes u are checked for each moment, and the control input is changed according to the amount of insulin required and the sliding surface. Hence, the proposed improved SMC is adopted based on the sliding surface and proposes an optimal K considering the insulin variations.

4 Simulation Results

In order to track the glucose its reference, the variable $q(t)$ is inserted into the model equations according to Eq. (15).

$$\begin{aligned}
 \frac{dG(t)}{dt} &= -K_{xgi}I(t)G(t) + \frac{T_{gh}}{V_g} \\
 \frac{dI(t)}{dt} &= -K_{xi}I(t) + \frac{T_{igmax}}{V_i}f(G(t - \tau_g)) + u(t) \\
 \frac{dq(t)}{dt} &= G_{ref} - G(t) \\
 G(\tau) &= G_0(\tau), I(\tau) = I_0(\tau), \tau \in [-\tau_g, 0]
 \end{aligned}
 \tag{15}$$

For a pseudo-linear representation of the above-mentioned equations, the matrices $A^\sim(xt^\sim)$, and $B^\sim(xt^\sim)$ will be defined as (16):

$$\begin{aligned}
 \tilde{A}(\tilde{X}_t) &= \begin{bmatrix} -K_{xgi}I(t) & \frac{T_{gh}}{V_g} \left(\frac{1}{I(t)} \right) & 0 \\ \frac{T_{igmax}}{V_i G(t)} \left(\frac{\left(\frac{G(t-\tau_g)}{G^*} \right)^\gamma}{1 + \left(\frac{G(t-\tau_g)}{G^*} \right)^\gamma} \right) & -K_{xi} & 0 \\ -1 & 0 & 0 \end{bmatrix} \\
 \tilde{B}(\tilde{X}_t) &= \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}
 \end{aligned}
 \tag{16}$$

Assuming that the initial system conditions are positive, the values $u(t)$ and $I(t)$ are positive for all times; therefore, the pseudo-linear matrix elements will be continuous and nonzero. On other hand, due to the full rank order of the controllability and observability matrices of the system, the HOSM controller can be designed. The parameters of the diabetes nonlinear model for the studied patients are obtained based on the fitting of the generalized least squares on the experimental data of the insulin intravenous injection test in [14].

The values of I_b and G_b can be measured directly; some parameters such as G^* and V_i are fixed and known, and parameters such as τ_g , V_g , K_{xgi} , and γ are estimated

for each patient. T_{igmax} and T_{gh} are also parameters determined based on the physical conditions of each patient based on algebraic equations.

In this study, the patient has a body mass index of 50. Besides, $G_b = 6.14$ indicates a higher-than-normal blood glucose, and the insulin resistance index is $K_{xgi} \ll 10^{-4}$. These factors indicate abnormal insulin secretion rate for a newly infected diabetic patient, in which some factors such as obesity, inactivity, and genetics have led to a gradual decrease in insulin secretion rate in the patient. The parameters of the patient delayed nonlinear model are illustrated in Table 1 [14].

The glucose reference signal is considered as a decrease of the initial value of the patient’s blood glucose $G_b = 6.14$ to the normal value of 5.2 mM as (17).

$$G_{ref}(t) = 5.2 + (6.14 - 5.2)e^{-0.02t} \tag{17}$$

Table 1 Parameters of the patient delayed nonlinear model [14]

Parameters	Value
G_b	6.14
I_b	93.669
T_{igmax}	1.573
γ	3.205
G^*	9
τ_g	24
V_g	0.187
K_{xi}	1.211×10^{-2}
T_{gh}	0.003
V_i	0.25
K_{xgi}	3.11×10^{-5}

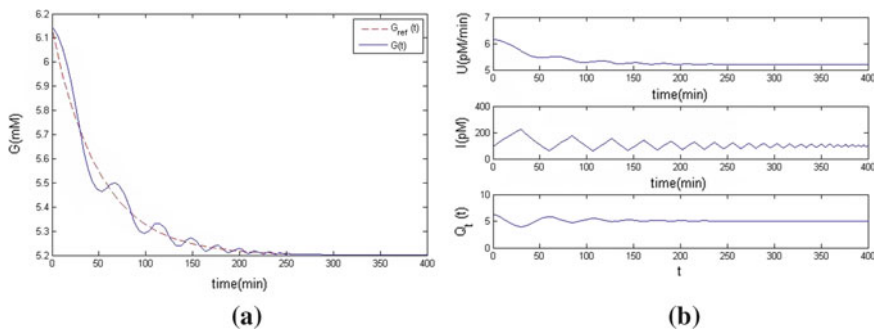


Fig. 2 **a** Tracking of the reference signal by the patient glucose applying the SMC. **b** (U): Insulin injection rate (control signal), (I): Patient blood insulin, (Q): $G_{ref} - G(t)$

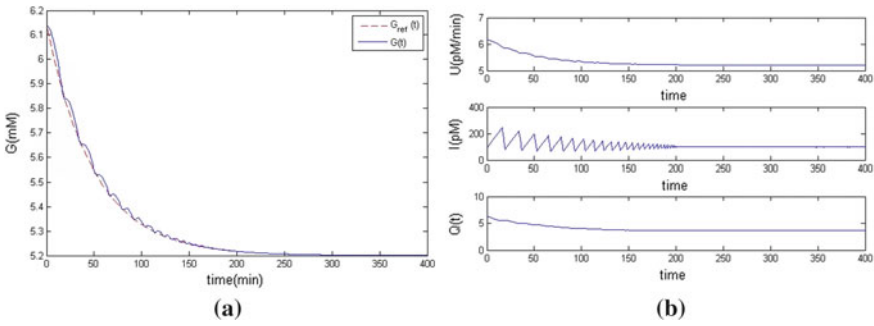


Fig. 3 **a** Tracking of the reference signal by the patient glucose applying the ISMC. **b** (U): Insulin injection rate (control signal), (I): Patient blood insulin, (Q): $G_{ref} - G(t)$

Figure 2 shows the convergence of the diabetic patient’s blood glucose to the glucose reference signal, insulin changes in the patient’s blood plasma, and the rate of insulin injection as the control signal provided in the closed-loop control system. The classical SMC is applied as the controller in this section.

By considering the same condition, the simulation results of applying the proposed ISMC for the blood glucose, insulin changes, and the rate of insulin injection are provided in Fig. 3.

The simulation results demonstrate the ability of the proposed controller in accurately tracking the optimal glucose levels for patients based on the best insulin injection rate. Compared to the feedback linearization method (the method used by Palmbo), the amplitude of the control signal or insulin injection rate for the patient is reduced, indicating that the patient will achieve normal body conditions in a shorter time. In addition, it can be comprehended from comparing Figs. 2 and 3 that tracking in the ISMC achieves significantly better convergence than the traditional SMC.

5 Conclusion

In this paper, a sliding mode design is considered for the delayed nonlinear model of Palumbo diabetes. In this study, the ISMC method is applied and a continuous and comprehensive controller is designed to maintain blood glucose at a pre-defined level. The simulation results also show that by changing the structure of the sliding mode and optimizing it, delayed biological systems have better controllability. Another advantage of this method is that the patient glucose reaches to its reference signal in a shorter time and applies lower amount of insulin injection rate.

References

1. Heydarinejad H, Delavari H (2017) Fractional order back stepping sliding mode control for blood glucose regulation in type I diabetes patients. In: *Theory and applications of non-integer order systems*, pp 187–202
2. von Scholten BJ, Kreiner FF, Gough SC, von Herrath M (2021) Current and future therapies for type 1 diabetes. *Diabetologia*, pp 1–12
3. Khodadadi H, Khaki-Sedigh A, Ataei M, Jahed-Motlagh MR (2018) Applying a modified version of Lyapunov exponent for cancer diagnosis in biomedical images: the case of breast mammograms. *Multidimensional Syst Signal Process* 29(1):19–33
4. Ahmadvan MR, Ghadiri H, Hajian SR (2021) Model predictive control optimisation using the metaheuristic optimisation for blood pressure control. *IET Syst Bio* 15(2):41–52
5. Mohabati F, Molaei M (2020) Bifurcation analysis in a delay model of IVGTT glucose-insulin interaction. *Theory Biosci* 139(1):9–20
6. Dalla Man C, Rizza RA, Cobelli C (2007) Meal simulation model of the glucose-insulin system. *IEEE Trans Biomed Eng* 54(10):1740–1749
7. Boughton CK, Hartnell S, Thabit H, Poettler T, Herzig D, Wilinska ME, Hovorka R (2021) Hybrid closed-loop glucose control with faster insulin aspart compared with standard insulin aspart in adults with type 1 diabetes: A double-blind, multicentre, multinational, randomized, crossover study. *Diab Obes Metab* 23(6):1389–1396
8. Roostalu U, Lercke Skytte J, Gravesen Salinas C, Klein T, Vrang N, Jelsing J, Hecksher-Sørensen J (2020) 3D quantification of changes in pancreatic islets in mouse models of diabetes type I and II. *Dis Models Mech* 13(12):dmm045351
9. Palumbo P, Pepe P, Panunzi S, De Gaetano A (2012) Time-delay model-based control of the glucose-insulin system, by means of a state observer. *European J Control* 6:591–606
10. Palumbo P, Pepe P, Panunzi S, De Gaetano A (2011) Glucose control by subcutaneous insulin administration: a DDE modelling approach. *IFAC Proc Vol* 44(1):1471–1476
11. Yadav J, Rani A, Singh V (2016) Performance analysis of fuzzy-PID controller for blood glucose regulation in type-1 diabetic patients. *J Med Syst* 40(12):1–15
12. Olmagna P et al (2014) Reducing risks in type 1 diabetes using H_{∞} control. *IEEE Trans Biomed Eng* 61(12):2939–2947
13. Humaidi AJ, Hasan S, Al-Jodah AA (2018) Design of second order sliding mode for glucose regulation systems with disturbance. *Int J Eng Technol* 7(2.28):243–247
14. Patra AK, Rout PK (2017) Adaptive continuous-time model predictive controller for implantable insulin delivery system in type I diabetic patient. 38.2:184–204
15. Nath A, Dey R, Aguilar-Avelar C (2019) Observer based nonlinear control design for glucose regulation in type 1 diabetic patients: An LMI approach. *Biomed Signal Process Control* 47:7–15
16. Ghadiri H, Mohammadi A, Khodadadi H (2022) Fast terminal sliding mode control based on SDRE observer for two-axis gimbal with external disturbances. *J Brazilian Soc Mech Sci Eng* 44(70):1–23
17. Ghadiri H, Emami M, Khodadadi H (2021) Adaptive super-twisting non-singular terminal sliding mode control for tracking of quadrotor with bounded disturbances. *Aerosp Sci Technol* 112:106616
18. Emamifard A, Ghadiri H (2021) Robust control of nonlinear fractional-order systems with unknown upper bound of uncertainties and external disturbance. *IETE J Res*, pp 1–12
19. Ghadiri H, Khodadadi H, Mobayen S, Asad JH, Rojsiraphisal T, Chang A (2021) Observer-based robust control method for switched neutral systems in the presence of interval time-varying delays. *Mathematics* 9(19):2473
20. Rahmanipour P, Ghadiri H (2020) Stability analysis for a class of fractional-order nonlinear systems with time-varying delays. *Soft Comput* 24(22):17445–17453

Overview and Computational Analysis of PSO Variants for Solving Systems of Nonlinear Equations



Sérgio Ribeiro and Luiz Guerreiro Lopes

Abstract The problem of solving systems of nonlinear equations is one of great difficulty, especially as the scale of these systems grow. Traditional numerical methods rely on selecting good initial estimates for the roots and refine them iteratively, which is not a simple task, especially with large systems of nonlinear equations. Particle swarm optimization (PSO), in turn, is a nature-inspired metaheuristic algorithm for finding the minimum of a function for which multiple improvements and different hybridizations have been proposed. These modifications range from dynamically choosing parameters and topologies to hybridization with other population-based optimization algorithms. All of these modifications have the goal of improving the basic algorithm, and a broader comparison between these proposals is necessary to further understand what might be the path forward in achieving better and more exact results. In this paper, after a brief overview of PSO-based algorithms for nonlinear equation systems, several PSO variants are tested on a number of problems in order to find the solution to non-trivial systems of nonlinear equations. Each variant was tested 100 times on each problem, in order to produce enough samples for a legitimate comparison.

Keywords Computational intelligence · Particle swarm optimization · PSO variants · Nonlinear equation systems

S. Ribeiro (✉)

Postgraduate Programme in Informatics Engineering, University of Madeira, Funchal, Madeira Is., Portugal

e-mail: sergioribeiro_91@hotmail.com

L. G. Lopes

Faculty of Exact Sciences and Engineering, University of Madeira, Funchal, Madeira Is.9020-105, Portugal

e-mail: lopes@uma.pt

1 Introduction

Solving systems of nonlinear equations is possibly the most difficult and challenging problem in all of numerical mathematics [15, 26], and there is not a general method sufficiently efficient and robust for its solution [24].

The difficulties associated with obtaining good numerical approximations to the solutions of nonlinear equation systems are amplified as the number of equations increases. However, it is not hard to find many examples of systems of nonlinear equations with less than ten algebraic or transcendental equations, from different areas of science and engineering, that are difficult to solve adequately by traditional numerical techniques.

In addition, although there are a variety of iterative methods in the extensive literature for the numerical solution of systems of nonlinear equations (see, e.g., the methods and references in [9, 16, 25]) much of them are severely limited by their domains of convergence, and the success of their application is strongly dependent on the initial approximations used. Cases of non-convergence to any of the solutions are relatively frequent in practice.

Nonlinear equations and nonlinear equation systems appear in nearly all simulations of physical processes [17]. An example of this are the physical models expressed mathematically by nonlinear partial differential equations which, when discretized for numerical solution, are transformed into large systems of nonlinear equations, many of them difficult to solve by traditional iterative methods.

In fact, solving systems of nonlinear equations is of fundamental importance in many fields, such as chemistry, economics, electronics, mechanics, robotics, medicine and different branches of engineering [1, 23, 28].

Due to the drawbacks mentioned above, nature-inspired metaheuristic and hybrid approaches, including PSO-based algorithms, have attracted increasing interest in more recent years due to their potential for solving systems of nonlinear equations and other difficult numerical problems. In the particular case of solving nonlinear equation systems, these approaches usually consist in transforming the original problem into a corresponding nonlinear optimization problem [3], and numerically approximating the solutions of this problem by using a pure metaheuristic search algorithm or even a hybrid metaheuristic-based strategy.

When the use of exact optimization algorithms is not possible due to the fact that the problems are large and highly complex, which occurs for example when viewing the problem of solving a large-scale system of nonlinear equations as a multiobjective optimization problem, it is necessary to use stochastic optimization algorithms [30], which allow to find near-optimal solutions within a reasonable execution time.

Nature-inspired metaheuristic optimization algorithms, a large and important class of stochastic optimization techniques [14] that allow the efficient exploration of search spaces and the solving of complex problems with multiple conflicting objectives [4], include swarm intelligence (SI) based algorithms.

These algorithms are characterized by their robustness and the ability to escape from local optima, which although fundamental for any search algorithm, it is not

found in traditional iterative methods of optimization. In addition, the complexities and discontinuities present in the equations that constitute a given system have little or no effect on their search performance.

This paper focuses in one of such SI-based algorithms, known as Particle Swarm Optimization (PSO), for which multiple modifications have been proposed (see, e.g., [11]), some of which specifically to deal with the problem of solving systems of nonlinear equations, whose performance is here compared and reported.

Since the focus of this comparison is the use of PSO variants for solving systems of nonlinear equations, other alterations proposed to the PSO algorithm for generic optimization, such as the one presented in [8] were not implemented.

In the comparison carried out in this study, there are some other PSO variants that could not be used for different reasons. For example, Abraham et al. [2] proposed some alterations to the PSO algorithm to solve Diophantine equations. However, since the solution space for the problems used for testing is different, the results could not be fairly compared. Other two variants, both from [29] were not used since that paper does not mention some of the parameters used, and tests of the implementations could not reproduce the original results.

2 PSO-based Algorithms for Nonlinear Equation Systems

A brief overview of the main PSO-based algorithms for solving systems of nonlinear equations is presented below beginning with a short description of the standard PSO algorithm with inertia weight parameter, followed by a succinct characterization of the essence of the other PSO-based algorithms used in this study for solving nonlinear equation systems.

2.1 Standard PSO

Particle swarm optimization (PSO) was introduced by Eberhart and Kennedy [10, 18]. It is a metaheuristic algorithm which is based on the movement of cooperative groups of animals, such as flocks of birds and schools of fish. There are a number of particles, with position x , each representing a potential solution to the problem. The particles in the swarm move with velocity v through the solution space, according to the following equations (see, e.g., [5, 11]), corresponding to the standard PSO algorithm with the inertia weight parameter introduced by Shi and Eberhart [27]:

$$v_i^{t+1} = w \cdot v_i^t + r_1^t \cdot c_1 \cdot (pbest_i^t - x_i^t) + r_2^t \cdot c_2 \cdot (gbest^t - x_i^t), \quad (1)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1}. \quad (2)$$

The algorithm is subject to three different parameters: w is the inertia weight, c_1 is called the cognitive factor, and c_2 the social factor. These parameters influence how much the movement of each particle i will tend to their best visited position up to the iteration t , $pbest_i^t$ or to the best solution found so far, $gbest^t$. In turn, r_1 and r_2 are vectors of uniformly distributed random numbers. The particles explore the solution space with a tendency to move toward better locations, following the particles closer to the minimum. There is no guarantee that the particles will not prematurely converge to a local minimum, and several modifications have been proposed to tackle this issue.

2.2 HPSO

The hybrid particle swarm optimization (HPSO) algorithm [22] was proposed as a potential improvement over the standard PSO algorithm. It hybridizes PSO with the Nelder–Mead Simplex method. The main goal of this idea was to use PSO's global search capabilities with simplex method local search.

The basic algorithm is similar to PSO but after evaluating each particle, the worst particle is replaced by a new particle using the simplex method. It has the advantage of being a simple hybridization that always improves the overall position of the particles in the swarm by adjusting the worst particle, at a low computational cost. One of the particularities of this approach is that the number of particles must be $N + 1$, where N is the number of variables of the system, so that the simplex can be calculated. Another particularity is the equation used to calculate the inertia weight,

$$w = (w_{\max} - w_{\min}) \cdot \exp\left(\frac{t_{\max} - t}{t_{\max}}\right) - w_{\min}. \quad (3)$$

Instead of using a fixed or linear inertia weight, this variant uses a weight that takes smaller steps towards the minimum as the number of iterations increases.

2.3 PPSO

Since the parameters of the PSO algorithm can change its performance, the Proposed Particle Swarm Optimization (PPSO) [13] chooses to bypass this choice by altering the velocity and position update equations. Using the following equations to calculate the velocity, inertia weight and position for each particle, the performance of the algorithm is not dependent on the social and cognitive factor:

$$v_i^{t+1} = (2r_{1i}^t - 0.5)v_i^t + (2r_{2i}^t - 0.5)(pbest_i^t - x_i^t) + (2r_{3i}^t - 0.5)(gbest^t - x_i^t), \quad (4)$$

$$w^{t+1} = (2r_{4i}^t - 0.5)(gbest^t - pbest_i^t) + (2r_{5i}^t - 0.5)(gbest^t - x_i^t), \quad (5)$$

$$x_i^{t+1} = pbest_i^t + (2r_{6i}^t - 0.5)v_i^{t+1} + (2r_{7i}^t - 0.5)w^{t+1}, \tag{6}$$

where r_{1i}^t to r_{7i}^t are uniform random numbers between 0 and 1.

One notable difference is that the position always has the *pbest* of the particle as a basis, so particles do not stray to non-promising areas of the search space. While this can be an advantage, it also limits exploration.

2.4 *nbest PSO*

The original PSO assumed a *gbest* topology, meaning that, for the purpose of calculating the velocity, the best point found by every particle is considered. However, this is not the only possible topology for PSO. In [10], Eberhart and Kennedy proposed an alternative topology, called *lbest*, where each particle only takes into account the *pbest* of k other particles, usually with $k = 2$. This has the effect of reducing the problem of premature convergence. Thus, the *gbest* topology is a particular case of *lbest*, where k is equal to the number of particles.

What the *nbest* PSO algorithm proposes is a dynamic topology [7], where each particle is in the same neighborhood as the k particles closest to itself. This way, even if some particles get stuck in local minima, some particles are still free to explore the search space. This can also be used to find multiple solutions to the system of nonlinear equations. This method has the disadvantage of being very computationally expensive, as for each iteration, it is necessary to calculate the distance of each particle in relation to each other and select the k closest particles.

There are multiple ways of selecting the number of particles in the neighborhood, from selecting a fixed number to using a formula to calculate the number at each iteration. One equation proposed in [6] sets a minimum and a maximum k . For each iteration t , k is calculated according to the following equation:

$$k_t = \left\lceil \frac{t_{\max} - t}{t_{\max}} \cdot (k_{\text{initial}} - k_{\text{final}}) + k_{\text{final}} \right\rceil. \tag{7}$$

2.5 *imPSO*

Improved PSO (*imPSO*) is a proposal for an improvement on standard PSO for its adapted inertia weight [19]. The inertia weight is calculated according to the following equation:

$$w = a - c \frac{1}{b^{gbest^t}} + d \frac{1}{f^{bfc} + 1}. \tag{8}$$

There is a large number of added parameters to this method. According to [19], a is a value between 0.8 and 1, b should have a value between 1 and 1.5, c a value between 0.6 and 1.2, d should be between 0.05 and 0.2, and f should have a value between 1

and 2.5. $gbest^t$ is the best value found on iteration t and bfc is the standard deviation of the position of the particles. According to the first fraction of the equation, as the $gbest$ tends towards 0 the inertia weight will also decrease. When the particles are spread out, the inertia weight will be larger, but as they converge to a single position, the inertia weight will decrease.

There is another factor associated with imPSO that will be ignored for the purpose of a fair comparison with the remaining algorithms, but must be here mentioned. The probability of getting the optimal value is calculated through multiple simulations in order to calculate the number of restarting times mb .

The algorithm is ran until the number of simulations reaches mb or a solution is found. If the number of simulations reaches mb , the output is “No results”, meaning that a solution does not exist. While this is a good method to achieve reliable results in order to solve a system of nonlinear equations, for the purpose of comparing different approaches—as any algorithm will achieve a better result if it is executed multiple times and only those where it converges to the global minimum are selected—comparisons using imPSO ignore this part of the algorithm.

2.6 APSO–BFA

APSO–BFA is an hybridization between PSO and Bacterial Foraging Algorithm (BFA) [20]. In a similar way to PSO, BFA is a nature-inspired optimization algorithm where every member of a population represents a candidate solution in the solution space. BFA contains three main components: chemotaxis, reproduction, and elimination-dispersal. Chemotaxis is the exploration phase of the algorithm. In APSO–BFA, this corresponds to the usual iteration of the PSO algorithm, where the velocity and position of each particle are calculated, as well as the update of the best values found. This is followed by the reproduction phase, where only the particle with the best value for the fitness function survive and duplicate. This means that the worst particles are removed every few iterations and each remaining particle is copied. Although the particles are copied, the random elements of the velocity equation guarantees that the duplicate particle will not have exactly the same velocity and position as the original one. Finally, there is the elimination-dispersal phase, where bacteria are eliminated with a small probability and replacements randomly initialized. This is the final phase, meaning it is the one that occurs the least number of times. The phases are executed in nested loops, meaning that for each reproduction phase, the chemotaxis phase will occur a fixed number of times and that the reproduction phase will occur a number of times for each elimination-dispersal phase.

PSO–BFA, unlike other PSO variants, does not keep going until a predefined maximum iteration number is reached or the minimum is found. Each phase is repeated several times, with the initial phases being nested inside the later ones. However, it would be simple to make the modification to the algorithm to include stopping criteria at the start of each chemotaxis phase. The equations to calculate velocity and position are the same as standard PSO, but each individual phase of BFA is integrated in this hybridization.

3 Experimental Setting

The aforementioned variants, as well as the standard PSO algorithm with inertia weight, were tested on a number of problems. As each variant was presented with different parameters on their respective papers, these values were followed, meaning that the only parameter in common was the number of particles used, with the exception of HPSO, since the number of particles is dependent on the number of variables of the system. This means that in terms of the raw number of particles, HPSO used much fewer particles. The number of particles for the remaining variants was 100. The number of iterations was always 1000, with the exception of the APSO–BFA, since it uses nested phases. The values of parameters used were the ones used in [20], meaning that the chemotaxis phase occurred 300 times.

3.1 Parameters Used for Each PSO Variant

While choosing the same parameters for each PSO variant was an option when testing the results, it was decided that using the values adopted in each variant’s original article would be a fairer method to compare them, since there was the possibility that each variant performed better with a different set of parameters.

For PSO, the inertia weight was 0.7, the cognitive factor 1.8, and the social factor 1.5. For HPSO, w_{\max} was 0.9 and w_{\min} was 0.4. PPSO had no parameters. Nbest’s inertia weight linearly decreased between 0.7 and 0.1, the cognitive factor and social factor were both 1.4, $k_{\text{initial}} = 5$, and $k_{\text{final}} = 1$. ImPSO’s set of parameters were $a = 1.0$, $b = 1.8$, $c = 1.5$, $d = 0.2$ and $f = 2.0$. APSO–BFA’s inertia weight linearly decreases between 0.9 and 0.4, the cognitive factor was 1.2, and the social factor was 1.8. The number of chemotaxis repetitions for each reproduction phase, Nc was 15. The number of reproduction phases for each dispersal-elimination phase, Nre was 10 and the number of dispersal-elimination phases was 2. During the dispersal-elimination phase, the probability that a particle is randomly deleted, Ped was 0.25.

3.2 Test Problems

The test problems used on each PSO variant were the following:

Problem 1 ([28], Interval arithmetic benchmark i1), $n = 10$.

$$\begin{aligned}
 f_1(\mathbf{x}) &= x_1 - 0.25428722 - 0.18324757 x_4 x_3 x_9 \\
 f_2(\mathbf{x}) &= x_2 - 0.37842197 - 0.16275449 x_1 x_{10} x_6 \\
 f_3(\mathbf{x}) &= x_3 - 0.27162577 - 0.16955071 x_1 x_2 x_{10} \\
 f_4(\mathbf{x}) &= x_4 - 0.19807914 - 0.15585316 x_7 x_1 x_6 \\
 f_5(\mathbf{x}) &= x_5 - 0.44166728 - 0.19950920 x_7 x_6 x_3 \\
 f_6(\mathbf{x}) &= x_6 - 0.14654113 - 0.18922793 x_8 x_5 x_{10}
 \end{aligned}$$

$$\begin{aligned}
 f_7(\mathbf{x}) &= x_7 - 0.42937161 - 0.21180486 x_2 x_5 x_8 \\
 f_8(\mathbf{x}) &= x_8 - 0.07056438 - 0.17081208 x_1 x_7 x_6 \\
 f_9(\mathbf{x}) &= x_9 - 0.34504906 - 0.19612740 x_{10} x_6 x_8 \\
 f_{10}(\mathbf{x}) &= x_{10} - 0.42651102 - 0.21466544 x_4 x_8 x_1 \\
 D &= ([-2, 2], \dots, [-2, 2])^T.
 \end{aligned}$$

Problem 2 ([12], Problem D1—Modified Rosenbrock), $n = 12$.

$$\begin{aligned}
 f_{2i-1}(\mathbf{x}) &= \frac{1}{1 + \exp(-x_{2i-1})} - 0.73 \\
 f_{2i}(\mathbf{x}) &= 10(x_{2i} - x_{2i-1}^2), \\
 i &= 1, \dots, \frac{n}{2} \\
 \mathbf{x}^{(0)} &= (-1.8, -1, \dots, -1.8, -1)^T \\
 D &= ([-10, 10], \dots, [-10, 10])^T.
 \end{aligned}$$

Problem 3 ([12], Problem D2—Augmented Rosenbrock), $n = 12$.

$$\begin{aligned}
 f_{4i-3}(\mathbf{x}) &= 10(x_{4i-2} - x_{4i-3}^2) \\
 f_{4i-2}(\mathbf{x}) &= 1 - x_{4i-3} \\
 f_{4i-1}(\mathbf{x}) &= 1.25x_{4i-1} - 0.25x_{4i-1}^3 \\
 f_{4i}(\mathbf{x}) &= x_{4i}, \\
 i &= 1, \dots, \frac{n}{4} \\
 \mathbf{x}^{(0)} &= (-1.2, 1, -1, 20, \dots, -1.2, 1, -1, 20)^T \\
 D &= ([-10, 10], \dots, [-10, 10])^T.
 \end{aligned}$$

Problem 4 ([12], Problem D3—Powell badly scaled), $n = 12$.

$$\begin{aligned}
 f_{2i-1}(\mathbf{x}) &= 10^4 x_{2i-1} x_{2i} - 1 \\
 f_{2i}(\mathbf{x}) &= \exp(-x_{2i-1}) + \exp(-x_{2i}) - 1.0001, \quad i = 1, \dots, \frac{n}{2} \\
 \mathbf{x}^{(0)} &= (0, 100, \dots, 0, 100)^T \\
 D &= ([0, 100], \dots, [0, 100])^T.
 \end{aligned}$$

Problem 5 ([12], Problem D4—Augmented Powell badly scaled), $n = 12$.

$$\begin{aligned}
 f_{3i-2}(\mathbf{x}) &= 10^4 x_{3i-2} x_{3i-1} - 1 \\
 f_{3i-1}(\mathbf{x}) &= \exp(-x_{3i-2}) + \exp(-x_{3i-1}) - 1.0001 \\
 f_{3i}(\mathbf{x}) &= \varphi(x_{3i}), \\
 i &= 1, \dots, \frac{n}{3}, \text{ where:} \\
 \varphi(t) &= \begin{cases} \frac{t}{2} - 2, & \text{if } t \leq -1 \\ \frac{1}{1998}(-1924 + 4551t + 888t^2 - 592t^3), & \text{if } t \in [-1, 2] \\ \frac{t}{2} + 2, & \text{if } t \geq 2 \end{cases} \\
 \mathbf{x}^{(0)} &= (0, 1, -4, \dots, 0, 1, -4)^T \\
 D &= ([-5, 5], \dots, [-5, 5])^T.
 \end{aligned}$$

Problem 6 ([12], Problem D5—Tridimensional valley), $n = 12$.

$$\begin{aligned}
 f_{3i-2}(\mathbf{x}) &= (c_2 x_{3i-2}^3 + c_1 x_{3i-1}) \exp\left(-\frac{x_{3i-2}^2}{100}\right) - 1, \\
 f_{3i-1}(\mathbf{x}) &= 10(\sin(x_{3i-2}) - x_{3i-1}) \\
 f_{3i}(\mathbf{x}) &= 10(\cos(x_{3i-2}) - x_{3i}),
 \end{aligned}$$

$$i = 1, \dots, \frac{n}{3}, \text{ where:}$$

$$c_1 = 1.003344481605351$$

$$c_2 = -3.344481605351171 \times 10^{-3}$$

$$\mathbf{x}^{(0)} = (-4, 1, 2, 1, 2, 1, 2, \dots)^T$$

$$D = ([-10, 10], \dots, [-10, 10])^T.$$

Problem 7 ([12], Problem D6—Shifted and augmented trigonometric function with an Euclidean sphere), $n = 12$.

$$f_i(\mathbf{x}) = n - 1 - \sum_{j=1}^{n-1} \cos(x_j - 1) + i(1 - \cos(x_i - 1)) - \sin(x_i - 1),$$

$$i = 1, \dots, n - 1$$

$$f_n(\mathbf{x}) = \sum_{j=1}^n x_j^2 - 10000$$

$$\mathbf{x}^{(0)} = (0, \dots, 0)^T$$

$$D = ([-200, 200], \dots, [-200, 200])^T.$$

Problem 8 ([12], Problem D7—Diagonal of three variables premultiplied by a quasi-orthogonal matrix), $n = 12$.

$$f_{3i-2}(\mathbf{x}) = 0.6x_{3i-2} + 1.6x_{3i-1}^3 - 7.2x_{3i-1}^2 + 9.6x_{3i-1} - 4.8$$

$$f_{3i-1}(\mathbf{x}) = 0.48x_{3i-2} - 0.72x_{3i-1}^3 + 3.24x_{3i-1}^2 - 4.32x_{3i-1} - x_{3i} + 0.2x_{3i}^3 + 2.16$$

$$f_{3i}(\mathbf{x}) = 1.25x_{3i} - 0.25x_{3i}^3,$$

$$i = 1, \dots, \frac{n}{3}$$

$$\mathbf{x}^{(0)} = (50, 0.5, -1, 50, 0.5, -1, \dots)^T$$

$$D = ([-5, 5], \dots, [-5, 5])^T.$$

Problem 9 ([12], Problem D8—Diagonal of three variables premultiplied by an orthogonal matrix, combined with inverse trigonometric function), $n = 12$.

$$f_{3i-2}(\mathbf{x}) = 64(x_{3i-2} + x_{3i-1} + x_{3i}) - 0.64 + 0.48 \arctan(x_{3i}) + 0.60(c_1 + c_2x_{3i-1} + c_3x_{3i-1}^2 + c_4x_{3i-1}^3)$$

$$f_{3i-1}(\mathbf{x}) = 0.48 - 48(x_{3i-2} + x_{3i-1} + x_{3i}) + 0.36 \arctan(x_{3i}) + 0.80(c_1 + c_2x_{3i-1} + c_3x_{3i-1}^2 + c_4x_{3i-1}^3)$$

$$f_{3i}(\mathbf{x}) = 0.60 - 60(x_{3i-2} + x_{3i-1} + x_{3i}) + 0.80 \arctan(x_{3i}),$$

$$i = 1, \dots, \frac{n}{3}$$

$$c_1 = 13.901020408163270000$$

$$c_2 = -1.4056122448979600000$$

$$c_3 = -2.2183673469387760000$$

$$c_4 = -0.27704081632653060000$$

$$\mathbf{x}^{(0)} = (10, -5.223, -1.393, 10, -5.223, -1.393, \dots)^T$$

$$D = ([-200, 200], \dots, [-200, 200])^T.$$

Problem 10 ([21], 20—Watson function), $n = 31$.

$$f_i(\mathbf{x}) = \sum_{j=2}^m (j - 1)x_j t_i^{j-2} - \left(\sum_{j=1}^m x_j t_i^{j-1} \right)^2 - 1,$$

$$\begin{aligned}
 t &= \frac{i}{29}, \quad i = 1, \dots, 29 \\
 f_{30}(\mathbf{x}) &= x_1 \\
 f_{31}(\mathbf{x}) &= x_2 - x_1^2 - 1 \\
 \mathbf{x}^{(0)} &= (0, \dots, 0)^T \\
 D &= ([-100, 100], \dots, [-100, 100])^T.
 \end{aligned}$$

Problem 11 ([21], 22—Extended Powell singular function), $n = 12$.

$$\begin{aligned}
 f_{4i-3}(\mathbf{x}) &= x_{4i-3} + 10x_{4i-2} \\
 f_{4i-2}(\mathbf{x}) &= \sqrt{5}(x_{4i-1} - x_{4i}) \\
 f_{4i-1}(\mathbf{x}) &= (x_{4i-2} - 2x_{4i-1})^2 \\
 f_{4i}(\mathbf{x}) &= \sqrt{10}(x_{4i-3} - x_{4i})^2, \\
 i &= 1, \dots, 5 \\
 \mathbf{x}^{(0)} &= (3, -1, 0, 1, \dots, 3, -1, 0, 1)^T \\
 D &= ([-100, 100], \dots, [-100, 100])^T.
 \end{aligned}$$

Problem 12 ([21], 25—Variable dimensioned function), $n = 12$.

$$\begin{aligned}
 f_i(\mathbf{x}) &= x_i - 1, \quad i = 1, \dots, n \\
 f_{n+1}(\mathbf{x}) &= \sum_{j=1}^n j(x_j - 1) \\
 f_{n+2}(\mathbf{x}) &= \left(\sum_{j=1}^n j(x_j - 1) \right)^2 \\
 \mathbf{x}^{(0)} &= \left(1 - \frac{1}{n}, 1 - \frac{2}{n}, \dots, 0 \right)^T \\
 D &= ([-100, 100], \dots, [-100, 100])^T.
 \end{aligned}$$

Problem 13 ([21], 28—Discrete boundary value function), $n = 12$.

$$\begin{aligned}
 f_1(\mathbf{x}) &= 2x_1 - x_2 + h^2(x_1 + h + 1)^3/2 \\
 f_n(\mathbf{x}) &= 2x_n - x_{n-1} + h^2(x_n + nh + 1)^3/2 \\
 f_i(\mathbf{x}) &= 2x_i - x_{i-1} - x_{i+1} + h^2(x_i + t_i + 1)^3/2, \quad i = 2, \dots, n - 1, \\
 \text{where } h &= \frac{1}{n+1} \text{ e } t_i = ih. \\
 \mathbf{x}^{(0)} &= (\varphi_j), \quad \varphi_j = t_j(t_j - 1), \quad j = 1, \dots, n \\
 D &= ([0, 5], \dots, [0, 5])^T.
 \end{aligned}$$

Problem 14 ([31], Example 4.2), $n = 100$.

$$\begin{aligned}
 f_i(\mathbf{x}) &= x_i - \frac{1}{2n} \left(\sum_{j=1}^n x_j^3 + i \right), \quad i = 1, \dots, n \\
 D &= ([-10, 10], \dots, [-10, 10])^T.
 \end{aligned}$$

4 Experimental Results and Discussion

Each of the PSO-based algorithms was run 100 times for each problem and the results were recorded. Table 1 shows the average value for each variant and for each

Table 1 Average fitness for each variant and problem

Probl.	PSO	HPSO	PPSO	nbest PSO	imPSO	APSO-BFA
1	<i>0.0528181</i>	1.2455202	0.1184616	1.0001038	0.0908982	0.3082658
2	2.9814693	43.7396248	4.3049946	19.5092511	63.8314145	<i>0.0532433</i>
3	8.9968749	28.3936415	7.778936	9.9269534	26.5228901	<i>3.0211635</i>
4	146074.6665	18933.73947	102024.748	9636600	1748.411864	<i>7.5670768</i>
5	8.6065558	305.9715265	5.3841644	14.6271948	257.0224481	<i>5.1945554</i>
6	3.9516086	26.9582765	5.5350005	16.1843605	28.7878497	<i>3.4509692</i>
7	<i>12.0141506</i>	5436.467177	14.513556	24.5525433	156.6607548	18.4907909
8	3.1546608	14.3154373	<i>2.5051079</i>	6.8354779	10.9075963	4.7466275
9	55.2570537	9512.331841	97.1277246	9620.411493	355.9871371	<i>52.9168442</i>
10	687.0008116	4111.382325	1203.176166	1979.955606	968.4655094	<i>11.4743482</i>
11	86.4270586	1335.560895	38.544432	615.5957049	1373.850167	<i>0.0000172</i>
12	103.1314993	96.6219912	82.1260091	141.369621	317.349339	<i>0.00001</i>
13	3.6060743	0.3698597	3.723098	6.4950727	7.7535702	<i>0.1959979</i>
14	<i>3.7440374</i>	12.9426592	4.1950816	11.2938565	15.6786825	8.0368876

Table 2 Best fitness for each variant and problem

Probl.	PSO	HPSO	PPSO	nbest PSO	imPSO	APSO-BFA
1	<i>0.0001408</i>	0.3478782	0.0046987	0.3204995	0.0020883	0.041756
2	0.6129596	2.4106494	0.4826853	3.9993273	16.74129	<i>0.0063515</i>
3	<i>1.6164305</i>	3.3723909	2.0379767	5.3806241	9.4492288	3.0000000
4	<i>0.0005094</i>	1674.549033	3.8449885	2800000	8.386666	5.879998
5	1.5390559	5.7505782	<i>1.4736512</i>	2.1391297	20.05712	1.7346438
6	<i>0.4550197</i>	7.5608556	0.7633491	7.1926957	8.8070766	1.7496797
7	<i>3.0020004</i>	21.8360809	4.0097878	12.7831969	47.5687671	4.8217703
8	0.4606752	4.071636	<i>0.1132738</i>	3.080901	1.35673	1.246511
9	15.6940951	138.9986975	30.8773803	454.4669816	79.2954677	<i>10.1141256</i>
10	67.613611	6.5066936	158.6487704	217.3058526	4.5687597	<i>1.7578917</i>
11	9.3435249	11.0110084	0.761295	132.8155348	562.7917561	<i>1.41 e-31</i>
12	33.3304672	14.8876008	22.8491931	69.2140424	197.5205035	<i>0</i>
13	1.0650655	0.2007966	1.3024803	4.8109839	3.1403194	<i>0.1611444</i>
14	<i>0.3559807</i>	6.6422099	0.6392254	4.9785431	7.3584499	1.6831947

problem. Table 2 shows the minimum value found. The minimum values for each problem are shown in bold.

The APSO–BFA variant achieved the best average results on 10 out of 14 problems, performing much better than the alternatives on problems 10, 11 and 12 in particular. However, when we look at the best fitness found, while it did achieve the best result in six problems, the standard PSO with inertia weight had the same number of minimum values found. PPSO had the best performance on problem 8 in particular. HPSO, nbest PSO and imPSO performed consistently worse than the alternatives.

It is interesting to note that the PSO with inertia weight reached the best fitness for many problems, when it did not perform particularly well on average. A possible explanation is that the PSO variants often rely on solving the problem of premature convergence of the PSO algorithm. However, when the particles are near the absolute minimum, this characteristic is a positive one, giving an advantage to the PSO algorithm.

5 Conclusion

Overall, the algorithms were not always successful in solving the nonlinear equation systems. One of the reasons for this is the number of variables in each system, which were considerably higher than the ones present in the variant's original articles, whose tests consisted often of systems with less than six variables.

As such, these results are a good way of comparing the variants in complex, non-trivial problems on a number of particles that was evidently not high enough to reach the solutions for each system.

The most successful PSO-based algorithm was APSO-BFA, which can be explained by being the one that changes the PSO algorithm much more drastically than the other alternatives. In stark contrast, imPSO, which only made slight alterations to the inertia weight, performed poorly, and so did HPSO, which modifies only the worst-performing particle, in addition to the modified inertia weight. The best fitness found for each variant show that the PSO algorithm, while not performing the best on average, finds the best solution at least some of the time, a result of rapid convergence if one of the particles happens to start in a promising position. This also occurs in APSO-BFA, due to the reproduction phase leading to an exploration of a promising local region.

References

1. Abdollahi M, Bouyer A, Abdollahi D (2016) Improved cuckoo optimization algorithm for solving systems of nonlinear equations. *J Supercomput* 72(3):1246–1269
2. Abraham S, Sanyal S, Sanglikar M (2010) Particle swarm optimisation based Diophantine equation solver. *Int J Bio-Inspired Comput* 2(2):100–114
3. Amaya I, Cruz J, Correa R (2011) Real roots of nonlinear systems of equations through a metaheuristic algorithm. *Dyna* 78(170):15–23
4. Blum C, Roli A (2003) Metaheuristics in combinatorial optimization: overview and conceptual comparison. *ACM Comput Surv* 35(3):268–308
5. Bonyadi MR, Michalewicz Z (2017) Particle swarm optimization for single objective continuous space problems: a review. *Evolut Comput* 25(1):1–54
6. Brits R (2002) Niching strategies for particle swarm optimization. M.Sc. thesis, University of Pretoria, Pretoria
7. Brits R, Engelbrecht AP, van den Bergh F (2002) Solving systems of unconstrained equations using particle swarm optimization. In: IEEE international conference on systems, man and cybernetics, Yasmine Hammamet, Tunisia, 6–9 October 2002, vol 3. IEEE, 6 pp

8. Clerc M (1999) The swarm and the queen: towards a deterministic and adaptive particle swarm optimization. In: 1999 congress on evolutionary computation—CEC99 (Cat. No. 99TH8406), vol 3, pp 1951–1957
9. Deufhard P (2006) Newton methods for nonlinear problems. Springer, Berlin
10. Eberhart R, Kennedy J (1995) A new optimizer using particle swarm theory. In: 6th international symposium on micro machine and human science, Nagoya, 4–6 October 1995. IEEE, pp 39–43
11. Freitas D, Lopes LG, Morgado-Dias F (2020) Particle swarm optimisation: a historical review up to the current developments. *Entropy* 22(3):362
12. Friedlander A, Gomes-Ruggiero MA, Kozakevich DN, Martínez JM, Santos SA (1997) Solving nonlinear systems of equations by means of quasi-newton methods with a nonmonotone strategy. *Opt Methods Softw* 8(1):25–51
13. Jaberipour M, Khorram E, Karimi B (2011) Particle swarm algorithm for solving systems of nonlinear equations. *Comput Math Appl* 62(2):566–576
14. Fister I Jr, Yang X-S, Fister I, Brest J, Fister D (2013) A brief review of nature-inspired algorithms for optimization. *Elektrotehniški Vestnik* 80(3):115–122
15. Karr CL, Weck B, Freeman LM (1998) Solutions to systems of nonlinear equations via genetic algorithms. *Eng Appl Artif Intel* 11(3):369–375
16. Kelley CT (1995) Iterative methods for linear and nonlinear equations. SIAM, Philadelphia, PA
17. Kelley CT (2003) Solving nonlinear equations with Newton’s method. SIAM, Philadelphia, PA
18. Kennedy J, Eberhart R (1995) Particle swarm optimization. In: International conference on neural networks, Perth, Australia, 27 November–1 December 1995, vol 4. IEEE, pp 1942–1948
19. Li Y, Wei Y, Chu Y (2015) Research on solving systems of nonlinear equations based on improved PSO. *Mathematical problems in engineering*, 2015, paper 727218
20. Mai X, Li L (2013) Bacterial foraging algorithm based on PSO with adaptive inertia weigh for solving nonlinear equations systems. In: *Advanced materials research*, vol 655–657. Trans Tech Publishers, pp 940–947
21. Moré JJ, Garbow BS, Hillstom KE (1981) Testing unconstrained optimization software. *ACM Trans Math Softw* 7(1):17–41
22. Ouyang A, Zhou Y, Luo Q (2009) Hybrid particle swarm optimization algorithm for solving systems of nonlinear equations. In: 2009 IEEE international conference on granular computing, Nanchang, 17–19 August 2009. IEEE, pp 460–465
23. Pérez R, Lopes VLR (2004) Recent applications and numerical implementation of quasi-Newton methods for solving nonlinear systems of equations. *Numer Alg* 35(2–4):261–285
24. Press WJ, Teukolsky SA, Vetterling WT, Flannery PB (2002) Numerical recipes in C++: the art of scientific computing, 2nd edn. Cambridge University Press, New York
25. Rheinboldt WC (1998) Methods for solving systems of nonlinear equations, 2nd edn. SIAM, Philadelphia, PA
26. Rice JR (1993) Numerical methods, software, and analysis, 2nd edn. Academic Press, Boston
27. Shi Y, Eberhart R (1998) A modified particle swarm optimizer. In: 1998 IEEE international conference on evolutionary computation proceedings. IEEE World Congress on Computational Intelligence (Cat. No. 98TH8360), 1998. IEEE, pp 69–73
28. van Hentenryck P, McAllester D, Kapur D (1997) Solving polynomial systems using a branch and prune approach. *SIAM J Numer Anal* 34(2):797–827
29. Wang Q, Zeng J, Jie J (2007) Modified particle swarm optimization for solving systems of equations. In: *Advanced intelligent computing theories and applications. with aspects of contemporary intelligent computing techniques*. Springer, pp 361–369
30. Weise T, Zapf M, Chiong R, Nebro-Urbaneja A-J (2009) Why is optimization difficult? In: Chiong R (ed) *Nature-inspired algorithms for optimisation*. Springer, pp 1–50
31. Yamamura K, Kawata H, Tokue A (1998) Interval solution of nonlinear equations using linear programming. *BIT Numer Math* 38(1):186–199

Multimedia Immersion System for Band Jumping Training



David Rivas-Lalaleo, Marcelo Alvarez-Veintimilla, Víctor Bautista-Naranjo, Rosa Granizo-López, Pepe Ibañez-Jacome, Hector Lasluisa-Naranjo, Daniel Yanez-Bravo, and Bryan Sandoval-Maiza

Abstract The military training activities are mostly carried out in real environments with the use of firing ranges, obstacle courses, combat zones, among others. In the field of military parachuting, the training is carried out in fixed infrastructure, in which it is instructed with techniques of lateral or posterior jump, driving, and landing, to later move on to the real jump; one of the main limitations in Ecuador is the high cost and logistics of this training, which is why a multimedia immersion system for band jumping is proposed, in this way to contribute to the improvement of the skills of paratroopers using the existing infrastructure in the military units and validating the system with a usability test.

Keywords Multimedia immersion · Virtual reality · Training · Band jumping

D. Rivas-Lalaleo (✉) · M. Alvarez-Veintimilla · V. Bautista-Naranjo · R. Granizo-López · P. Ibañez-Jacome · H. Lasluisa-Naranjo · D. Yanez-Bravo · B. Sandoval-Maiza
Departamento de Eléctrica y Electrónica, Universidad de las Fuerzas Armadas - ESPE Sangolquí, Sangolquí 171103, Ecuador
e-mail: drivas@espe.edu.ec

M. Alvarez-Veintimilla
e-mail: rmalvarez@espe.edu.ec

V. Bautista-Naranjo
e-mail: vrbautista@espe.edu.ec

R. Granizo-López
e-mail: ragranizo@espe.edu.ec

P. Ibañez-Jacome
e-mail: pfibanez@espe.edu.ec

H. Lasluisa-Naranjo
e-mail: hglasluisa@espe.edu.ec

D. Yanez-Bravo
e-mail: dayanez8@espe.edu.ec

B. Sandoval-Maiza
e-mail: bsandoval1@espe.edu.ec

1 Introduction

Technological advances for military tactical training today are one of the areas of greatest evolution and research. This training consists of developing and perfecting the skills and abilities of the combatants that will be used in tactical operations. For several years, these training activities have been carried out using real scenarios, facilities, material, and equipment [2, 6, 9, 10, 15, 16].

In Ecuador, the Army Special Forces are dedicated to training paratroopers in band jump [4]. For infiltration and combat operations, this training is carried out in various scenarios of the national geography, coastal, mountainous, and jungle type. The band jump is performed from the door (side) or from the ramp (rear) of an aircraft. In order to maintain the operation of the paratroopers, at least, one semester training is necessary, for which logistics is required in: mobilization of personnel, material, and equipment in the military detachments of the Ecuadorian territory.

According to data provided by the paratrooper training unit, the cost of a jump in a unit made up of 16 paratroopers ranges from \$35,000 to \$40,000. For this reason, it is evident that the operating costs of these trainings are high, which is why the need to innovate in the training methodology and techniques is generated. Multimedia immersion technology or also known as virtual reality has had a great impact in recent years in the area of entertainment, education, and training [3, 6, 8, 13, 17].

In this context, the present project aims to develop a conceptual test of the multimedia immersion of the band jump, using as resources a virtual environment that represents a side band jump in an Andean area. The tests will be carried out at the Special Forces School No. 9 “Capt. Alejandro Romo Escoba” located in the city of Latacunga, where the infrastructure of the driving simulation tower, virtual reality glasses, and a smart cell phone will be used. The experience is evaluated through a SUS test, obtaining an average of 86.25 points among all users, considering it satisfactory [1, 7, 11, 12].

This work is divided into a methodology section where the materials and methods used are described. The following section presents the progress of the project with which the development of the virtual environment for the paratrooper is explained, followed by an experimentation stage aimed at a group of test paratroopers, with which it generates a section of results based on surveys of usability to the personnel who tested the system and finally closes with the conclusions and future work.

2 Methodology

2.1 Band Jump

It is a jump with a military characteristic, where an elastic band from inside the aircraft stresses, producing the opening of the parachute. The skydiver executes the jump and waits four seconds for the parachute to open, from which he drives the

parachute toward the landing zone. Finally, to land, use 5 contact points that are toes, calves, thighs, buttocks, and lats.

2.2 Band Jump Training

In Ecuador, band jumping training is carried out in the Litoral, Andean, and Amazon regions. These jumps are performed more frequently on slopes and plains due to the wide spaces, areas without obstacles, having the most experienced skydiver can perform the jumps on all types of surfaces.

2.3 Virtual Trainings

The defense field will always require technological innovation to be able to innovate; in the case of the military band jump, a system is proposed that focuses on the sensations of driving a parachute from the jump to the landing of the combatant; immersive virtual reality will achieve this purpose through scenarios and conditions very similar to the real ones; in the same way, the synchronization of the virtual environment with the movements and actions of the personnel in training will offer a high characteristic of realism. The activity begins with the jump of the aircraft where you have the feeling of emptiness; after a few seconds, you feel the templan product of the opening of the parachute, and later, the orientation toward the direction of the wind known as “chest wind” is generated, in a period of no more than one minute using the left and right driving poles and being able to descend to the landing point, where the 5 points of contact necessary to finish the practice will be applied.

3 Virtual Environment

3.1 System Structure

The different processes involved in the virtual environment developed in the Unity3D programming language are based on the functionality defined in the control scripts, which are shown in Fig. 1.

A simulation phase of the scene or virtualized environment is defined, which contains the command controller, the virtual reality program, and the modules of the 3D model which are linked; the objective is to capture all the configurations that define the physical properties and directionality; this allows to simulate the process of driving the parachute in a very similar way to the real one. Regarding the Entry and

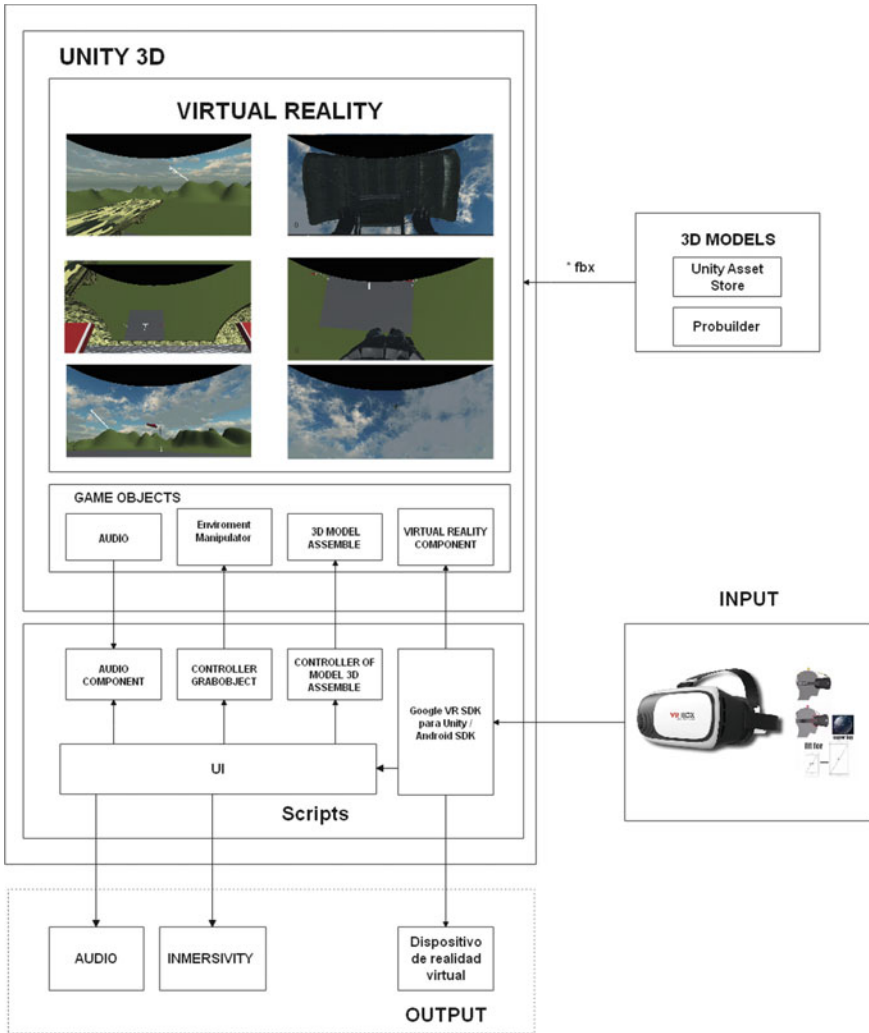


Fig. 1 System structure

Exit phases, they allow flexibility in terms of the devices used when observing the virtual environment, these are: virtual reality helmets (VRBox, HTC VIVE, GearVR, Oculus).

By making a general structure of the code, it allows the use of different input and output devices, without the need to carry out a total reconstruction of the project.

The next stage is called scripts, it is in charge of managing communication with each of the input and output devices; another aspect is to make the virtual environment work, develop the functionality of the interface where the user can select different levels of difficulty where the controller subroutines manipulate the objects in the

environment; with a solid response, these algorithms used are based on hierarchical component models. As the output stage, the one that provides the user with 360° surround audio is described, which encourages a haptic response of inputs, generating a visual feedback of the user's movements within virtual reality.

3.2 *Virtualization*

The design process begins with the generation of the 3D model through the use of computer-aided design software; this tool is based on solid modeling based on operations that takes advantage of the learning facilities of the graphical interface. For example, ProBuilder which is a hybrid 3D modeling and level design tool optimized for building simple geometries, but also capable of providing detailed editing and UV display [14]; on the other hand, it allows to carry out analysis of the movement and the forces that interact on the created models.

The present work proposes a structure composed of several phases which focus on describing the development of applications in virtual environments with the aim of providing greater immersion for military skydiving students in band jumping training.

Phase 1 It focuses on importing the 3D model of the airplane created by computer-aided design software.

Phase 2 It is in charge of determining the reference system and establishes the hierarchies of each of the parts of the 3D model.

Phase 3 Performs the interaction between the virtual reality input devices and the environment where the application is running Fig. 2a.

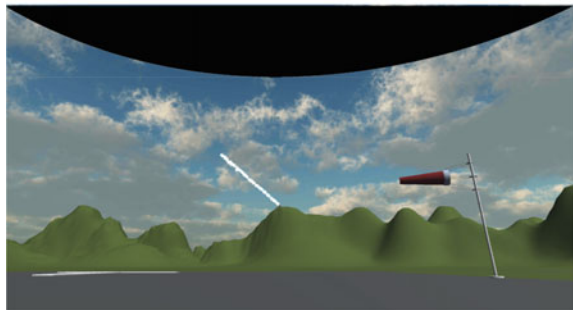
To use 3D models developed in a ProBuilder computer-aided design software embedded in Unity3D, it is necessary to define the hierarchy of the parts of the model [5]. The parts are assembled to obtain a consolidated model, obtaining a single object that can be used in the environment without any problem. It is important to define the position of the model on stage, as well as the rotation and scale, thus defining its orientation, since it must be moved with models of characters inside. The result is an object file compatible with Unity3D which is incorporated into the virtual environment scene where the paratrooper will jump Fig. 2b.

The model virtualization can be seen in Fig. 3, where the parts of the model are described. In the first sub-phase, the configuration of inputs for the interaction of the 3D model with the virtual environment is carried out, and providing the user with immersion using mobile devices and VRBox, the driving is carried out using the gyroscope functions of the SDK and the generation of the wind with the use of the Wired Zone object, Fig. 4a.

For the manipulation of the parachute object, the implementation of the "Open-Parachute" function is carried out, which consists of activating the parachute after having made the jump of the plane, and being activated by the fastening band of the same, Fig. 4b.

Fig. 2 Virtualization

(a) 3d object airplane.



(b) Direction of the wind.

In the second sub-phase, the position of the model is recorded in the virtual environment, in order to navigate through the effect of the transform component of the position, when using the gyroscope in the device. Through this system, the parachute fall control is carried out taking into account the wind direction; said control is configured to respond to the orders that the parachutist gives them Fig. 6.

4 Experimentation

To check the multimedia immersion, the parachute training field of the Special Forces Brigade No. 9 “Capt. Alejandro Romo Escobar” was used, in which the infrastructure called “spider” is used, which consists of a mechanism that allows to maintain suspended in the air to the personnel in training; Fig. 5a shows the structure to be used which is made up of the base for securing the parachute harness, driving poles.

For this test, the personnel are equipped with the corresponding safety equipment and virtual reality glasses, and then the test personnel are suspended in the air, as shown in Fig. 5b.

Finally, the virtual environment is reproduced along with the parachute driving simulation, as shown in Fig. 6.

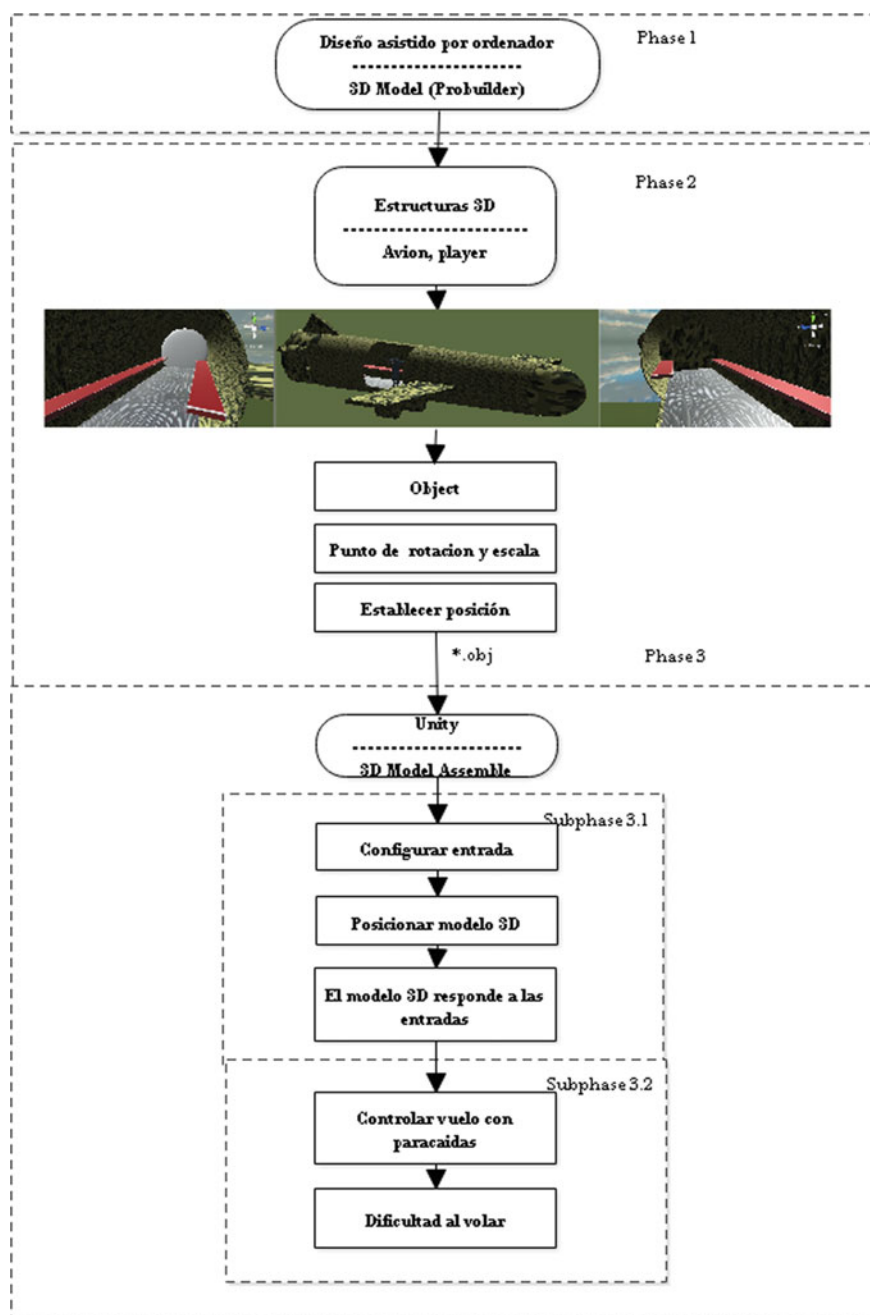


Fig. 3 Model virtualization



(a) Domedrive.



(b) Descentcontrol.

Fig. 4 Virtualization



(a)Preparationofpersonnelfor thetest.



(b)Teststart

Fig. 5 Virtualization

Fig. 6 Running the test

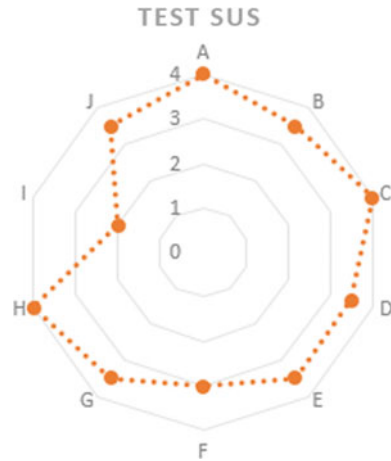
5 Results

A Usability Scales System (SUS) was carried out, of the multimedia immersion system for training in band jumping with 16 paratroopers who have responded 12, obtaining a gross SUS score of 86.25 points, which is above the 68 points required for be accepted and interpreted in a grade of A^- .

The questions that were applied in the survey are:

- A. I think I would like to use this system frequently for my workouts.
- B. I found the application unnecessarily complex.
- C. I thought the system was easy to use.
- D. I think I would need the support of a technician to be able to use this system.
- E. I found the various functions of this system to be well integrated.
- F. I thought there was too much inconsistency in this system.
- G. I imagine that most people would learn to use this system very quickly.
- H. I found the system very complicated to use.
- I. I felt very safe using the system.
- J. I needed to learn a lot of things before I could get up and running with this system.
- K. Give a brief comment about the multimedia immersion system for band jump training.

Fig. 7 SUS radial chart



According to Fig. 7, it is observed with the weighting of the questions:

- I think I would need the support of a technician to be able to use this system.
- I needed to learn a lot of things before I could get going with this system.

Which refer to the level of *dimension of learning capacity* by the respondents, it is indicated that it is in the third quartile and indicates that the system is very intuitive and that it would not need additional technical personnel.

The other eight questions provide the *usability dimension* that are in optimal values over the third quartile, with the exception of “I felt very safe using the system.” This is because the system is demonstrative and they were used commercial VR goggles which do not meet military standards for durability and staff were on the lookout for the goggles not to be dropped or damaged.

6 Conclusions

With the application of these methodology, it has been found that this tool presents an alternative to training paratroopers personnel from Brigade No. 9 “Capt. Alejandro Romo Escobar” reducing operating costs and providing an immersive experience. On the other hand, it is worth highlighting the values obtained in the SUS Test where the perception of users toward the system was weighted at a value of 86 points, considering that the application was developed in a standard environment.

7 Future Jobs

These results obtained open the doors to future research and improvements in the system, such is the case of the use of specialized equipment for virtual reality, the development of environments with greater realism, interaction with the training systems by integrating controls with the training poles. Direction of the parachute, inclusion of emergency maneuvers, and in this way evaluate the learning of the personnel within physical and psychological parameters.

Acknowledgements To the Brigade No. “9” Capt. Alejandro Romo Escobar,” the Universidad de las Fuerzas Armadas—ESPE, and the project’s “Simulador de realidad virtual para entrenamiento de paracaidismo militar de banda” and the “Sistema ciber-físico de cuantificación de impactos para polígono de tiro laser de las Unidades Militares Del Ejército Ecuatoriano” .

References

1. Andaluz VH et al (2016) Virtual reality integration with force feedback in upper limb rehabilitation. Vol. 10073 LNCS. Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics). Cited By: 14. pp 259–268. <http://www.scopus.com>
2. Brand J et al (1999) A virtual reality C3 network battle management and analysis tool. In: MILCOM 1999. IEEE military communications. Conference proceedings (Cat. No.99CH36341), vol 2, pp 1176–1180. <https://doi.org/10.1109/MILCOM.1999.821388>
3. Choensawat W, Sookhanaphibarn K (2019) Aircraft recognition training simulator using virtual reality. In: 2019 IEEE 8th global conference on consumer electronics (GCCE), pp 47–48. <https://doi.org/10.1109/GCCE46687.2019.9015524>
4. Clarke A, Gutman P-O (2020) Actuation strategy of a virtual sky-diver derived by reinforcement learning. In: IFAC-PapersOnLine 53(2). 21th IFAC World Congress, pp 1569–1574. ISSN: 2405-8963. <https://doi.org/10.1016/j.ifacol.2020.12.2187>. <https://www.sciencedirect.com/science/article/pii/S240589632032841X>
5. Expósito KE et al (2018) Realidad virtual en San Cristóbal de La Laguna Patrimonio Histórico
6. Gace I et al (2019) Virtual reality serious game prototype for presenting military units. In: 2019 15th international conference on telecommunications (ConTEL), pp 1–6. <https://doi.org/10.1109/ConTEL.2019.8848505>
7. Kim MG et al (2017) Scaled jump in gravity-reduced virtual environments. In: IEEE transactions on visualization and computer graphics 23(4):1360–1368. <https://doi.org/10.1109/TVCG.2017.2657139>
8. Moshell M (1993) Three views of virtual reality: virtual environments in the US military. *Computer* 26(2):81–82. <https://doi.org/10.1109/2.192003>
9. Naranjo CA et al (2018) Virtual reality system for assistance in treating respiratory disorders. English. Vol. 10851 LNCS. Lecture notes in computer science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Cited By: 2, pp 118–135. www.scopus.com
10. Parreño MA et al (2017) Teaching-learning of basic language of signs through didactic games. In: ACM international conference proceeding series. Cited By: 2, pp 46–51. <http://www.scopus.com>

11. Quevedo WX et al (2017) Assistance system for rehabilitation and valuation of motor skills, Vol. 10325 LNCS. Lecture notes in computer science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Cited By: 9, pp 166–174. <http://www.scopus.com>
12. Rivas D et al (2017) LeSigLa EC: Learning sign language of Ecuador, Vol. 10676 LNCS. Lecture notes in computer science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Cited By: 3, pp 170–179. <http://www.scopus.com>
13. Rivas-Lalaleo D et al (2017) System of evaluation for reading based on eye tracking, Vol. 10676 LNCS. Lecture notes in computer science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Cited By: 3, pp 234–241. www.scopus.com
14. Quevedo Washington X, Sánchez Jorge S, Oscar A, Marcelo A, Zambrano Víctor D, Sánchez Carlos R, Andaluz Víctor H (2017) Sistema de realidad virtual para la formación en mecánica automotriz. In: Conferencia internacional sobre realidad aumentada, realidad virtual y gráficos por computadora
15. Sauk P, Satorius D (2007) Creating a multi-dimensional communication space to improve the effectiveness of 3-D audio. In: MILCOM 2007—IEEE military communications conference, pp 1–7. <https://doi.org/10.1109/MILCOM.2007.4455085>
16. Shao S, Zhou Q, Liu Z (2019) A new assessment method of the pilot stress using ECG signals during complex special flight operation. IEEE Access 7:185360–185368. <https://doi.org/10.1109/ACCESS.2019.2959626>
17. de Souza e Almeida et al R (2020) Topological and tactical study of terrains modeled into a synthetic training environment as a military educational resource. In: 2020 22nd symposium on virtual and augmented reality (SVR), pp 10–19. <https://doi.org/10.1109/SVR51698.2020.00018>

Modeling Simulation of SIR PC Infection Spreading Model with Fuzzy Parameters



M. N. Srinivas, B. S. N. Murthy, M. A. S. Srinivas, and M. Naga Raju

Abstract In this article, we considered SIR epidemic PC infection model with fuzzy parameters. The shortcoming that exists in the PC network by the infection contamination when the resources are revealed requires the examination of the possibility of multiplication of infection into the association. The infection rate, recovery rate, and death rate due to PC infection are all modeled as fuzzy numbers, with their membership functions employed as fuzzy parameters in the model. The fuzzy reproduction number is expressed in terms of an infection load and bifurcation parameter plays an important role in pc spreading model as well as the model's stability discussed at both the infection-free and endemic equilibrium points.

Keywords Personal computer (PC) · Fuzzy parameter · Fuzzy basic reproduction number · Stability

1 Introduction

PC infection is a malignant portable code which including infection, Trojan ponies, Worm, and rationale bomb. A program can duplicate itself and assault different PCs. Furthermore, they are dwelling by eradicating information, harming documents, or changing the ordinary activity. Because of the great similitude between PC infection and organic infection [1], different PC infection proliferation models are proposed. The dynamical demonstrating of the spread course of PC infection is a compelling

M. N. Srinivas (✉)

Department of Mathematics, School of Advanced Sciences, Vellore Institute of Technology,
Vellore 632014, Tamil Nadu, India
e-mail: mnsrinivaselr@gmail.com

B. S. N. Murthy · M. Naga Raju

Department of Mathematics, Aditya College of Engineering and Technology,
Surampalem,, A.P., India

M. A. S. Srinivas

Department of Mathematics, Jawaharlal Nehru Technological University,
Hyderabad, Telangana, India

way to deal with the comprehension of the conduct of PC infections on the grounds that on this premise, some powerful measures can be presented to forestall contamination. The primary pandemic model of PC infections has set up by White [2]. From that point on, diverse pestilence models have been proposed dependent on the completely interconnected suspicion of the web, with center around their complex dynamical properties [3–13]. The finding toward the finish of last century that the web geography asymptotically adheres to a force law degree conveyance has significantly animated the interest in understanding the impact of organization geography on infection spreading, prompting the amazing the interest in understanding the impact of organization geography on infection spreading, prompting the astonishing outcome that an infection could spread across a boundless measured scale free organization in any event, when the disease likelihood is vanishingly little.

The human contaminated disease [14] may burn-through some time-casing to respond inside the body prior to being recognized. Essentially PC infection, for example, email infection likewise burns-through some an ideal opportunity to contaminate the assets confidential the system or organization. This disease discovery period-casing is alluded to as the hatching time system. To identify the presence of infection in specific structure, numerous programming exists on the lookout. The proliferation example of the overall infection can be investigated utilizing which the new infection spread example can be anticipated. The utilization of PC network displaying approach might be considered in dissecting the expansion of infection inside the organization since it is like anthropoid contaminated infection spread. In the writing SIS [15, 16], SIR [17, 18], SIRS [19], SEIRS [20] e-plague models of infection engendering have been created and questioned by the scientists during the new year's utilizing epidemiological methodology. These classes of models have been created by scientists for dissecting the conduct of infection spread in the PC organization.

Numerical models have become more significant instruments for examining the spread of infection in PC organizing systems. Fundamentally, standard differential condition is utilized for plan of this sort of issue and gives some numerical answers and clarification. There exist numerous numerical models in crisp climate on spreading of infection which investigate the spread and control systems of the infection, for example, [21–24]. Because of sensible circumstance, span boundaries or fuzzy boundaries in PC network demonstrating assume a significant part for investigating the infection engendering in the system. Utilization of fuzzy rationale and fuzzy sets in PC network systems has its gigantic potential however they are minimal in number. There are a few utilizations of fuzzy science which can be originate in [25, 26]. The main theme of this paper, we used the fuzzy set hypothesis, which is an extension of the crisp set hypothesis. It also handles the procedures for calculating and deploying fuzzy sets. In writing, there are just a few applications and research publications that use fuzzy logic in the communication of desired goals through a computer network. Figure 1 shows a schematic graph illustrating worm progression in a PC network.

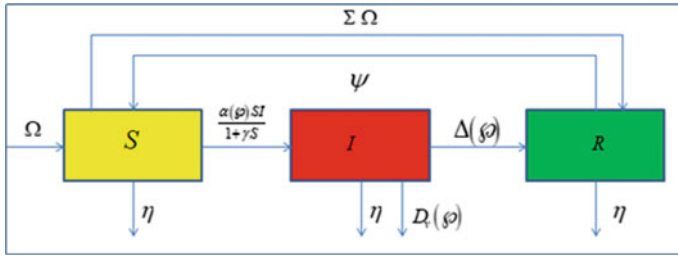


Fig. 1 A schematic diagram of fuzzy SIR flow of worms in PC network

2 Formulation of Fuzzy PC Infection Model

We have considered the classical fuzzy SIR model portray the elements of straight forwardly sent worms with association among susceptible, infected, and recovered nodes in the PC network without neither indispensable elements. Let \wp be the PC infection load in every PC. Presently, the ability to taint in every PC is considered a component of the PC infection load in order to account for variability in the model. As a result, the larger a person’s PC infection burden, the greater the chance of infection transfer through a contact connection. The parameters $\alpha(\wp)$, $D_v(\wp)$ and $\Delta(\wp)$ are determined by taking into account each person’s PC virus load. Table 1 represents the physical interpretation of fuzzy PC infection model.

$$\begin{cases} \frac{dS}{dt} = \Omega - \frac{\alpha(\wp)SI}{1+\gamma S} - \Sigma\Omega - \eta S + \Psi R \\ \frac{dI}{dt} = \frac{\alpha(\wp)SI}{1+\gamma S} - \eta I - D_v(\wp) I - \Delta(\wp) I \\ \frac{dR}{dt} = \Delta(\wp) I + \Sigma\Omega - \eta R - \Psi R \end{cases} \quad (1)$$

3 Analysis of Fuzzy PC Infection Model

The fuzzy notions of infection rate $\alpha(\wp)$, increased death rate due to virus $D_v(\wp)$, recovery rate $\Delta(\wp)$, and virus load $\Theta(\wp)$ are all discussed and interpreted in this section.

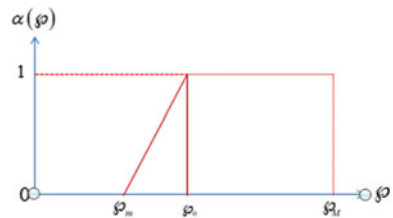
3.1 The Infection Rate’s Fuzzy Membership Function

Let α the probability of transmission between a suspected and an infected computer be proportional to the quantity of computer virus load on each computer. Certain standards of α are extra acceptable than others, transforming alpha into a fuzzy mem-

Table 1 Physical interpretation of fuzzy PC infection model

Parameter	Interpretation	Value
Ω	New no of PC	1
α	Infection rate of infected PC	0.2–1.5
γ	Saturation factor that measure the inhibited effect	0.3
Σ	Immune rate of the PC	0.7
η	Death rate of individual class	0.2
Ψ	Loss rate of immunity of the recovered PC	0.5
\wp	PC infection load	0–1
Δ	Recovery rate of infected	0.05–1
D_v	Death rate due to infection	0.1–1
S	Susceptible nodes	1–100
I	Infected nodes	1–50
R	Recovered nodes	1–50

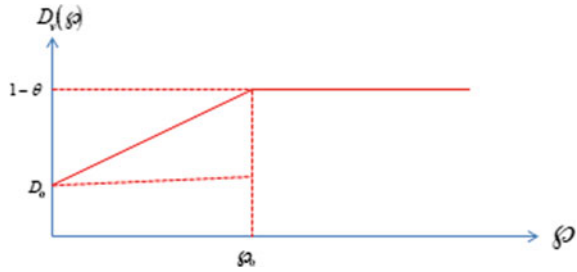
Fig. 2 A membership function of infection rate $\alpha(\wp)$



bership function (Fig. 2). To build the membership function, we assume that if a person’s number of pc infection loads is relatively low, the risk of communication is small. There is also a particular level of pc infection load \wp_0 at which the transmission rate is maximum and equal to one. The number of virus that can be produced is always restricted by \wp_M .

$$\alpha(\wp) = \begin{cases} 0 & \text{if } \wp < \wp_m \\ \frac{\wp - \wp_m}{\wp_0 - \wp_m} & \text{if } \wp_m \leq \wp \leq \wp_0 \\ 1 & \text{if } \wp_0 < \wp < \wp_M \end{cases}$$

Fig. 3 Membership function of additional death rate due to virus $D_v(\wp)$



3.2 Additional Death Rate Owing to Infection Using a Fuzzy Membership Function

The additional death rate, which arises as a result of virus infection (Fig. 3), can also be assumed to be a hazy number. When virus transmission is modest, the virus load is low. There is no virus transmission owing to infection (\wp_0). Also, because \wp is a rising function, when the amount of virus is at its peak ($\wp_0 < \wp$), the death rate is higher. We assume that the greatest additional death value is $1 - \theta$, ($\theta \geq 0$), and that the lowest virus-related additional death rate is $0 < D_0 < 1$.

$$D_v(\wp) = \begin{cases} 1 - \theta & \text{if } \wp_0 < \wp \\ \frac{(1-\theta)-D_0}{\wp_0} \wp + D_0 & \text{if } 0 \leq \wp \leq \wp_0 \end{cases}$$

3.3 Fuzzy Member Function of Recovery Rate

The virus infection group’s recovery rate $\Delta = \Delta(\wp)$ is also expressed in terms of the infection load (Fig. 4). The longer it takes to recover from infection, the larger the infection load, i.e., it is a decreasing function of infected burden \wp .

$$\Delta(\wp) = \begin{cases} \frac{\Delta_0-1}{\wp_0} \wp + 1 & \text{if } 0 \leq \wp \leq \wp_0 \\ \Delta_0 & \text{if } \wp \geq \wp_0 \end{cases}$$

Where Δ_0 is the lowest recovery rate from the infection.

3.4 Fuzzy Member Function of Infection Load

We also suppose that the infection load of the researched group G varies between pc, and hence G can be expressed as a phonological mutable with expert arrangement

Fig. 4 Membership function of recovery rate $\Delta (\Upsilon)$

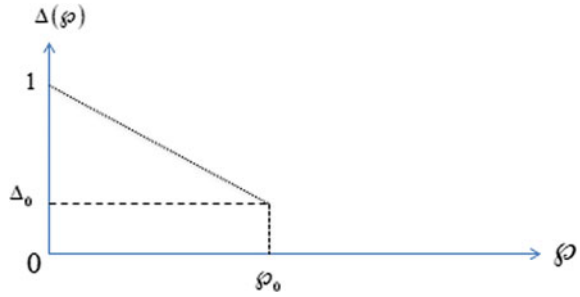


Fig. 5 Membership function of virus load $\Theta (\varphi)$

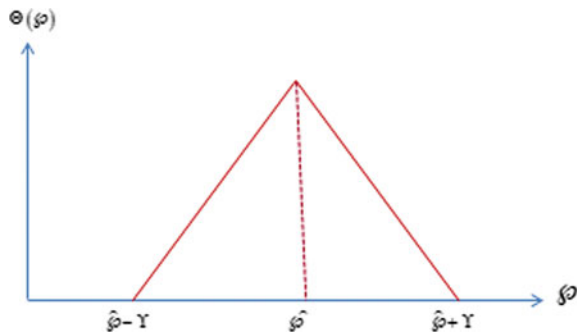
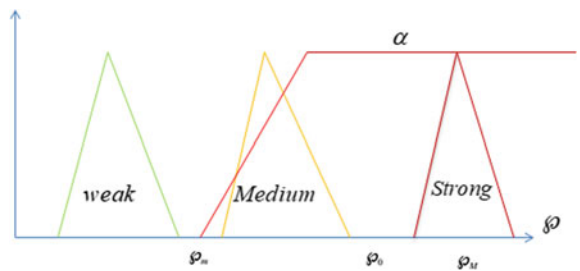


Fig. 6 Classification of linguistic variables



based on the studied group (Fig. 5). The limitation $\widehat{\varphi}$ is a center value that Υ gives the spreading of each of the fuzzy sets presumed by φ , for static $\widehat{\varphi}$, $\Theta (\varphi)$ can have a language implication, such as weak, modern, and high, as determined by an expert (Fig. 6).

$$\Theta (\varphi) = \begin{cases} 0 & \text{if } \varphi < \widehat{\varphi} - \Upsilon \\ \frac{\varphi - \widehat{\varphi} + \Upsilon}{\Upsilon} & \text{if } \widehat{\varphi} - \Upsilon \leq \varphi \leq \widehat{\varphi} \\ -\frac{\varphi - \widehat{\varphi} - \Upsilon}{\Upsilon} & \text{if } \widehat{\varphi} < \varphi \leq \widehat{\varphi} + \Upsilon \\ 0 & \text{if } \varphi > \widehat{\varphi} + \Upsilon \end{cases}$$

4 Bifurcation Parameter and Fuzzy Basic Reproduction Number

The fuzzy basic reproduction number (R_0^f) may shift critical for various irresistible infection yet additionally for similar infection in various networks. The limit after effect of organization hypothesis relates the episodes of plagues and the determination of endemic levels basic reproduction number more prominent than one and the infection vanishes when its worth is not as much as solidarity. The steadiness of infection-free stability changes from temperamental to stable while the basic reproduction number increments through one,; however when $R_0^f = 1$, the system (1) acquires a bifurcation at the contamination free strength point. Presently we consider the bifurcation esteem at \wp^* .

$$R_0^f = \frac{\alpha(\wp)\Omega(1-\Sigma)}{(\eta + D_v(\wp) + \Delta(\wp))(\eta + \gamma\Omega(1-\Sigma))}$$

$$\wp^* = \frac{k_1(\eta^2 + a_1\eta\gamma + a_2(\eta + \gamma a_1) + a_3\eta + a_1 a_3 \gamma)}{a_1 - k_1 a_4(\eta + \gamma a_1)}$$

where $a_1 = \Omega(1-\Sigma)$; $a_2 = \frac{[(1-\theta)-D_0]}{a_0}$; $a_3 = D_0$; $a_4 = \frac{\Delta_0-1}{a_0}$; $k_1 = \wp_0 - \wp_m$.

5 Equilibrium and Its Stability Analysis

In this section, all equilibrium points of system (1) are calculated, namely infection-free equilibrium point (V_0) and the endemic equilibrium point (V_1).

5.1 The Fuzzy Infection-Free Stable Point

The points of equilibrium for infection free are conditions where there is no spread of infection, in particular $I = I^* = 0$. Thus, the infection-free equilibrium point for the fuzzy SIR model (1) is

$$V_0 = (S^*, 0, R^*) = \left(\frac{(1-\Sigma)\Omega}{\eta}, 0, \frac{\Sigma\Omega}{\eta + \psi} \right)$$

5.2 The Fuzzy Endemic Stable Point

The point of equilibrium for endemic is a set of circumstances in which there is a risk of infection spreading, especially $S = S^{**} \neq 0, I = I^{**} \neq 0$ and $R = R^{**} \neq 0$. We get the endemic stable point for the fuzzy SIR model (1) is where $S^{**} = \frac{\eta + D_v(\wp) + \Delta(\wp)}{\alpha(\wp) - \gamma(\eta + D_v(\wp) + \Delta(\wp))}$; $R^{**} = \frac{\Delta(\wp)I^{**} + \Sigma\Omega}{\eta + \psi}$,

$$I^{**} = \frac{\Omega(\alpha(\wp) - \eta(\eta + D_v(\wp) + \Delta(\wp)))(\Sigma\eta - (\eta + \psi)) + \eta(\eta + D_v(\wp) + \Delta(\wp))(\eta + \psi)}{\psi\Delta - (\eta + \psi)(\eta + D_v(\wp) + \Delta(\wp))};$$

Theorem 1 *The virus free equilibrium point $V_0(S^*, 0, R^*)$ of the system (1) is locally asymptotically stable if $R_0^f < 1$ and unstable if $R_0^f > 1$.*

Proof The characteristic equation of the system (1) at infection-free stable point $V_0(S^*, 0, R^*)$ is

$$(-\eta - \mu)(-\eta + \psi) - \mu \left(\frac{\alpha(\wp)S^*}{1 + \gamma S^*} - (\eta + D_v(\wp) + \Delta(\wp)) - \mu \right) = 0 \quad (2)$$

It is easily obtained that the Eq. (2) has two negative eigen values $\mu_1 = -\eta, \mu_2 = -(\eta + \psi)$, the other eigen value of Eq. (2) is $\mu_3 = \frac{\alpha(\wp)S^*}{1 + \gamma S^*} - (\eta + D_v(\wp) + \Delta(\wp))$. According to Routh–Hurwitz criteria, the value of μ_3 is negative if $\frac{\alpha(\wp)\Omega(1 - \Sigma)}{(\eta + D_v(\wp) + \Delta(\wp))(\eta + \gamma\Omega(1 - \Sigma))} - 1 < 0$ i.e., $R_0^f < 1$. Hence, the system (1) is locally asymptotically stable at virus free equilibrium point if $R_0^f < 1$. While $R_0^f > 1$, (2) has positive real roots, which means V_0 is unstable.

Theorem 2 *The endemic equilibrium point $V_1(S^{**}, I^{**}, R^{**})$ is locally asymptotically stable if $R_0^f > 1$.*

Proof The characteristic equation of the system (1) at endemic equilibrium point $V_1(S^{**}, I^{**}, R^{**})$ is

$$\mu^3 + \omega_1\mu^2 + \omega_2\mu + \omega_3 = 0 \quad (3)$$

where $\omega_1 = \eta + \frac{\alpha(\wp)I^{**}}{(1 + \gamma S^{**})^2} + \frac{\alpha(\wp)S^{**}}{1 + \gamma S^{**}} + (D_v(\wp) + \eta + \Delta(\wp)) + \eta + \psi$;

$$\omega_2 = \left(\eta + \frac{\alpha(\wp)I^{**}}{(1 + \gamma S^{**})^2} \right) (\eta + \psi) + \left(\eta + \frac{\alpha(\wp)I^{**}}{(1 + \gamma S^{**})^2} \right) \left(\frac{\alpha(\wp)S^{**}}{1 + \gamma S^{**}} + (D_v(\wp) + \eta + \Delta(\wp)) \right) + \left(\frac{\alpha(\wp)S^{**}}{1 + \gamma S^{**}} + (D_v(\wp) + \eta + \Delta(\wp)) \right) (\eta + \psi);$$

$$\omega_3 = \left(\eta + \frac{\alpha(\wp)I^{**}}{(1+\gamma S^{**})^2} \right) \left(\frac{\alpha(\wp)S^{**}}{1+\gamma S^{**}} + (D_v(\wp) + \eta + \Delta(\wp)) \right) (\eta + \psi) - \frac{\alpha(\wp)\psi I^{**}}{(1+\gamma S^{**})^2} \Delta(\wp) + \left(\frac{\alpha(\wp)S^{**}}{1+\gamma S^{**}} \right) \left(\frac{\alpha(\wp)I^{**}}{(1+\gamma S^{**})^2} \right) (\eta + \psi) ;$$

According to Routh–Hurwitz criteria it follows that all eigen values of Eq. (3) has negative real parts if $\omega_1 > 0$; $\omega_3 > 0$ and $\omega_1\omega_2 - \omega_3 > 0$. Using the conditions $\omega_3 > 0$ and $\omega_1\omega_2 - \omega_3 > 0$ if $\frac{\alpha(\wp)\Omega(1-\Sigma)}{(\eta+D_v(\wp)+\Delta(\wp))(\eta+\gamma\Omega(1-\Sigma))} - 1 > 0$ i.e., $R_0^f > 1$. Hence, the system (1) is locally asymptotically stable at the endemic equilibrium point if $R_0^f > 1$.

6 Controlling Infection in a Complex PC System

We break down the control of the PC infection assessment in the system (1) using the fuzzy basic reproduction number R_0^f , in this segment. In the fuzzy system, infection transmission is determined by the variable, as well as the contact rate, infection-related death rate, and recovery rate. We portray a section of the accompanying cases about the presence and stability of the infection in the system in the accompanying.

Case (i): Minimal infection: If $\widehat{\wp} < \widehat{\wp} + \gamma \leq \wp_m$ the fuzzy basic reproduction number is zero, the infection will disappear from the system.

Case (ii): Moderately infected system: If $\wp_1 > \wp$ then fuzzy basic reproduction number is smaller than unity, the system is infection-free (Theorem 1). The system will become endemic; if $\wp_1 < \wp$, then fuzzy basic reproduction number is bigger than unity (Theorem 5.2).

Case (iii): Infection with a high level of severity: If $\widehat{\wp} > \widehat{\wp} + \gamma \geq \wp_1$ then number of fuzzy reproductions is larger than one, the virus will spread. Currently, the assumption of a fuzzy fundamental reproduction number is linked to infection control policies.

1. Fuzzy basic reproduction number can be reduced by increasing \wp_m
2. Since, $\widehat{\wp} \in (\widehat{\wp}, \widehat{\wp} + \gamma)$, if the amount of medium infection load is very less than the value of fuzzy basic reproduction number value can reduce. For example, by using the antivirus.

7 Numerical Simulation

In this section, the pictorial representation of time series evolution of different nodes for the attributes of Table 1 is carried out in Fig. 7. Figures 8, 9 and 10 represents time series evaluations of nodes at various initial values. It is clearly observed that infected nodes are attains stability and the trajectories are emerged rapidly when compared with other nodes. Next faster emerging trajectories exhibited by recovered

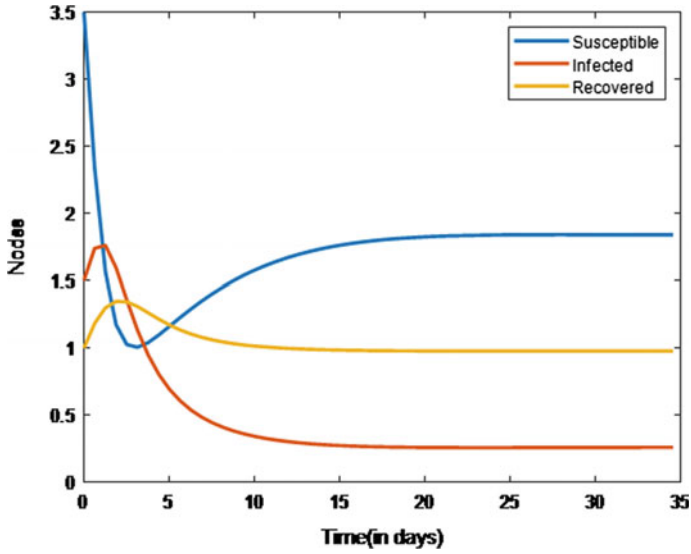


Fig. 7 Represents time series evaluation of susceptible, infected, and recovered nodes against time (in days) with Initial conditions [3.5; 1.5; 1.0] and attributes from Table 1

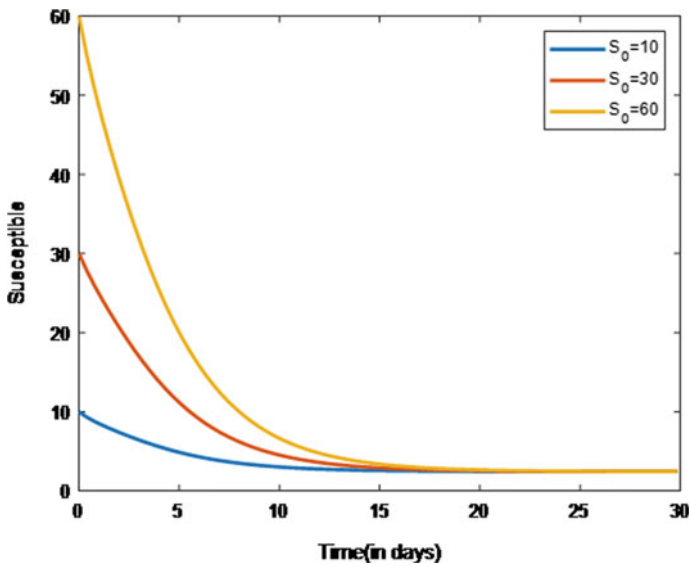


Fig. 8 Represents time series evaluation of susceptible, infected, and recovered nodes against time (in days) with Initial conditions [10; 30; 60] respectively and remaining attributes from Table 1

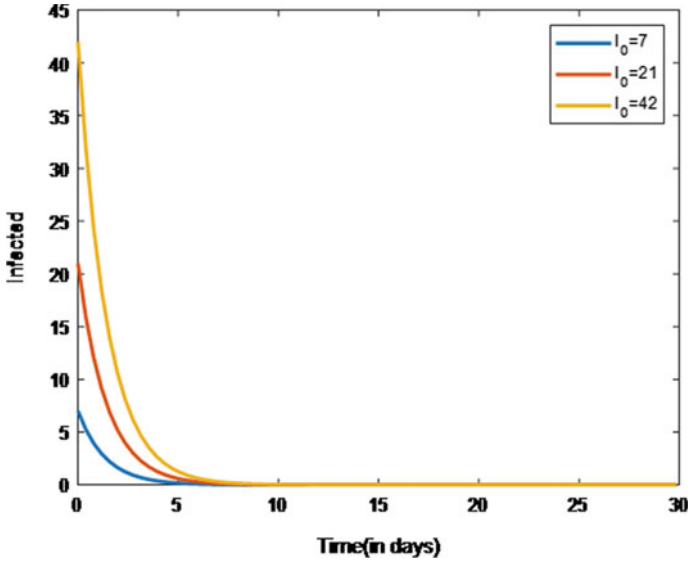


Fig. 9 Represents time series evaluation of susceptible, infected, and recovered nodes against time (in days) with Initial conditions [7; 21; 42] respectively and remaining attributes from Table 1

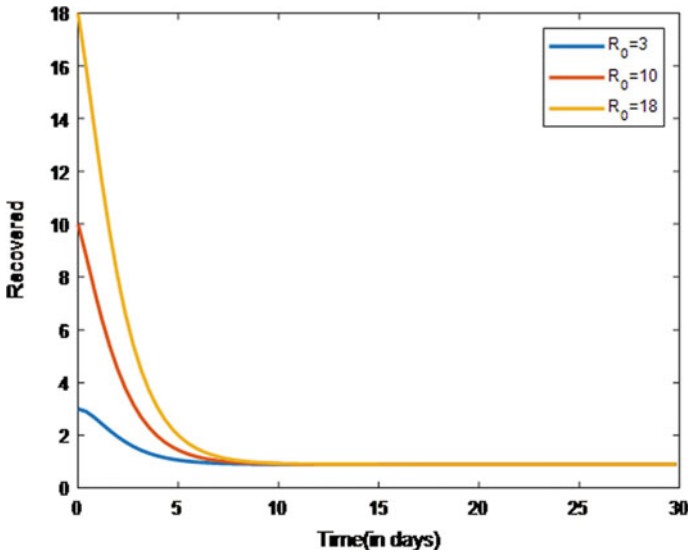


Fig. 10 Represents time series evaluation of susceptible, infected, and recovered nodes against time (in days) with Initial conditions [3; 10; 18] respectively and remaining attributes from Table 1

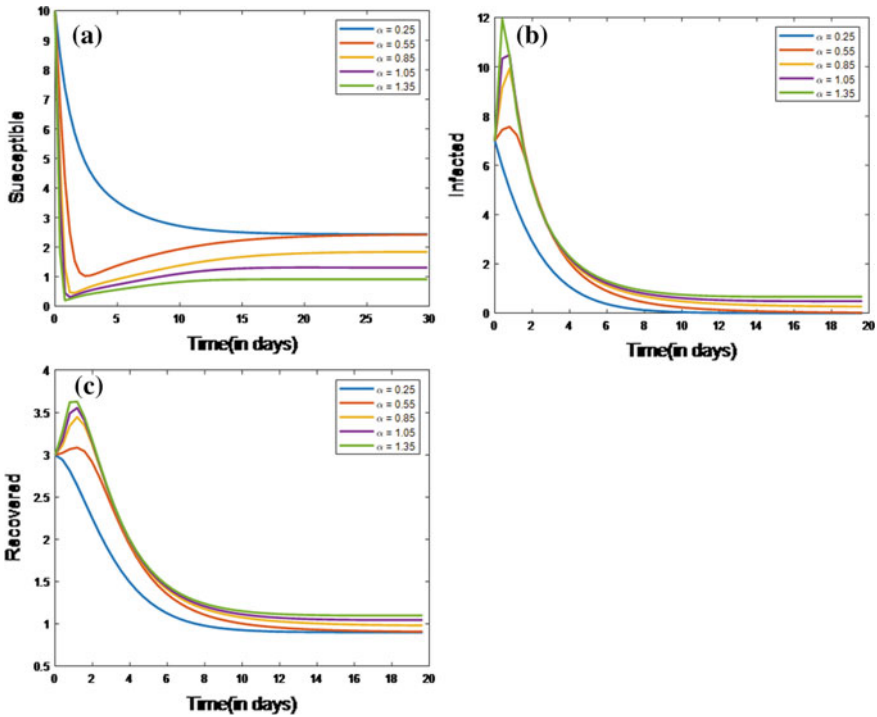


Fig. 11 a, b, c Represents the time series evaluation of susceptible, infected, and recovered nodes for various values of α

nodes. Now coming to susceptible nodes that they exhibit a sustained dynamics for some period before attaining stability. Finally, trajectories emerging for various initial values after certain period.

Later, we tried to explore some of attributes which plays a rich dynamic of the system for various values is demonstrated by Figs. 11a–c, 12, 13 and 14a–c successfully. It is clearly observed that Fig. 11a shows, as α is increasing the susceptible nodes decreasing correspondingly. Figure 11b shows that the infectious nodes are increasing slightly as α , is increasing and Fig. 11c shows that the recovered nodes are also increasing slightly by an increase in the values of α , which tells us that apart from α , some other factors like disguised form of hacking, sudden collapse of fire walls and so on, driving the system slightly more. The susceptible activity observed for various values of α turns rapidly into infectious but not much transmittable and harmful. The roots of infectious in infectious nodes can be traceable, remediable, and recoverable over a period of time and finally the system attains stability which makes the system process smoothen without any interrupts.

It is clearly observed that Fig. 12a shows, as Δ is increasing the susceptible nodes increasing correspondingly. Figure 12b shows that the infectious nodes are decreasing slightly as Δ is increasing and Fig. 12c shows that the recovered nodes are increasing

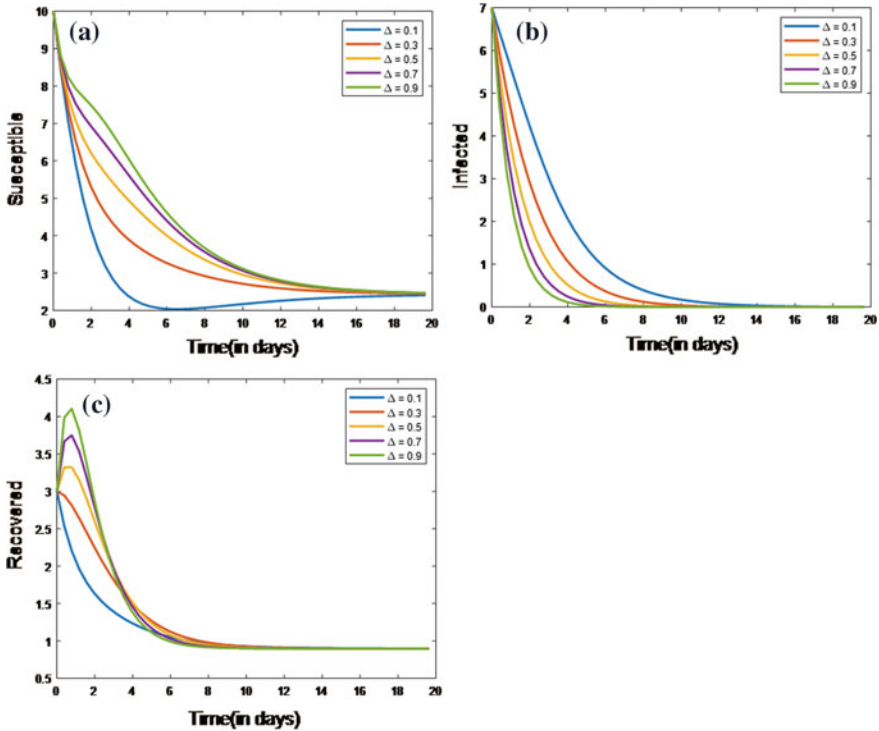


Fig. 12 a, b, c Represents time series evaluation of susceptible, infected, recovered nodes for various values of α and attributes from Table 1

slightly by an increase in the values of Δ , which tells us that for various values of Δ , the susceptible nodes are increasing which results there may or may not be an increase in infectious nodes. Figure 12b shows the decreasing trajectories for increasing values of Δ , which means that the susceptible activity is less effective and not much harmful to the network. It is quite clear that as infectious nodes are decreasing, there will be an increase in recoveries. Figure 12c shows the increasing trajectories for increasing values of Δ , which means that the transmission of infection is slightly less and also not much harmful to the network. The susceptible activity observed for various values of Δ can be detectable, curable, and recoverable over a period of time and finally the system attains stability without any interrupts in the process.

It is clearly observed that Fig. 13a shows, as D_v is increasing the susceptible nodes increasing slightly. Figure 13b shows that the infectious nodes are decreasing as D_v is increasing and Fig. 13c shows that the recovered nodes are increasing as D_v increases, which tells us that for various values of D_v , the susceptible nodes are increasing which results there may or may not be an increase in infectious nodes. Figure 13b shows the decreasing trajectories for increasing values of D_v , which tells us that the susceptible activity is very less effective and almost negligible. It is quite

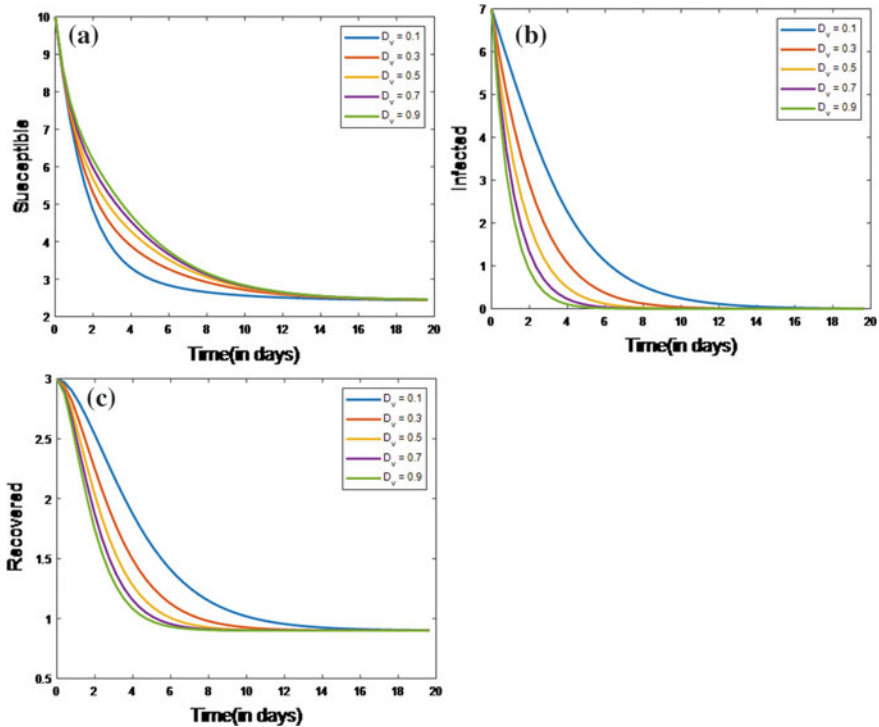


Fig. 13 a, b, c Represents the time series evaluation of susceptible, infected, and recovered nodes for various values of D_v

clear that as infectious nodes are decreasing, there will be an increase in recoveries. Figure 13c shows the increasing trajectories for increasing values of D_v , which means that the transmission of infection is almost negligible as susceptible doesn't turn to infectious which results there is an increase in recoveries exhibited by Fig. 13c. The susceptible activity observed for various values of D_v can be negligible and ignored over a period of time and finally the system attains stability without any interrupts in the process.

Figure 7 represents time series evaluation of susceptible, infected, and recovered nodes against time (in days) with initial conditions [3.5; 1.5; 1.0] and attributes from Table 1 Figs. 8, 9 and 10 represents time series evaluation of susceptible, infected, and recovered nodes against time (in days) with initial conditions [10; 30; 60], [7; 21; 42], [3; 10; 18], respectively, and remaining attributes from Table 1.

Figure 11a–c represents the time series evaluation of susceptible, infected, and recovered nodes for various values of α , Fig. 12a–c represents time series evaluation of susceptible, infected, recovered nodes for various values of Δ and attributes from Table 1.

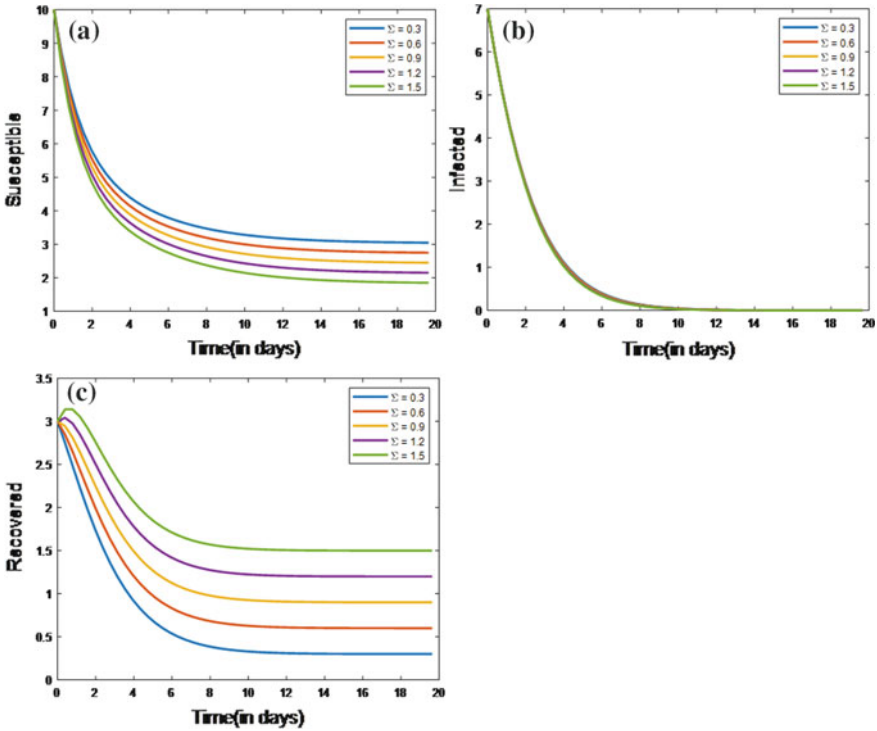


Fig. 14 a, b, c Represents time series evaluation of susceptible, infected, recovered nodes for various values of Σ and attributes from Table 1

Figure 13a–c represents the time series evaluation of susceptible, infected, and recovered nodes for various values of D_v , Fig. 14a–c represents time series evaluation of susceptible, infected, recovered nodes for various values of Σ and attributes from Table 1.

It is clearly observed that Fig. 14a shows, as Σ is increasing the susceptible nodes decreasing slightly. Figure 14b shows that the infectious nodes are emerging and not showing variation as Σ is increasing and Fig. 14c shows that the recovered nodes are increasing as Σ increases, which tells us that for various values of Σ , the susceptible nodes are decreasing which results there may or may not be decrease in infectious nodes. Figure 14b shows the trajectories are emerging and does not affect for increasing values of Σ , which tells us that the system attains stability for various values of Σ as there is no spread of infection and there is an increase in recovered and a decrease in susceptible nodes which maintains the harmony of the system. We may conclude that whenever there is an imbalance in the system, along with traditional recover tools, experimental role of Σ is very crucial to recover it and bring the system to the steadiness.

8 Conclusions

The goal of this research is to apply fuzzy set theory to the nature of PC infection spread in the SIR model, which contracts with infected bulges in the system and the development of protection afterward recovery. We looked examined the impact of three critical fuzzy parameters on the dynamics of the system (1), transmission rate $\alpha(\wp)$, death rate owing to infection $D_v(\wp)$, and recovery rate $\Delta(\wp)$ is expressed in terms of infection load (\wp). The fuzzy basic reproduction number R_0^f is, interestingly, a function of infection load. For $R_0^f < 1$ and $R_0^f > 1$, the points of infection-free equilibrium and endemic equilibrium are asymptotically stable locally. When the number of infected nodes hits the intervention threshold, this model represents a realistic scenario of how the intervention is implemented. The fuzzy influence on PC infection transmission in the SIR model is investigated using theoretical analysis and numerical evaluation.

References

1. Sun C, Hsieh YH (2010) Global dynamics of an SEIR epidemic model with varying population size and vaccination. *Appl Math Model* 34(10):2685–2697
2. Kephart JO, White (1991) Directed graph epidemiological models of computer viruses. In: *Proceeding of the IEEE computer society symposium on research in security and privacy*, pp 343–359
3. Piqueria JRC, Navarro BF, Monteiro LHA (2005) Epidemiological models applied to viruses in computer networks. *J Comput Sci* 1(1):31–34
4. Mishra BK, Jha N (2007) Fixed period of temporary immunity after run of anti-malicious software on computer nodes. *Appl Math Comput* 190(2):1207–1212
5. Mishra BK, Saini DK (2007) SEIRS epidemic model with delay for transmission of malicious objects in computer network. *Appl Math Comput* 188(2):1476–1482
6. Yuan H, Chen G (2008) Network virus epidemic model with the point to group information propagation. *Appl Math Comput* 206(1):357–367
7. Piqueria JRC, de Vasconcelos RR, Gabriel CECJ, Araujo VO (2008) Dynamic models for computer viruses. *Comput Secur* 27:355–359
8. Yuan H, Chen G, Wu J, Xiong H (2009) Towards controlling virus's propagation in information systems with point to group information sharing. *Decis Support Syst* 48(1):57–68
9. Piqueria JRC, Araujo VO (2009) A modified epidemiological model for computer viruses. *Appl Math Comput* 213(2):355–360
10. Han X, Qatar (2010) Dynamical behaviour of computer virus on internet. *Appl Math Comput* 217(6):2520–2526
11. Mishra BK, Jha N (2010) SEIQRS model for the transmission of malicious objects in computer network. *Appl Math Model* 34(30):710–715
12. Zizhen Z, Huizhong Y (2013) Stability and Hopf bifurcation in a delayed SEIRS worms model in computer network. *Math Probl Eng*
13. Ren J, Yang X, Zhu Q, Zhang C (2012) A Novel computer virus model and its dynamics. *Nonlinear Anal Real World Appl* 13(1):376–384
14. Tian-Mu C, Jia R, Qiu-Peng W, Ling Y (2012) A mathematical model for simulating the phase-based transmissibility of a novel corona virus. *Infect Diseases Poverty* 9(24)
15. Mishra BK, Pandey SK (2011) Dynamic model of worms with vertical transmission in computer network. *Appl Math Comput* 217(21):8438–8446

16. Ren J, Yang X, Yang LX, Yang F (2012) A delayed computer virus propagation model and its dynamics. *Chaos Solutions Fract* 45(1):74–77
17. Ren J, Yang X, Zhu Q, Yang LX, Zhang C (2012) A novel computer virus model and its dynamics. *Nonlinear Anal Real* 13(1):376–384
18. Dong T, Liao X, Li H (2012) Stability and hopf bifurcation in a computer virus model with multistate antivirus. *Abstr Appl Anal*
19. Feng L, Liao X, Li H, Han Q (2012) Hopf bifurcation analysis of a delayed viral infection model in computer networks. *Math Comput Model* 56(2):167–179
20. Zhu Q, Yang X, Yang LX, Zhang C (2012) Optimal control of computer virus under a delayed model. *Appl Math Comput* 218(23):11613–11619
21. Mondal PK, Jana S, Halder P, Kar TK (2015) Dynamical behaviour of an epidemic model in a fuzzy transmission. *Int J Uncertain Fuzziness Knowl Syst* 23:651–665
22. Mishra BK, Pandey SK (2010) Fuzzy epidemic model for the transmission of worms in computer network. *Nonlinear Anal Real World Appl* 11:4335–4341
23. Rivera SP, Tiwari RK, Upadhyay (2018) Dynamical behaviours of fuzzy SIR epidemic model. *Adv Intel Syst Comput*
24. Ortega NRS, Sallum PC (2000) Amassed. *Fuzzy Dyn Syst Epidemic Model Kybernetes* 29:201–218
25. Massad E et al (2008) Fuzzy logic in action: applications and epidemiology and beyond. In: *Studfuzz*, vol 232. Springer, Berlin, Heidelberg
26. Barros LC, Bassanzi RC, Leite MBF (2003) The epidemiological models SI with a fuzzy transmission. *Comput Math Appl* 45:1619–1628

CatBoost Encoded Tree-Based Model for the Identification of Microbes at Genes Level in 16S rRNA Sequence



M. Meharunnisa and M. Sornam

Abstract Bioinformatics' classification of gene sequences is an important study area since it allows researchers to undertake genomic analysis and detect potential abnormalities. Genomic sequencing can reveal variants that can lead to disease development or increase the risk of getting sick. It has the potential to allow individuals to act preemptively before a disease is diagnosed. In this paper, two tree-based algorithms, namely decision tree and random forest, are used for the identification of microbes at the gene level in 16S rRNA gene sequence dataset. The survival of all living things depends on ribosomal rRNA. In the evolutionary process of bacteria and other microbes, 16S rRNA is remarkably conserved. At the same time, its conservatism is relative and its length is 1500bp. In this paper, the unequal length of all the sequences is aligned into equal length of 5897bp after performing multiple sequence alignments using MAFFT. The features with the conservation score of greater than 50% were taken into consideration and leaving the "-" resulted in 1447 features in 12,507 sequences. In this work, label encoding, one-hot encoding, and category boosted encoder with tree-based methods for the identification of microbes at gene level are employed. Label encoding converts the nucleotide base A, T, C, G into numbers 0, 1, 2, 3 which might be misinterpreted as orders or hierarchical relationship by some algorithms. This issue is addressed by "one-hot encoding", but the feature size increases drastically and leads to the curse of dimensionality. These problems are addressed by "CatBoost encoding" technique. When it is used along with tree-based methods, the accuracy is not affected as it is same as the other techniques. But the resulted R-square value states that the high percentage of the variance in the dataset is explained by CatBoost encoded tree-based methods. The time taken to execute the model is drastically reduced by using this technique which is basically a time-consuming process for any gene sequencing method. 5×2 statistical test proves that CatBoost encoded random forest is highly significant than other models.

M. Meharunnisa

Department of BCA, Ethiraj College For Women, Chennai, Tamil Nadu, India
e-mail: meharunnisa@ethirajcollege.edu.in

M. Sornam (✉)

Department of Computer Science, University of Madras, Chennai, Tamil Nadu, India

Keywords 16S rRNA sequences · Categorical boosting encoder · Decision tree · Multiple sequence alignments · Random forest

1 Introduction

Bioinformatics is a branch of study that combines biology, computer science, and information technology to evaluate biological data using computers and statistical tools. Biological experiments can generate large amount of data (macromolecular sequences, structures, expression profiles, pathways, etc.). Dauer [1] presented a detailed study to understand the ontological dimensions of complexity, components such as functional relationships, processes, manifestations, and interpretation are independently evaluated while studying the notion of gene expression. DNA and RNA sequences, ribosomes, and proteins are all parts of the system. Different biological systems have different ways to develop. So, start with the DNA nucleotide sequence and then proceed with the nucleotide structure and with the gene expression. In order to get the protein sequence, RNA and DNA are needed, after that, into the next level, which is called protein structure, and then, for cell signalling and protein interactions. Through the various applications of bioinformatics, it is possible to understand the complexity of biological systems. A lot of sequences are stored in databases for analysis. In other words, they are stored in a huge number of databases. The huge amount of data created by sequences is one of the most important characteristics of genomics as a data science. The total volume of data generated in genomics is far less than that generated in earth science, but orders of magnitude greater than that generated in the social sciences. However, the rate of data expansion in genomics is faster than in other fields. Stephen et al. [2] believe that if the current rate of genomic data collection continues, genomics could eventually generate more data than social media, earth sciences, and astronomy combined. Shadab et al. [3] proposed a model for the identification of DNA-binding proteins which is helpful in the early detection of viruses that can prevent outbreaks like COVID-19 and aid in drug development. As a result, DNA sequence categorisation is extremely important in computational biology.

Blanco and Blanco [4] demonstrated that nucleic acids (NA) are fundamental for protein generation since they contain hereditary information. Nucleotides, which are made up of a nitrogenous base, an aldopentose, and phosphoric acid, are polymerised to deliver them. Thymine (T), cytosine (C), and uracil (U) are pyrimidine bases, whereas adenine (A) and guanine (G) are purines (G). RNA could be a polynucleotide that contrasts from DNA in that it as it was and has one chain instead of two, and it contains ribose instead of deoxyribose and uracil instead of thymine. Messenger RNA (mRNA) transmits hereditary data from the cell core to the cytoplasm; exchange RNA (tRNA) transports amino acids to the location of protein blend. Fadeev et al. [5] concluded that the short ribosomal subunit (16S rRNA) gene is frequently used to investigate the taxonomic composition of microbial communities in ecology. Putri et al. [6] demonstrated the three types of rRNA in prokaryotic organisms: 23S rRNA (S

= Svedberg units; 2900 nucleotides), 16S rRNA (1550 nucleotides), and 5S rRNA (120 nucleotides). The 16S rRNA molecule is the most widely used of the three rRNA molecules. The 16S rRNA molecule covers a collection of genetic data and is easy to research. Mazzarelli et al. [7] found that 16S rRNA gene sequencing indicated an altered gut microbiota composition in COVID-19 pneumonia patients hospitalised to the intensive care unit. (ICU).

Bukhari et al. [8] built an ensemble-based prediction model for identifying epitope and non-epitope to generate vaccines for Zika virus infection using support vector machine (SVM), random forest (RF), decision tree (DT), neural network (NN), and AdaBoost classifiers (Ada). To guarantee consistency, the model was subjected to fivefold cross-validation, with an average accuracy of 96.072% recorded. The suggested ensemble model will aid in the prediction of innovative ZIKV vaccine candidates in order to protect people worldwide and avert increased incidence breakouts in the future.

Zhou et al. [9] used category encoding method for bulk RNA-seq and scRNA-seq data classification, and the feature genes are selected using a category encoding strategy. To efficiently classify scRNA-seq data, a two-step approach is used. To begin, each gene's samples is scored in order to create an ordered category. Second, using the sample rank for each gene in each class, the category is re-encoded. The proposed category encoding (CAEN) technique not only modifies the order of class labels to make feature genes correspond to class labels, but it also overcomes the problem of class "ties", which arises when several observations are assigned to the same common class label. The selection of important feature genes considerably increased categorisation accuracy.

Deep learning (DL) models may extract meaningful parameters from input. Gunasekaran et al. [10], CNN, CNN-LSTM, and CNN-bidirectional LSTM architectures with Label and K-mer encoding were used to categorise DNA sequences. The models are assessed using a variety of classification metrics. The CNN and CNN-bidirectional LSTM with K-mer encoding exhibit good accuracy, according to the trials, with 93.16% and 93.13% on testing data, respectively.

Concluding the detailed discussion on the different types of encoding methods on DNA/RNA sequence, the remainder of the paper is described as follows. The proposed research technique is presented in Sect. 2. The RNA encoding schemes for machine learning are presented in Sect. 3. Section 4 compares and contrasts the proposed model with traditional tree-based models utilising various encoding approaches. Finally, in Sect. 5 of this research paper, the conclusion is reached.

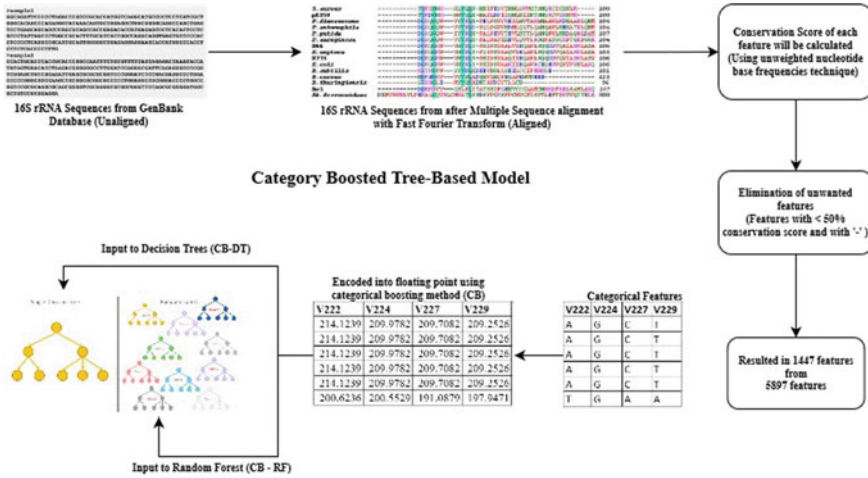


Fig. 1 Proposed tree-based model architecture

2 Research Methodology

2.1 Proposed Method

The following is a list of stages that must be completed in order to complete the suggested category boosted encoding method with tree-based model architecture . Figure 1 illustrates the diagrammatic representation of the proposed architecture.

- (i) File that contains the 16S rRNA sequence of microbes is extracted from the GenBank database. The metadata information about the dataset is presented in Table 1.
- (ii) Multiple sequence alignments with fast Fourier transformation are used to align the sequences.
- (iii) Using an unweighted nucleotide base frequencies technique, the conservation score of each feature is determined.
- (iv) The unwanted data such as column with gaps “-” and the columns with the conservation score of less than 50% are eliminated resulting in 1447 features.
- (v) The categorical features are converted into floating point values using category boosting method.
- (vi) These floating point values are then used to train tree-based models, which are then validated using tenfold cross-validation scores.
- (vii) To examine the proposed category boosted tree-based model’s effectiveness (CB-TB), an extensive comparative study made with label encoding and one-hot encoded tree-based model.

Table 1 Schema of the dataset

Column name	Type
SequenceID	String
Sequence description	String
RNA sequence	String
Length of the sequence	Number

Table 2 Sample class distribution of 16S rRNA sequence w.r.t genes level

Genes	Number of samples
Streptomyces	1168
Bacillus	356
Paenibacillus	343
Lactobacillus	285
Pseudomonas	267
Nocardia	223
Vibrio	207
Clostridium	200
Flavobacterium	196
Streptococcus	164
Sphingomonas	147
Corynebacterium	117
Mycobacterium	117
Microbacterium	116
Halomonas	114

- (viii) The proposed CB-TB achieves the highest performance with highest R2 values, which explains the relationship between the variables in the dataset with less execution time.
- (ix) Using 5x2 statistical test, it is concluded that CB-random forest is highly significant than CB-DT.

2.2 Data Collection

The entire DNA/genomic sequence of 16S rRNA was retrieved from the “National Center for Biotechnology Information (NCBI)” public nucleotide sequence database. The annotation for the DNA sequencing data is provided in Table 1.

The sequences are between 459 and 1833 nucleotides in length. This paper focuses on the prediction of genus of the microbes based on the conserved region of the sequence. Therefore, the class distribution is based on the number of samples with respect to genus level which is shown in Table 2.

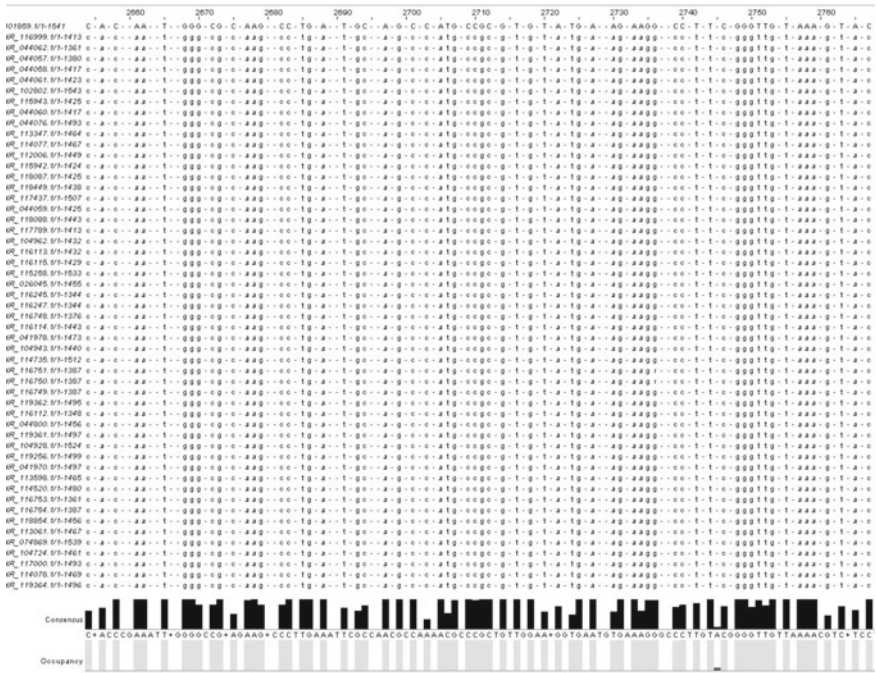


Fig. 2 The few Positions of the 16S rRNA multiple sequence alignment of instances of bacteria. Generated with MAFFT

Figure 2 shows sample DNA sequences from the collection that include the whole genomic sequence of a bacteria, as well as the length of the sequence and the class labels.

2.3 Data Pre-processing Using Multiple Sequence Alignments

The method of aligning numerous biological sequences is known as multiple sequence alignments. In many cases, the resulting sequence has an evolutionary relationship with a common ancestor by certain evolutionary events such as insertion, deletion, mutation, and rearrangement that occur under certain conditions [11]. Given n sequences $X_i, i = 1, \dots, n$ similar to the form below:

$$X = \left\{ \begin{array}{l} X_1 = (x_{11}, x_{12}, x_{13}, \dots, x_{1m_1}) \\ X_2 = (x_{21}, x_{22}, x_{23}, \dots, x_{2m_2}) \\ X_3 = (x_{31}, x_{32}, x_{33}, \dots, x_{3m_3}) \\ \vdots \\ X_n = (x_{n1}, x_{n2}, x_{n3}, \dots, x_{nm_n}) \end{array} \right\} \tag{1}$$

This set of sequences X is aligned using multiple sequence alignments by introducing any number of gaps into each of the X_i sequences of X until the changed sequences, X_i' , all conform to length $L \geq n_i | i = 1, \dots, m$ and no values in the sequences of X of the same column consist of only gaps. The following is the mathematical form of an multiple sequence alignments of Eq. (1):

$$X' = \left\{ \begin{array}{l} X'_1 = (x'_{11}, x'_{12}, x'_{13}, \dots, x'_{1m_1}) \\ X'_2 = (x'_{21}, x'_{22}, x'_{23}, \dots, x'_{2m_2}) \\ X'_3 = (x'_{31}, x'_{32}, x'_{33}, \dots, x'_{3m_3}) \\ \vdots \\ X'_n = (x'_{n1}, x'_{n2}, x'_{n3}, \dots, x'_{nm_n}) \end{array} \right\} \tag{2}$$

The most commonly used multiple sequence alignments algorithms are ClustalW [12], Clustal Omega [13], MUSCLE [14], MAFFT [16], T-Coffee [15], and Prob-Cons [17].

2.4 Elimination of Unwanted Data by Calculating Conservation Score

Conserved sequences are similar or identical sequences found in nucleic acid or protein sequences in biology. The evolutionary pace of a place is reflected in its conservation score. The rate of evolution of nucleotide sites is not constant: some places evolve slowly and are known as “conserved”, while others evolve quickly and are known as “variable”.

Each column in Table 3 represents a nucleotide location in an MSA. The sequence number of a particular 16S rRNA sequence is indicated in the row. Nucleotides are designated by a one-letter code, while gaps are denoted by a “-.”

Valdar [18] proposed a mathematical score that measures the degree of conservation at an aligned site must satisfy the following characteristics:

- (a) The score should be a function that fulfils certain mathematical properties.
- (b) The relative frequency of a nucleotide in a column.
- (c) Recognise conservative replacements.
- (d) Penalise the gaps.

Table 3 Example columns after multiple sequence alignments

		Columns					
		(a)	(b)	(c)	(d)	(e)	(f)
Sequences	1	A	A	C	G	C	T
	2	A	A	C	G	C	T
	3	A	A	C	G	C	T
	4	-	A	C	G	C	T
	5	A	A	C	G	C	T
	6	A	A	C	G	C	T
	7	A	A	C	G	C	T
	8	-	-	-	-	-	-
	9	A	A	C	G	C	T
	10	A	A	C	G	C	T

- (e) Without compromising evolutionary information, normalise against redundancy and bias.
- (f) Should not be more complex.

2.5 Calculation of Conservation Score

Valdar [18] proposed a method for measuring sequence conservation. The method is as follows:

$$SC = \frac{x}{nx} \times NS \tag{3}$$

where x denotes the number of amino acid types in a column, nx is the number of times the most common amino acid symbol appears in the column, and NS denotes the number of scoring sequences. To overcome the problem with [18], a new scoring scheme was presented by Wu et al. [19]. This score is calculated by dividing the number of different pairs by the total number of amino acids in a column. The scoring system is as follows:

$$SC = \frac{x_{\text{pairs}}}{y_{\text{pairs}}} \times \frac{1}{2} NS(NS - 1) \tag{4}$$

where $1/2NS(NS - 1)$ denotes the overall number of possible protein pairings in a column, x_{pairs} pairs denote the number of unique pairs, and y_{pairs} specifies how often the most frequent pairs appear.

A different scoring method is proposed by Jores et al. [20] to evaluate the conservation of an amino acid. If a symbol A first appears in the sequence database with a frequency of q_A , then the probability of setting it n_A times in a column of K

Table 4 Sample columns with its conservation scores

Position	Nucleotide	Score
2728	–	99.84
2729	–	99.95
2730	T	92.88
2731	G	99.73
2732	–	99.95
2733	A	100
2734	C	62.51
2735	G	99.95
2736	G	100
2737	–	100
2738	–	99.95
2739	C	72.58
2740	C	71.72
2741	–	100

residues is $P(X = n_A)$. In this paper, to identify the conserved region, unweighted position-specific nucleotide frequencies were calculated using Eq. 5.

$$f_a^u(i) = n_a(i)/n(i) \quad (5)$$

where $n_a(i)$ —nucleotide “ a ” occupies position “ i ” in the following number of sequences. $n(i)$ —total number of sequences aligned = $\sum n_a(i)$, $i = 1, 2, 3, \dots, 12,507$. Table 4 shows the sample columns of the 16S rRNA sequence with its conservation scores.

From Table 4, the columns which consist of “–” with 100% conservation score have been removed completely and the columns with the conservation score greater than 50% have been taken for study which resulted in 12,507 sequences with 1447 columns or features to input the model.

3 RNA Encoding Schemes for Machine Learning

DNA/RNA storage has evolved significantly over time. The latest generation of encodings provides a solid basis for long-term data storage. A DNA storage framework should be able to handle the cost of sequencing and synthesis, as well as the recovery and storage of information. It is not advisable to give values to DNA genome data arbitrarily because this could cause feature learning to fail or need data normalisation/regularisation methods. However, the normalisation/regularisation procedure does not always ensure that the deficiencies in data representation are compensated.

Yu et al. [21] developed a boosting feature learning performance with an appropriate encoding technique that ensures normal distribution is usually desirable, and this was only employed before the development of normalisation/regularisation algorithms. It also worked well in this study while dealing with nucleotides A, C, T, and G in RNA data. Nguyen et al. [22] employed convolutional neural network model which significantly improved the performance in all testing datasets by employing one-hot vectors to represent DNA sequences and applying a convolutional neural network model.

Hancock and Khoshgoftaar [23] used label encoding, which is a technique that automatically assigns an integer value to a categorical variable. It can be used to convert values that are drawn from a set to numerical values. The procedure of assigning an integer value to each of a categorical variable's potential values is known as label encoding. If a dataset contains a categorical variable with values from the categories "apple", "banana", and "orange", label encoding could very well assign mapped values from the ranges "0", "1", "2", and so on. There is not a more straightforward approach to convert attribute variables to numerical ones. Label encoding is the least effective strategy. Label encoding is expected to perform poorly because it introduces artificial order among categorical data.

In this paper, CatBoost encoding technique is introduced to encode the features of the dataset because the disadvantages of using one-hot encoding are that for large cardinality, and the feature space might quickly expand and will be fighting the curse of dimensionality. CatBoost is a category encoder that outputs columns according to a target value. It does so by estimating the values based on the binomial and continuous objectives. A CatBoost encoder works in a similar way to a target encoder, but it additionally uses an ordering approach to avoid target leaking. The target statistic's values are based on the observed history, i.e. the target probability for the current feature is computed only from the rows before it. Categorical values are encoded using Eq. 6

$$\text{Encoded_Value} = \frac{TS + P}{FC + 1} \quad (6)$$

where TS is the sum of the target value for that particular categorical feature in the entire RNA dataset.

P is a constant prior value determined by (sum of target value)/(total no. of sequences in the dataset), and FC is the total number of category features seen with the same value as the present one up to this point. The goal statistics for the first few observations in the sample are always substantially larger variance than those for subsequent observations. To reduce this effect, target statistics are produced using several random permutations of the same data, and the final encoding is derived by averaging across these variants. The final encoded values obey Eq. 6 if the number of permutations is large enough. Table 5 shows the sample columns before and after CatBoost encoding.

Table 5 Category Boosted Encoder

(a) Before CatBoost Encoding				(b) After CatBoost Encoding			
A	-	C	T	214.43	209.83	208.456	207.781
G	A	-	A	214.123	200.552	209.708	198.383
C	T	T	A	191.085	207.787	198.786	207.788

4 Conventional Tree-Based Models

The position of the nucleotides whose conservation score is greater than or equal to 50 was taken into consideration for feature extraction and was used to train and test the tree-based methods, namely decision trees and random forest. This resulted in 1447 feature vectors. Before the feature vectors are fed into the machine learning algorithms, the usage of encoding process is crucial. Since all the feature vectors are categorical, performing one-hot encoding is a crucial process for large dataset because it adds still more dimensionality. From the perspective of machine learning, Cerda et al. [24] concluded that it is not a good choice for encoding categorical variables. Other categorical encoding strategies have been investigated in the field of machine learning in addition to one-hot encoding [25–27].

For supervised learning tasks, tree-based algorithms are a common machine learning method. Tree-based models produce predictions from one or more decision trees using a sequence of if–then rules. All tree-based models can be used for classification or regression models. The approach of selecting the tree structure is a greedy one. In each leaf, features are chosen in sequence, along with their splits, for substitution. Candidates are chosen based on information obtained from preliminary split calculations and the conversion of categorical to numerical attributes. Hussain et al. [27] proposed categorical boosting (CatBoost) encoding technique which is used to convert categorical features to numerical characteristics. CatBoost is distinct from other encoding methods in that it has two distinct features: efficient categorical feature processing and ordered boosting [28].

Both classification and regression problems can be solved with decision trees. The name implies that it uses a tree-like flow chart to display the predictions that result from a sequence of feature-based splits. When using decision trees for knowledge discovery, domain experts should usually be involved in the analytic process. As a result, domain experts will almost always analyse the final decision trees for extracted knowledge—that is, rules that can be derived from a decision tree [29]. It begins with a root node and finishes with a leaf decision. Building a decision tree model essentially consists of two steps: deciding which features to split and when to stop dividing. The amount of uncertainty in the dataset is measured by a statistic called “entropy”, which is used to make this decision. Stiglic et al. [29] have released visual tuning of decision tree (VTDT), a new decision tree version that allows users to create effective decision tree representations while taking less time adjusting decision tree

Fig. 3 Visualisation of rules generated by decision tree for 16S rRNA sequence

```
|--- feature_896 <= 274.39
|   |--- feature_548 <= 200.51
|   |   |--- feature_1105 <= 236.54
|   |   |   |--- feature_1063 <= 269.52
|   |   |   |   |--- feature_1094 <= 173.50
|   |   |   |   |   |--- feature_200 <= 194.65
|   |   |   |   |   |   |--- feature_567 <= 176.61
|   |   |   |   |   |   |   |--- class: 326
|   |   |   |   |   |   |   |--- feature_567 > 176.61
|   |   |   |   |   |   |   |   |--- feature_946 <= 172.88
|   |   |   |   |   |   |   |   |   |--- class: 172
|   |   |   |   |   |   |   |   |   |--- feature_946 > 172.88
|   |   |   |   |   |   |   |   |   |   |--- class: 180
|   |   |   |   |   |   |   |--- feature_200 > 194.65
|   |   |   |   |   |   |   |   |--- feature_1361 <= 184.44
|   |   |   |   |   |   |   |   |   |--- feature_139 <= 201.43
|   |   |   |   |   |   |   |   |   |   |--- feature_1092 <= 203.55
|   |   |   |   |   |   |   |   |   |   |   |--- class: 186
```

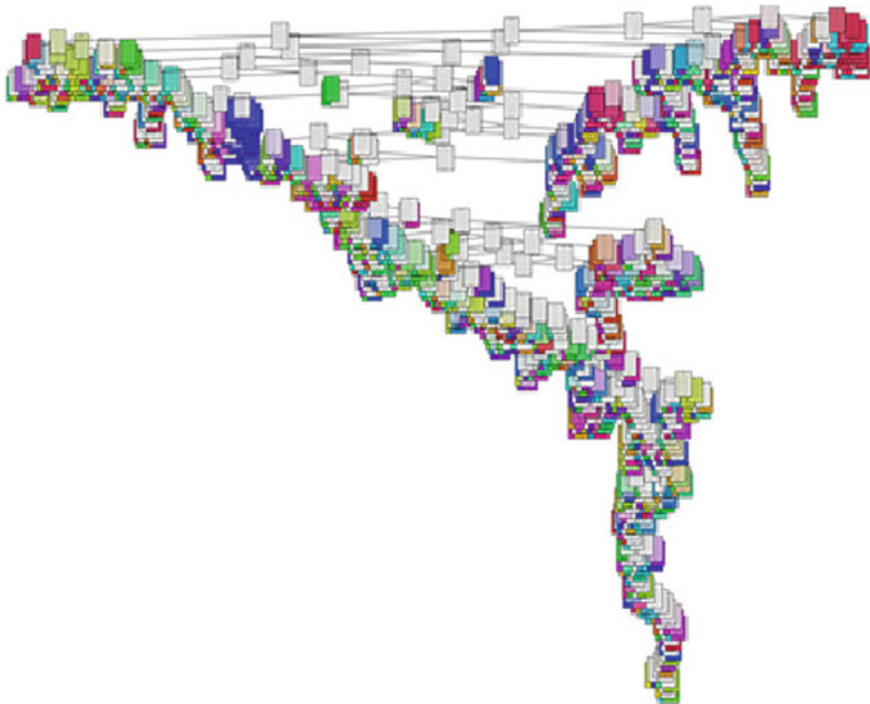


Fig. 4 Visualisation of decision tree for 16S rRNA sequence

parameters. The tree can be reproduced on a separate page or exhibited on a computer monitor without the need for navigation or scaling.

Random forest is a group-based machine learning approach. It is possibly the most popular and widely used machine learning algorithm due to its outstanding or excellent performance across a wide range of classification and regression predictive

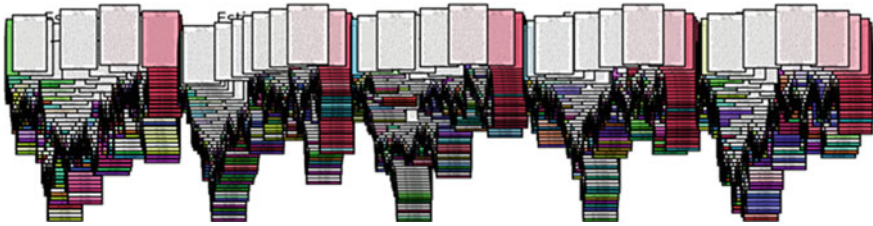


Fig. 5 Visualisation of random forest for 16S rRNA sequence (five trees)

modelling applications. It is also straightforward to use, with only a few important hyperparameters to tune and reasonable heuristics to do so. Overfitting is a decision tree problem in which the model adheres too closely to the features and traits of the train dataset and performs poorly on a new dataset—the test data. According to Schonlau and Zou [30], overfitting decision trees leads to low generalisability or the capacity to predict things in general. To enhance generalisation accuracy, only consider a subset of the observations and create multiple distinct trees. The random forest model is an aggregate tree-based learning system, which implies that it averages predictions from multiple individual trees when solving regression problems and utilises majority voting for solving classification problems. Rather than using the original sample, the separate trees are based on bootstrap samples. Bootstrap aggregating, sometimes known as bagging, is a technique for reducing overfitting.

When compared to decision trees, the random forest algorithm calculates the error rate more precisely. Figure 3 visualises the rules generated by decision tree for 16S rRNA gene sequences. Figures 4 and 5 visualise the trees generated by decision tree and random forest, respectively. From Fig. 5, it is clearly visible that random forest is a combination of multiple trees.

5 Experimental Results

The proposed framework is implemented using AMD Ryzen 5 4600H with Radeon Graphics 3.00GHz, RAM 8 GB unit.

The dataset consists of 12,507 sequences, and the maximum sequence length is 5897 after performing multiple sequence alignments. After calculating the conservation score for all the columns, the columns with “-” as 100% conservation score and the columns with the conservation score of less than 50% are eliminated. This resulted in 1447 columns after the elimination of unwanted data. All the results presented in this section are based on the mean accuracy over ten folds of cross-validation. The comparison predicting genes from the conserved region of the RNA sequences was only made on the 16S rRNA dataset of bacterial microbes. An overview of all results, in which different encoding methods are compared with the proposed CB-TB algorithms.

Table 6 Tenfold cross-validation accuracy results achieved using decision tree with label encoder (LE), one-hot encoder (OHE) and CatBoost encoder (CBE)

No. of folds	DT-LE	DT-OHE	DT-CBE
1	0.944	0.9440	0.950
2	0.957	0.9576	0.946
3	0.945	0.9456	0.937
4	0.948	0.948	0.943
5	0.949	0.949	0.947
6	0.952	0.952	0.947
7	0.944	0.944	0.957
8	0.936	0.936	0.939
9	0.955	0.955	0.952
10	0.956	0.956	0.943

Table 7 Tenfold cross-validation accuracy results achieved using random forest with label encoder (LE), one-hot encoder (OHE), and CatBoost encoder (CBE)

No. of folds	RF-LE	RF-OHE	RF-CBE
1	0.99	0.99	0.991
2	0.98	0.98	0.983
3	0.98	0.98	0.981
4	0.98	0.98	0.984
5	0.98	0.98	0.982
6	0.99	0.99	0.990
7	0.98	0.98	0.987
8	0.98	0.98	0.988
9	0.99	0.99	0.990
10	0.98	0.98	0.985

The tenfold cross-validation technique is a procedure for measuring the model's performance on new data, and it is used in conjunction with commonly used performance metrics such as accuracy and R-squared value. Also, the proposed framework, CatBoost encoding technique with tree-based algorithm, is compared with other mostly used encoding techniques such as one-hot encoding and label encoding techniques. The outcomes of this comparison are depicted in Tables 6 and 7 .

Tables 6 and 7 illustrate the tenfold cross-validation accuracy results achieved with different encoding techniques such as one-hot encoding, label encoding, and CatBoost encoding technique along with decision tree and random forest model, respectively. Table 8 shows the mean accuracy results of these tree-based models with different encoding techniques. Since the RNA data has more categorical features, they may have a very large number of levels, known as high cardinality, where most of the levels appear in relatively small number of instances.

Table 8 Mean accuracy values of tree-based models using different encoding methods

Encoder	Decision tree	Random forest
Label encoding	0.95	0.99
One-hot encoding	0.95	0.99
CatBoost encoding	0.95	0.99

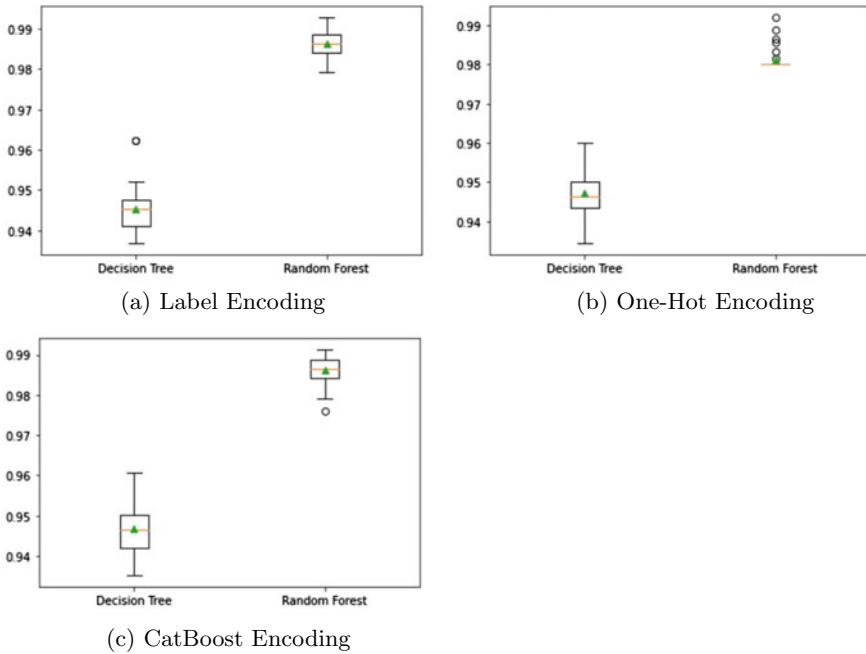


Fig. 6 Boxplot showing the performance of tree-based models on different types of encoding methods

After the elimination of unwanted data based on conservation score, the 16S rRNA dataset contains 12,507 sequences with 1447 features. Each feature consists of different levels of nucleotide bases. Typically, these features are maintained as text values that indicate distinct characteristics of the observations.

Each category is assigned a value from 1 to N in label encoding, where N is the number of categories for the feature. Each category is mapped to a vector that comprises 1 and 0, representing the presence and absence of the characteristic, in the one-hot encoding approach. The number of vectors is determined by the number of feature categories. If the number of categories for the feature is really large, this approach produces many columns, which considerably slows down learning. The results prove that applying these encoding techniques to this data does not affect the accuracy of the tree-based model. From Table 8, it is clear that CatBoost encoding techniques also provide the same accuracy as that of other methods.

Table 9 R-squared values of tree-based models using different encoding methods

Encoder	Decision tree	Random forest
Label encoding	0.90	0.98
One-hot encoding	0.90	0.98
CatBoost encoding	0.98	0.99

Table 10 Time taken to execute the tree-based models (in seconds) using different encoding methods

Encoder	Decision tree	Random forest
Label encoding	86.803	134.143
One-hot encoding	296.958	120.300
CatBoost encoding	80.837	69.4389

So, in order to investigate the performance of CatBoost encoded tree-based model (CB-TB), R-squared value of the model is calculated. R-squared helps to identify the percentage of variance explained by the model. Table 9 shows that overall CatBoost encoding when combined with either decision tree or random forest obtains highest R-square of all than the other encoding techniques. For 16S rRNA dataset, the other methods also perform well, but the percentage of variance of the independent variables with respect to dependent variable is highly explained by CatBoost encoding technique. From Fig. 6, the boxplot illustrates that the random forest model outperforms in all the encoding methods compared with decision trees.

When the CatBoost encoded model was executed over the entire dataset, the execution speed was very much slower with a mean execution time of 42.837 s for decision trees and 69.4389 s for random forest than the other two models as depicted in Table 10. This model outperforms the others based on the execution time also as analysing the DNA/RNA sequence is a time-consuming process due its size and complexity, which usually takes more than an hour to analyse without encoding techniques.

From Fig. 7, it is clear that CatBoost encoded random forest models outperform other models in terms of mean accuracy with 99%, R-squared value with 0.99 and with the lease execution time of 69.438 s compared to CatBoost encoded decision trees.

5.1 Statistical Analysis Using 5 × 2 Cross-Fold Approach

Choosing the correct machine learning algorithm is critical because it determines the model's performance. The most important factor in selecting a model is its performance, which is achieved using the K-fold cross-validation technique. When two

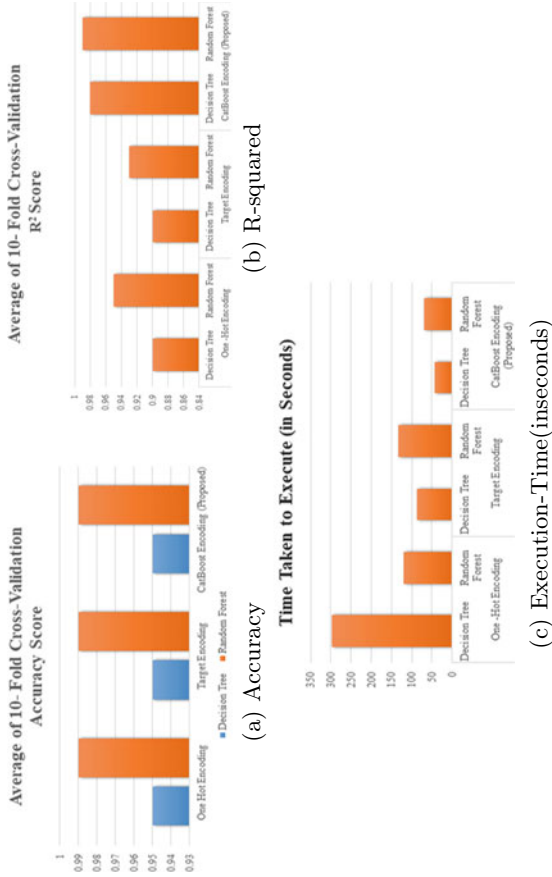


Fig. 7 Different performance metrics of tree-based models

data samples have the same distribution, a statistical hypothesis test determines how likely it is to witness them. The null hypothesis is described in this way. Statistical calculations can be used to examine this null hypothesis.

- If the test result indicates that there is insufficient evidence to reject the null hypothesis, any observed variation in model scores is purely coincidental.
- Any observed difference in model scores is real if the test result infers enough evidence to reject the null hypothesis.

Each model is evaluated using the same k-fold cross-validation split of the data, and each split score is calculated. For tenfold cross-validation, this would yield a sample of ten scores. The paired statistical test can then be used to compare the results. The formulation of the null hypothesis would be the first step.

H0: On the dataset, both models perform equally well.

H1: On the dataset, neither model performs equally well.

Acceptable level of significance: 0.5.

In order to compare the two models with the highest accuracy score, CatBoost encoded decision tree model and CatBoost encoded random forest model are selected to conduct 5×2 cross-fold statistical test to test the hypothesis. With the resultant p-value of 0.000 which is less than 0.5 and t-statistic of -14.033 , it is concluded to reject the null hypothesis that both models perform equally well on the dataset. Therefore, the two models are significantly different. This strongly proves the selection of CatBoost encoded random forest approach than CatBoost encoded decision tree for the prediction of genes from 16S rRNA dataset.

6 Conclusion

With label encoding, one-hot encoding, and CatBoost encoding, this paper examined two tree-based approaches, namely decision tree and random forest. With its R-squared value and execution time, it is discovered that CatBoost encoded tree-based model surpasses the other models. The 16S rRNA dataset is used to calculate mean accuracy, R-squared value, and the time it takes to run the model in a stand-alone system using tenfold cross-validation. The fact that the mean accuracy is the same for all encoding strategies shows that tree-based models outperform all other encoding techniques with gene sequence datasets. However, based on the R-squared value, CatBoost encoded tree-based methods clearly demonstrate the strong association between the model and the response variable. In decision forest, the suggested model explains 98% of the variability of the response data around its mean, while in random forest, the proposed model explains 99% of the variability of the response data around its mean. Another time-consuming operation with DNA/RNA sequence datasets is executing the model due to its vast size, although the CatBoost encoded tree-based model significantly reduces the execution time when compared to conventional encoding techniques. As a result, it is inferred that the CatBoost encoded random

forest technique can be successfully used to the gene sequence dataset to accurately identify bacteria at the gene and species level in less time.

References

1. Dauer J, Dauer J (2016) A framework for understanding the characteristics of complexity in biology. *Int J STEM Educ* 3(1):1–8. <https://doi.org/10.1186/s40594-016-0047-y>
2. Stephens ZD, Lee SY, Faghri F, Campbell RH, Zhai C, Efron MJ, Robinson GE (2015) Big data: astronomical or genomics? *PLoS Biol* 13(7):e1002195. <https://doi.org/10.1371/journal.pbio.1002195>
3. Shadab S, Khan MTA, Neezi NA, Adilina S, Shatabda S (2020) DeepDBP: deep neural networks for identification of DNA-binding proteins. *Inform Med Unlocked* 19:100318
4. Blanco G, Blanco A (2017) *Medical biochemistry*. Academic Press
5. Fadeev E, Cardozo-Mino MG, Rapp JZ, Bienhold C, Salter I, Salman-Carvalho V, Boetius A (2021) Comparison of two 16S rRNA primers (V3-V4 and V4-V5) for studies of arctic microbial
6. Putri HY, Agustien A, Alamsjah F (2021) Screening, characterization and 16S rRNA sequencing of thermophilic bacteria producing amylase and protease from Pekonina hot springs, South Solok. *Int J Progr Sci Technol* 28(2):154–161
7. Mazzarelli A, Giancola ML, Farina A, Marchioni L, Rueca M, Gruber CEM, INMI COVID-19 study group (2021) 16S rRNA gene sequencing of rectal swab in patients affected by COVID-19. *PLoS One* 16(2):e0247041
8. Bukhari SNH, Jain A, Haq E, Khder MA, Neware R, Bhola J, Lari Najafi M (2021) Machine learning-based ensemble model for zika virus T-cell epitope prediction. *J Healthcare Eng*
9. Zhou Y, Zhang L, Xu J, Zhang J, Yan X (2021) Category encoding method to select feature genes for the classification of bulk and single-cell RNA-seq data. *Stat Med*
10. Gunasekaran H, Ramalakshmi K, Rex Macedo Arokia Raj A, Deepa Kanmani S, Venkatesan C, Suresh Gnana Dhas C (2021) Analysis of DNA sequence classification using CNN and hybrid models. *Comput Math Methods Med*
11. Chatzou M, Magis C, Chang JM, Kemena C, Bussotti G, Erb I, Notredame C (2016) Multiple sequence alignment modeling: methods and applications. *Brief Bioinform* 17(6):1009–1023
12. Thompson JD, Higgins DG, Gibson TJ (1994) CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucl Acids Res* 22(22):4673–4680. <https://doi.org/10.1093/nar/22.22.4673>
13. Sievers F, Wilm A, Dineen D, Gibson TJ, Karplus K, Li W, Higgins DG (2011) Fast, scalable generation of high-quality protein multiple sequence alignments using Clustal Omega. *Mol Syst Biol* 7(1):539
14. Edgar RC (2004) MUSCLE: a multiple sequence alignment method with reduced time and space complexity. *BMC Bioinform* 5(1):1–19
15. Di Tommaso P, Moretti S, Xenarios I, Orobitg M, Montanyola A, Chang JM, Taly JF, Notredame C (2011) T-Coffee: a web server for the multiple sequence alignment of protein and RNA sequences using structural information and homology extension. *Nucl Acids Res* 39(SUPPL. 2):13–17. <https://doi.org/10.1093/nar/gkr245>
16. Katoh K, Standley DM (2013) MAFFT multiple sequence alignment software version 7: improvements in performance and usability. *Mol Biol Evol* 30(4):772–780. <https://doi.org/10.1093/molbev/mst010>
17. Do CB, Mahabhashyam MS, Brudno M, Batzoglou S (2005) ProbCons: probabilistic consistency-based multiple sequence alignment. *Genome Res* 15(2):330–340
18. Valdar WS (2002) Scoring residue conservation. *Proteins Struct Func Bioinform* 48(2):227–241

19. Wu TT, Kabat EA (1970) An analysis of the sequences of the variable regions of Bence Jones proteins and myeloma light chains and their implications for antibody complementarity. *J Exp Med* 132(2):211–250
20. Jores R, Alzari PM, Meo T (1990) Resolution of hypervariable regions in T-cell receptor chains by a modified Wu-Kabat index of amino acid diversity. *Proc National Acad Sci USA* 87(23):9138–9142. <https://doi.org/10.1073/pnas.87.23.9138>
21. Yu N, Li Z, Yu Z (2018) Survey on encoding schemes for genomic data representation and feature learning—from signal processing to machine learning. *Big Data Mining Anal* 1(3):191–210
22. Nguyen NG, Tran VA, Phan D, Lumbanraja FR, Faisal MR, Abapihi B, Satou K (2016) DNA sequence classification by convolutional neural network. *J Biomed Sci Eng* 9(5):280–286
23. Hancock JT, Khoshgoftaar TM (2020) Survey on categorical data for neural networks. *J Big Data* 7(1):1–41
24. Cerda P, Varoquaux G, Kégl B (2018) Similarity encoding for learning with dirty categorical variables. *Mach Learn* 107(8–10):1477–1494. <https://doi.org/10.1007/s10994-018-5724-2>
25. Larionov M (2020) Sampling techniques in bayesian target encoding, pp 1–12. <http://arxiv.org/abs/2006.01317>
26. Micci-Barreca D (2001) A preprocessing scheme for high-cardinality categorical attributes in classification and prediction problems. *ACM SIGKDD Explor Newslett* 3(1):27–32. <https://doi.org/10.1145/507533.507538>
27. Hussain S, Mustafa MW, Jumani TA, Baloch SK, Alotaibi H, Khan I, Khan A (2021) A novel feature engineered-CatBoost-based supervised machine learning framework for electricity theft detection. *Energy Rep* 7:4425–4436. <https://doi.org/10.1016/j.egy.2021.07.008>
28. Prokhorenkova L, Gusev G, Vorobev A, Dorogush AV, Gulin A (2018) Catboost: unbiased boosting with categorical features. *Adv Neural Inf Process Syst (Section 4)*:6638–6648
29. Stiglic G, Kocbek S, Pernek I, Kokol P (2012) Comprehensive decision tree models in bioinformatics. *PLoS ONE* 7(3). <https://doi.org/10.1371/journal.pone.0033812>
30. Schonlau M, Zou RY (2020) The random forest algorithm for statistical learning. *Stata J* 20(1):3–29. <https://doi.org/10.1177/1536867X20909688>

Roadkill Avoidance System Using YOLOv5



Mrunal Mendgudle and Mrunal Shidore

Abstract Roadkills due to collisions have been on the rise lately, and many endangered animal species are on the affected radar. Traditional animal-vehicle collision avoidance systems use object detection for detecting animals, but often these systems fail due to innumerable reasons, one of them being lack of training on appropriate data. In this paper, we identify and overcome the failure spot of most animal detection systems to present a fast, efficient, and reliable animal-vehicle collision avoidance framework. It is based on a custom collected dataset containing preprocessed, augmented, and annotated images belonging to 20 categories of animals most frequently encountered on roads. Further, the model is trained over the YOLOv5 deep learning architecture that gives a detection processing speed of 140 fps and renders an exceptional mAP of 91.27%.

Keywords Roadkill · Road ecology · Augmentation · YOLOv5 · CNN · mAP · IoU

1 Introduction

An extremely large numbers of animals, majorly mammals, birds, and reptiles are killed on roads every day across the globe. The mortality resulting from road accidents due to crashes or collisions can be very significant for animal species that are already threatened or endangered with small populations. A million animals are estimated to have been road-killed every day, alone in the USA [1]. Around 10 million animals are hit on Australian roads every year [2]. In 2019, more than 80 leopards were killed in India solely due to road and train crashes [3]. These numbers are overwhelming when picked up from densely populated countries having a decently developed automobile industry.

M. Mendgudle (✉) · M. Shidore
Vishwakarma Institute of Technology, Pune, India
e-mail: mrunal.mendgudle18@vit.edu

M. Shidore
e-mail: mrunal.shidore@vit.edu

Computer vision-based techniques show enormous potential to solve this problem through object detection to detect animals on roads to alert drivers. In this paper, we present a robust, reliable, and efficient system to automatically detect animals while they are crossing roads and along roadside, using the state-of-the-art you only look once (YOLO) deep learning architecture which renders an exceptional detection processing speed of 140 frames per second.

This work is organized as follows: Section 1 gave an introduction to the problem statement. Section 2 outlines related works based on road ecology surveys estimating roadkills and on previous work done in the field of deep learning for animal-vehicle collision avoidance. Section 3 describes the proposed work—the custom dataset and the methodology implemented. The experimentation results and evaluation metrics' scores are discussed in Sect. 4. Section 5 concludes and summarizes the proposed work, followed by the future work in Sect. 6.

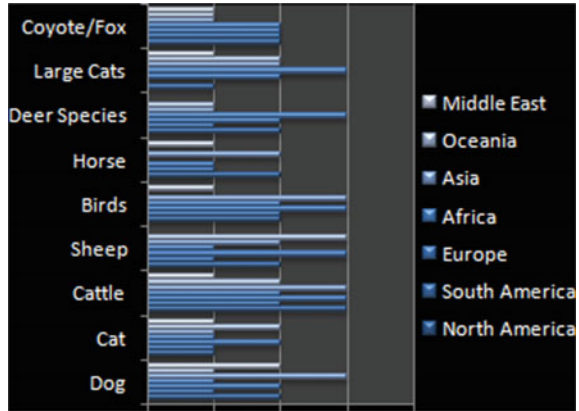
2 Literature Survey

Over the past couple of years, object detection has paced exponentially as its applications cater to a palette of industry requirements and research objectives. In the current age of autonomous and self-driving vehicles, the task of detecting animals accurately becomes an important one. Thus, an analysis was done to identify the animals that are most prone to collisions, so as to connect the dots between road ecology and real-time animal detection on road.

It was found that in Northern America, the animals that were most vulnerable to roadkills were deer, dogs, squirrels, raccoons, and rabbits [4]. In European countries, badgers, dogs, and birds were found to be most susceptible to roadkills. Around 194 million birds and 29 million mammals are estimated to have been killed each year on European roads, according to a study conducted by University of Reading [5]. While in Australia alone, Kangaroos are involved in eight out of total 10 car crashes with animals [6]. In South America, hare, rabbits, and foxes captured higher shares in the amount of roadkills that happen each year, apart from dogs and cattle [7]. In Asia, dogs, cattle, donkeys, horses, and monkeys cross roads more often than other animals. Leopards and large wild cats remain the ones badly hit by vehicle collisions given to their already dwindling population [8]. Middle Eastern countries encounter camels and horses moving by the roadside, while the African subcontinent sees more diverse mammals crossing roads, such as deer species, elephants, giraffes, zebras, wild buffalo, and wild cats [9]. A combined worldwide distribution of roadkills is shown in Fig. 1.

Conventional animal collision avoidance systems use outworn classifiers like HOG and HAAR cascade classifiers which have shown poor performance in terms of accuracy, precision, recall, and speed as compared to convolutional neural networks [10]. Further the evolution of CNNs led systems to use architectures like region-based CNN (R-CNN) and faster R-CNN [11] which are region-based detection techniques, but their slow detection processing speed holds them back from being practically

Fig. 1 Distribution of roadkills worldwide



deployed in real-time. Sharma et al. in [10] propose a system to alert drivers about approaching animals only up to a speed of 35 km/hr, exceeding which animals are not detected, meaning that the speed of vehicle and animal would decide whether an animal is detected, and if driver is alerted. Autonomous driving heavily depends on object detection [12] but often different movements, angles, and postures of animals confuse the systems due to lack of training on ‘on-road’ image dataset, resulting in animal injuries and fatalities. For instance, Volvo automobile company admitted that hopping Kangaroos puzzled its cars [13]. Many other traditional animal detection models are fairly geography centered, detecting only a handful local animals, for example, detecting only cows and dogs on Indian roads [10]; thus, making it difficult to be adapted as a generalized system throughout the globe.

Ever since its inception, you only look once (YOLO) architecture has been used extensively to train custom datasets for detection purposes due to its high accuracy and speed. Buric et al. in [14] describe detecting sports players with handball using custom dataset to compare YOLO in terms of accuracy and speed with Mask R-CNN. Cao et al. detect shuttlecock for badminton robots, using YOLO, overcoming the issue of fast-moving shuttlecocks in complex contexts [15]. Further, various other applications such as crop harvesting systems [16] and detecting fruits like apples in different stages of growth [17] have used YOLO architecture for extracting higher accuracy. More recently, YOLO has widely been used to detect face masks in the fight against COVID-19 [18].

A key element missing from previously proposed animal-vehicle collision avoidance systems is a practical harmony between road ecology estimations and findings, and a solution-based deep learning approach.

3 Proposed Work

In this proposed system, we present a framework for animal-vehicle collision avoidance using a custom dataset comprising images categorized in 21 classes, and a custom trained YOLOv5 model duly taking care of the trade-off between the mAP of the model and its real-time performance on live data. The proposed YOLOv5 model is based on analyzed and surveyed data and has been designed taking into consideration that different geographies invite different animals on roads, leading to different expectations for collision avoidance systems worldwide. We use the generalization property of YOLO to customize the architecture for the purpose of animals-on-roads detection, to train on the custom dataset.

The system block diagram is shown in Fig. 2. First stage is image data acquisition, after which the image dataset is subjected to preprocessing and augmentations. To describe classes and bounding boxes in an image, data annotation is performed. This annotated data is finally fed to the single stage object detector YOLOv5 which has three parts—backbone, neck, and head, i.e., the final detector.

3.1 Dataset

This section describes the work done for building the custom dataset through data acquisition, augmentation, and annotation as illustrated in block diagram (Fig. 2).

Data Acquisition: Most of the animals encountered on roads as described in section I are not found in open-source public datasets like CIFAR-10 and MS-COCO, thus a need for a new custom dataset arises. The dataset used to train the model here consists of a homogeneous mixture of custom captured pictures and Internet sourced public images. It contains 2500 images of animals on road and by roadside, consisting of preprocessed and augmented images picturing animals hopping, running, jumping, striding, crossing, moving etc., in different illumination intensities, captured when approached by vehicles of varying speeds, finally resized to 416×416 pixels to suit our YOLOv5 architecture.

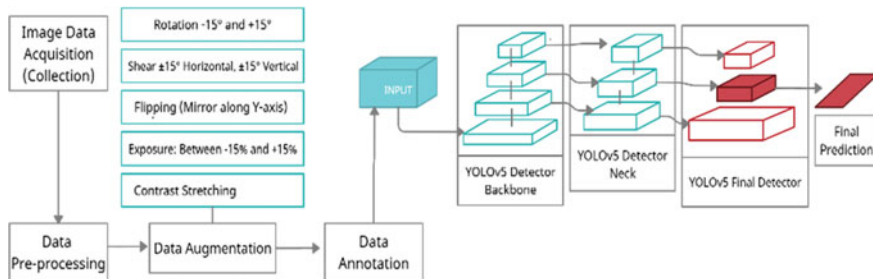


Fig. 2 Block diagram of the proposed system

Table 1 Clubbing and categorization of animals

Animal Species	Category
Coyote, Fox, Jackal, Wolf	Fox
Antelopes, Impalas, Kudus, Bushbucks, Puku, Fallow Deer, Spotted Horned Deer	Deer species
Cows, Calves, Oxen, Bulls	Cattle
Tigers, Leopards, Cougars, Jaguars, Cheetahs, Lionesses, Panthers	Large cats
Brown Bear, Sloth Bear, Black Bear	Bear
Baboons, Macaque, Orangutan, Gibbon, Monkeys	Monkeys
All types of bird species	Birds

The custom part of the image dataset was captured on local Indian roads as well as on the highways passing through forests and urban neighborhoods of various African countries. The Internet sourced public images were collected and verified to suit the on-road condition, processed to avoid over-fitting and to prepare the model for detection in all kinds of geographies, backgrounds and lighting conditions. The background of cattle crossing a road in Asia can look different from cattle crossing a road in the USA, thus data acquisition was done understanding the similarities and differences, and acknowledging the diversity through an equal geographical representation.

The dataset is divided into 21 classes—dog, cat, horse, sheep, pig, cattle, donkey, camel, elephant, giraffe, zebra, bear, kangaroo, wild buffalo, deer species, fox, monkey, birds, large wild cats, lions, and animal-crossing signboard. There are 20 animal-type classes, and one class is denoted to ‘animal-crossing signboard’. It is important to identify animal-crossing signboard, which can indicate the likelihood of animals crossing the road at that place to warn the driver. To avoid the curse of dimensionality, an effort was made to logically club animals belonging to a particular sub-species in one single category as depicted in Table 1.

Data Augmentation. To enhance the dataset, the acquired images were preprocessed and augmented. Often it is considered tough to implement augmentation strategies for object detection than for simple classification, as one must handle the intricacy of bounding boxes [19]. Examples of image augmentations are depicted in Figs. 3 and 4. On an average, three augmentations per image were generated; thus, the size of the dataset changed to three times the originally acquired dataset. The following augmentations were performed on the data:

- Rotation: Between -15° and $+15^\circ$ —this rotates the image 15° clockwise and 15° anti-clockwise.
- Shear: $\pm 15^\circ$ Horizontal, $\pm 15^\circ$ Vertical—Shearing changes the shape and size of the image along x and y axis.
- Exposure: Between -15% and $+15\%$ —Increases the overall illumination, i.e., brightness or darkness of the image.
- Flipping: Horizontal mirror reflections.



Fig. 3 Image augmentation—rotation



Fig. 4 Image augmentation—exposure

- **Contrast Stretching:** Stretching the range of values of intensity in the image, to increase the dynamic range of the image.

Data Annotation: Data annotation is a fundamental part of any deep learning project. The aim of data annotation is to set boundaries by defining bounding boxes in an image, within which the probability of the object being present is 100%. We define the bounding box values along with the class label. YOLOv5 in PyTorch reads the bounding box data in text file format (.txt), meaning all the object bounding boxes in one image would be combined and exported in one text file corresponding to the image.

All the images were manually labeled by the authors, according to the previously mentioned animal categories. Since a lot of images picture animals crossing roads in herds or collections, they were carefully annotated to ensure the bounding boxes do not lose the common dominant features of animal categories, which are used to build feature pyramids in YOLOv5.

3.2 Methodology

YOLO is a revolutionary architecture that has made real-time object detection fast and accurate. YOLOv5—the latest version, is written in PyTorch framework, which makes its inferences faster compared to older versions. Its performance is higher than all older versions of YOLO both in terms of accuracy and speed. Shortcomings of most of the previously proposed deep learning architectures can be overcome by YOLOv5 that brings with itself an advantageous detection processing speed of 140 frames per second.

YOLO architecture sees the entire image during training and test time, so it implicitly encodes contextual information about classes as well as their appearance [20]. The YOLOv5 architecture has three prominent parts. The backbone of the architecture is instrumental in detecting rich features from an input image. Further, the neck part of the architecture builds feature pyramids which in turn help in detecting same objects (animals) in different sizes and shapes. The head part is the final detector which uses anchor boxes techniques. Anchor box structure was introduced in YOLOv2, which tends to reduce training time and increases the accuracy of the network. However, a drawback of anchor boxes is that they cannot quickly adapt to unique and diverse dataset as in our case. To solve this, YOLOv5 integrates anchor box selection process by automatically learning the best anchor boxes for a dataset to use them during training over the dataset.

4 Results and Discussion

From the experimental results, one can observe that the YOLOv5 animals-on-road detection model can accurately identify animals in different geographies with complex backgrounds, different orientations and varying lighting conditions.

Figure 5a–h displays the test results for animals detected on roads. In Fig. 5a, we observe that long-haired highland cattle species has been detected accurately as cattle. In Fig. 5b, three dogs have been detected in complex and crowded background, and in Fig. 5d, all the birds have been detected properly while they are pictured crossing the road, irrespective of their sizes. A horse carriage is pictured from behind in Fig. 5e, and horse has been detected accurately. In Fig. 5g, we can see that a herd of elephants is crossing the road, though elephants overlap each other in the picture, they have all been detected accurately by our model.

To mathematically evaluate the model's performance, and to compare it with previously published works, standard evaluation metrics were used—precision, recall, and mAP. Precision is defined as the fraction of results predicted correctly among all the predicted results. However recall is the ratio of correct positive results to the ground truth positive results. From Fig. 6, it is observed that precision and recall of the proposed model increase with time. Thus, it is deduced that the model is performing well. Our proposed model has a recall of 91.46%, precision of 85.62% and reaches

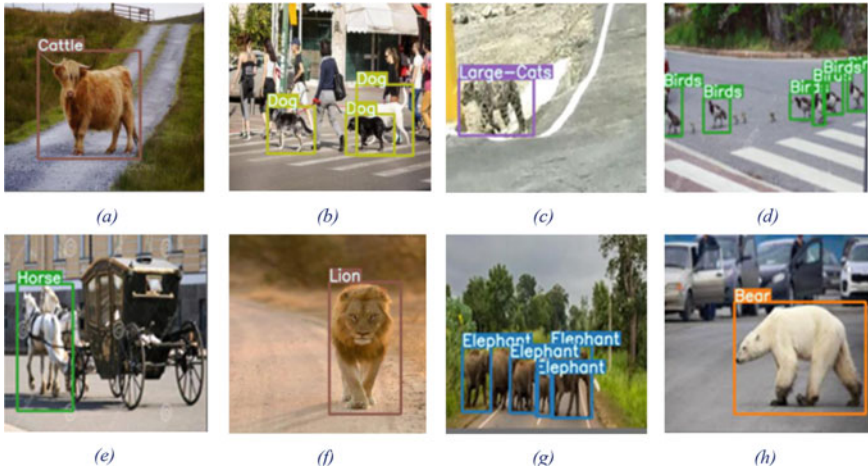


Fig. 5 Some detections in complex backgrounds

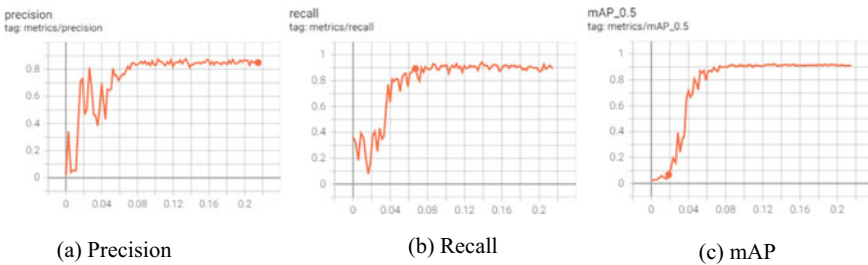


Fig. 6 Evaluation metrics

an accuracy of 90.20%. To correctly evaluate the performance of object detection models, mean average precision (mAP) score is used. The mAP compares the ground truth bounding box to the bounding box detected by the trained model and returns a score. In (1), AP^k is average precision for class k , and n is number of classes.

$$mAP = \frac{1}{n} \sum_{k=1}^n AP^k \tag{1}$$

$$IoU = \frac{\text{Area of overlap of predicted \& actual bounding box}}{\text{Area of union of predicted \& actual bounding box}} \tag{2}$$

Intersection over union (IoU) helps in determining the tightness of the bounding box. It is a ratio of intersection and union of ground truth and predicted bounding box. In our case, we consider mAP with IoU threshold > 0.5 . Our model yields mAP score of 91.27%, which depicts a significant improvement compared to previous

Table 2 Comparative analysis of performance

Architecture	No. of objects	Recall	Accuracy (%)	mAP
YOLOv1 [21]	6	–	98.8	–
HOG, HAAR [10]	1	80.4%	82.5	–
SSD [11]	25	86.45	–	80.5% (100 fps)
R-CNN [11]	25	88.34	–	82.11% (10 fps)
YOLOv5 [Proposed]	21	91.46%	90.20	91.27% (140 fps)

works. Table 2 depicts a comparative analysis of the proposed work with some of the established previously published works.

As observed from section II, we understand that this is the first time that YOLOv5 has been implemented for the purpose of animal-vehicle collision detection, which is robust, accurate, and fast as its speed is compatible with human driven as well as self-driving cars. Additionally, we understand that neither of the previously proposed systems uses dataset containing images of animals specifically on roads or by roadside, to train the system on, which defines the very base of our system. Most of the previous systems have trained their models on open-source images from domestic farms, forests, or camera traps, which hardly meet the criterion for ‘right dataset’. The behavior, activity, orientation, movement, and posture of animals matter and are absolutely different on and along roads, as compared to their natural habitat. Thus, correct images from the viewpoint of drivers across the windshield of the vehicle can help train the model efficiently to detect animals when driving on roads, and when trained over YOLOv5 architecture the combination proves to be accurate and fast.

5 Conclusion

We have introduced an improved, robust real-time animal collision avoidance framework which is based on relevant road ecology surveys, with a highly relevant dataset trained on YOLOv5. We also introduced a fully annotated dataset, consisting of 2500 ‘on-road’ and ‘along roadside’ images of over 21 classes—20 species of animals and one common ‘animal-crossing signboard’ class with an aim to make it publicly available for further work. These are real-world images and include natural, complex, indistinguishable backgrounds, varying lighting conditions, differing animal movements, postures, activities, shapes, numbers, etc.

Despite the challenging data setup, we can conclude that our YOLOv5 model is safe and reliable to use in all geographies for most common animals encountered on roads and seen along roadside. Finally, our model yields outstanding mAP score of 91.27%, which is a significant improvement compared to previous work in this field.

6 Future work

For future work, the number of animals crossing roads can be counted to indicate whether it is a herd of animals or lone ones moving by, which can further help in alerting drivers. To ensure that no other researcher faces the limitation in image acquisition, an open-source data using camera traps can be set up. These camera traps can be set up with assistance from local authorities in different regions. To use our model on working systems, for future, it is necessary to collect more real-world dataset consisting of on-road images, which can be collected by setting camera traps near animal-crossing signboards, traffic signals, highway deviations, etc.

References

1. Bekoff M (2010) Animals and cars: one million animals are killed on our roads every day. *Psychology today*. Accessed 16 Apr 2021 <https://www.psychologytoday.com/us/blog/animal-emotions/201007/animals-and-cars-one-million-animals-are-killed-our-roads-every-day>
2. Parrott M (2021) 10 million animals are hit on our roads each year here's how you can help them (and steer clear of them) these holidays. *The conversation*, Accessed 21 Mar 2021 <https://theconversation.com/10-million-animals-are-hit-on-our-roads-each-year-heres-how-you-can-help-them-and-steer-clear-of-them-these-holidays-149733>
3. Chatterjee B (2019) India saw 83 leopards killed in road and train mishaps in 2019, highest in 10 yrs. *Hindustan Times* Dec 31, 2019, Accessed 2021/04/16. <https://www.hindustantimes.com/mumbai-news/india-saw-83-leopards-killed-in-road-and-train-mishaps-in-2019-highest-in-10-yrs/story-7LJ6UuXTj71cf5XvpTsBiM.html>
4. Wildlife on the road, [Online], Accessed 18 Apr 2021, Available <https://www.havahart.com/wildlife-on-the-road>
5. University of Reading (2020) Roadkill study identifies animals most at risk in Europe: Most frequently killed species on European roads identified, as well as most vulnerable. *ScienceDaily* [Online], Accessed 24 Mar 2021 www.sciencedaily.com/releases/2020/06/200610135101.htm
6. Kloecker U, Croft D, Ramp D (2006) Frequency and causes of kangaroo-vehicle collisions on an Australian outback highway. *CSIRO Wildlife Research*. 33:5–15. <https://doi.org/10.1071/WR04066>
7. Abra FD, Huijser MP et al (2021) An estimate of wild mammal roadkill in São Paulo state Brazil. *Heliyon* 7(1):e06015
8. Gubbi S (2014) Roads emerging as a critical threat to leopards in India? *CAT News* 60:30–31
9. Gandiwa E, Mashapa C, Muboko N, Chemura A, Kuvaoga P, Mabika CT (2020) Wildlife-vehicle collisions in Hurungwe Safari Area, northern Zimbabwe, *Sci African* 9:e00518, ISSN 2468–2276
10. Sharma SU, Shah DJ (2017) A practical animal detection and collision avoidance system using computer vision technique. *IEEE Access* 5:347–358. <https://doi.org/10.1109/ACCESS.2016.2642981>
11. Saxena A, Gupta DK, Singh S (2021) An animal detection and collision avoidance system using deep learning. In: Hura G, Singh A, Siong Hoe L (eds) *Advances in communication and computational technology. Lecture notes in electrical engineering*, vol 668. Springer, Singapore
12. Sarda A, Dixit S, Bhan A (2021) Object Detection for Autonomous Driving using YOLO [You Only Look Once] algorithm. In: 2021 Third international conference on intelligent communication technologies and virtual mobile networks (ICICV), pp 1370-1374. <https://doi.org/10.1109/ICICV50876.2021.9388577>

13. Gray A (2021) Kangaroos are confusing self-driving cars, World Economic Forum. Accessed 31 Mar 2021 <https://www.weforum.org/agenda/2017/07/engineers-testing-volvo-s-driverless-technology-have-hit-a-problem-kangaroos/>
14. Buric M, Pobar M, Ivasic-Kos M (2018) Ball detection using yolo and mask R-CNN. In: 2018 International conference on computational science and computational intelligence (CSCI), pp 319–323
15. Cao Z, Liao T, Song W, Chen Z, Li C (2021) Detecting the shuttlecock for a badminton robot: a YOLO based approach. *Expert Syst Appl* 164:113833
16. Junos MH, Mohd Khairuddin AS, Thannirmalai S, Dahari M (2021) An optimized YOLO-based object detection model for crop harvesting system. *IET Image Processing*
17. Yunong T et al (2019) Apple detection during different growth stages in orchards using the improved YOLO-V3 model. *Comput Electron Agric* 157:417–426
18. Liu R, Ren Z (2021) Application of yolo on mask detection task. In: 2021 IEEE 13th international conference on computer research and development (ICCRD) pp 130–136. <https://doi.org/10.1109/ICCRD51685.2021.9386366>.
19. Zoph B, Cubuk ED et al. (2020) Learning data augmentation strategies for object detection. In: Vedaldi A, Bischof H, Brox T, Frahm JM (eds) *Computer vision—ECCV 2020*. ECCV 2020. Lecture notes in computer science, vol 12372. Springer, Cham. https://doi.org/10.1007/978-3-030-58583-9_34.
20. Joseph R et al. (2016) You only look once: unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition.
21. Sayagavi AV, Sudarshan TS et al. (2020) Deep learning methods for animal recognition and tracking to detect intrusions. In: *International conference on information and communication technology for intelligent systems*. Springer, Singapore

Automated Cluster Head Selection in Fog-VANET Via Machine Learning



Anshu Devi, Ramesh Kait, and Virender Ranga

Abstract Vehicle autonomous networks (VANETs) are widely acknowledged as one of the most important technologies for upgrading today's transportation networks. Because VANET is divided into two distinct groups, safety and non-safety applications, continuous data interchange is required. Data that is highly time sensitive should be exchanged as soon as possible. It necessitates the use of intelligent infrastructure capable of storing, computing, and communicating. FOG computing provides a storage, computation, and communication platform. This study investigates the integration of fog computing (FC) with Lagrange Polynomial Interpolation for the purpose of authentication of the node via machine learning. The Authentic Vehicle Node with FOG Computing (AVNFC) scheme, which is a fog-enabled VANET architecture, is described in this article.

Keywords Fog computing · VANET · Machine learning

1 Introduction

It is discernible that industry of automation has made significant strides. Integrating components of both hardware and software improves the drivability and delight of customers. Mobile automobiles with on-board units and roadside assistance units make up a vehicular ad hoc network [1]. It is now possible to communicate from one vehicle to another (V2V) to improve drivers' awareness of roadside accidents, traffic bottlenecks, and other hazards [5]. It makes driving in city traffic and on highways safer and more comfortable for the driver. Vehicle tracking and weather

A. Devi (✉) · R. Kait
Kurukshetra University, Kurukshetra, Haryana, India
e-mail: anshu.cse@kuk.ac.in

R. Kait
e-mail: rameshkait@kuk.ac.in

V. Ranga
Delhi Technological University, Delhi, New Delhi, India
e-mail: virenderranga@dtu.ac.in

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
H. Sharma et al. (eds.), *Communication and Intelligent Systems*, Lecture Notes in Networks and Systems 461, https://doi.org/10.1007/978-981-19-2130-8_89

1169

monitoring are among the VANET safety applications that must now be followed. In order to gather prior knowledge of junctions and highways, as well as knowledge of the situations of other vehicles, VANET makes use of safety apps.

Real-time warning signals can be broadcast and handled via VANET or RSU. The VANET should also be able to notify other vehicles. The TMO regularly generates DDoS and incursion alarms. These include CC (collision/congestion), link failure, and bad roads. An accident or a crippled vehicle can cause CC (collision/congestion) in VANET. DoS attacks are examined as sophisticated and typical attacks. Links can be sabotaged by DoS and SNI. Surges in network traffic cause the VANET V2V connection to be terminated. Attacks on VANETs, such as denial of service and distributed denial of service, constitute road hazards. These interfere with the proper operation of VANET safety applications.

Furthermore, they may pave the way for later VANET attacks by creating unfavourable road conditions or traffic congestion. As a result, drivers will have a harder time preventing road fatalities. SNI and DRA attacks, which are known to overwhelm the RSU in the event of an attack, are examples of disruptive attacks. DoS and SNI, as previously noted, can employ RSU computation and communication capabilities to overwhelm the system with data.

Conversely, real-time sensor units (RSUs) as well as the safety applications they deploy are built to gather and analyse live data from vehicles. The data obtained through vehicle-to-vehicle should be properly analysed and quickly transmitted to other nearby automobiles connected via VANET and safety apps via end-to-end (E2E) connection in order to be useful. It is critical in VANET that E2E (end-to-end) communication is protected from DoS and SNI attacks, which might cause the RSU to become overwhelmed and require quick attention.

It is possible that the RSU will lose time processing incorrect messages or information. As a result, for sending V2RSU and V2V messages in VANET, the RSU requires a storage solution that is both efficient and safe [9]. The VCF concept was built with the use of cloud-based logical interaction and fog computing technologies. Based on VCF, vehicles can share sensing, processing, and resource resources in order to improve traffic management and road safety on the highway. Routes are developed based on information obtained from vehicles in the surrounding area [4].

Organisations in charge of traffic management at the local and state levels analyse vast amounts of data. It is conceivable to rebuild significant events in the future, such as traffic jams or attacks, using virtual reality technology (including DoS and SNI attacks).

Secure VANET RSUs allow V2V and V2RSU communication in real time. Delay/jitter and throughput QoS aspects are crucial in VANET. Because of the VANET's dynamism, wireless communication is required. Secure wireless link deployment and related connectivity should be a top priority for the RSU. This study's main contributions are as follows: Fog computing is employed to assess the authenticity of the node by applying the Lagrange interpolation method for key evaluation. It also cuts down on communication time. Fog computing extends cloud computing by providing processing, storage, and network connectivity between end nodes. The communication pattern of automobiles has been used to determine the

amount of information that can be stored rapidly in a vehicle's storage. Ascertain the trustworthiness of VANET nodes and the network as a whole. Improvements to the delay, packet delivery ratio, and throughput of the VANET were made.

The remainder of the sections are organised as follows. The second section discusses related work. The proposed flowchart is discussed in detail in Sect. 3. Detailed simulation results and an analysis of the simulation findings are presented in Sect. 4. Section 5 concludes with future research plans.

2 Related Work

The following are the topics covered in this section of VANET-related work: An active and constant monitoring of vehicle location, route, speed, and alcohol consumption is required to reduce road accident risk. These parameters are extracted from real-time data utilising IoVs and high-performance fog computing. Machine learning algorithms classify vehicles as dangerous (accident-prone) or non-risky (non-accident-prone). Risky cars are now saved in real time by transmitting safety notifications. Various machine learning methods are utilised to find the best accident detection model. Generally, this work reduces road accidents [10].

The authors [16] propose FSDN, a novel VANET architecture that integrates two emerging computing and network paradigms, SDN and fog computing. Flexible, programmable, and global knowledge SDN-based architecture, while fog computing provides delay-sensitive and location-aware services that could meet future VANET situations. The authors identify all SDN-based VANET components and their roles in the system. The authors also discuss how resource managers and fog orchestration models might be used to support surveillance services. The suggested architecture could address the key VANET issues by combining V2V, V2I, and V2B communications with SDN centralised control, maximising resource efficiency and lowering latency. Aside from demonstrating the benefits of our proposed architecture, two non-safety use cases (data streaming and lane-change assistance) are described.

The authors employed a key sharing strategy to assure secure data transfers. It was found that adding an intermediary fog layer between the cloud and mobile device improved the usability of cloud computing overall [15].

Furthermore, the authors investigated the applicability of computing in the fog by utilising an data based on events collecting approach, which they developed themselves [8].

Every time a network node receives a data transfer request, a task is assigned to it and an event is triggered. When planning a route, consider ' n ' occurrences like connecting hops between places. When determining a route, you might incorporate extra hops between the starting point and the final destination. Additionally, adding a hop requires the selecting a network of dependable nodes, which is performed via optimization techniques (OAs) [14].

The authors [13] demonstrated the feasibility of a transferrable VANET geocast routing belief model, which was then put into practise (TBM). The use of two phases

of location verification was suggested. One level involved using tile-based techniques to ensure location information accuracy, while the second level involved the use of TBM to gather collective data on the announced location details for each vehicle, using data derived from all nearby vehicles using data obtained from all nearby vehicles. The proposed protocol has a flaw in that it does not describe a technique for evaluating network security in the VANET environment, which is problematic. Basically, it called into question long-standing security practises and proposed an Internet-based location information verification system. A further problem was the lack of a workable storage option for real-time data exchange.

The author's system [11] detecting abnormalities in city traffic and calculating bus arrival times are only two of the fog applications that are utilised to feed traveller information into the system. It is necessary to assess the dependability of these applications using real-world mobility data from a large vehicular testbed already in operation. It is ideal for this type of application to use Fog Computing with a small amount of current regional data, because its estimations of traffic anomalies and bus arrival times are quite near to those provided by the cloud. Fog apps, according to the findings of the network performance testing, can deliver accurate information in a fraction of the time while also decreasing traffic on the VANET backhaul network architecture.

The authors [3] look into the use of blockchain and SDN for VANET systems in 5G and fog computing, as well as other applications. Because of the omnipresent processing, the blockchain and SDN share managerial responsibilities, relieving the controller of these responsibilities. In addition, a trust-based network security paradigm is proposed. The simulation findings provide a high level of assurance for network performance while also promoting entity trust.

The authors proposed [12] a DCCS in ad hoc networks for safety applications in cars that only addresses network level security and ignores traffic flow. In essence, the proposed initiative aims to ensure that road users and drivers receive accurate and timely data when utilising safety applications on their mobile devices. The DCCS scheme's goal also included disseminating safety alerts to all network neighbours in a reliable and timely manner. Another problem with the proposed strategy is that there is no long-term infrastructure in place. Further complicating the matter was the lack of a dependable and effective storage mechanism for evaluating information that is updated in real time.

The protocol for mobile ad hoc networks, known as LER-GR, was proposed by authors [7] in order to primarily determine security at the network level. In the suggested LER-GR protocol, an attempt is made to evaluate the location error by employing a Rayleigh distribution-based error computation technique. With the use of the LER-GR protocol, it was determined which forwarding automobiles would be dispatched after them in order to ensure the least amount of error. In order to handle the mobility in a dynamic manner of the VANET, an alternative plan would have recommended an effective storage solution as well as intelligence pertaining to the communication of information about the location, among other things. Data transfer dependability and cost savings would have been secured as a result of this approach

Aside from that, there were no known network concerns that could compromise the safe transfer of location data.

The authors [2] proposed a solution for VANET that is secured time stable Geocast (S-TSG), but only at the network level. The suggested protocol was designed to detect flaws like denial of service (DoS) attacks due to VANET's decentralised and dynamic structure, to say nothing of its inability to regulate extraneous information and its restricted capacity. The suggested protocol did not include any evaluation of either efficient storage or a smart and safe technical solution for real-time data transmission in VANETs, as required by the proposed protocol. In addition, the protocol lacked the capability of interpreting real-time traffic information.

In order to safeguard the security of a network, the authors [6] suggested a geometry-based localization system (GeoLV) to make up for the lack of a GPS outages in a cyber-physical system for vehicles (VCPS). The efficiency of a GPS-assisted localization technique was demonstrated with the purpose of minimising location-aware neighbour restrictions in cooperative localization as a result of the technique's implementation. Vehicle dynamics and roadway trajectories would be major considerations in the GeoLV's design. To mitigate the consequences of a GPS outage for services based on location, re-locations, both static and dynamic a procedure was carried out using the proposed scheme. The use of GeoLV-based localization in the event of a GPS failure in the VCPS model has the disadvantage of not ensuring dependability, and no FC technique was recommended or defined. The proposed systems' node-level security detection proved to be a significant challenge.

3 Proposed Work

Figure 1 depicts to deploy nodes and creating vehicle attributes such as location and power consumption. When a node transfers a packet, how much power does it consume? Afterwards, the origin and destination are established. It is important to transmit data from one area to another in order to function properly. It is recommended that direct data be provided if that node is within its range; otherwise, it is recommended that the route request be broadcast. The respondent node will be assigned to a group with a k-number based on his or her response. In order to remain in the network, each respondent must contribute a percentage of their network share to the overall network. After that, the fog server computes the network key from the respondent-network share, which it does via E-Lagrange interpolation. There is a single global key that is maintained for the entire network; as a result, the global key is used to identify each vehicle in the network. It is necessary to utilise a shared mechanism since it is insecure to distribute the global key to a big number of cars at the same time. Depending on whether the car seeks information directly from a fog server or through an RSU, it will either issue a demand for three shares from any other vehicle on the network or will randomly select two from a pool of shares that are accessible. For the high-performance vehicle, it will be one of three total shares under consideration. The Lagrange polynomial will be utilised by the fog

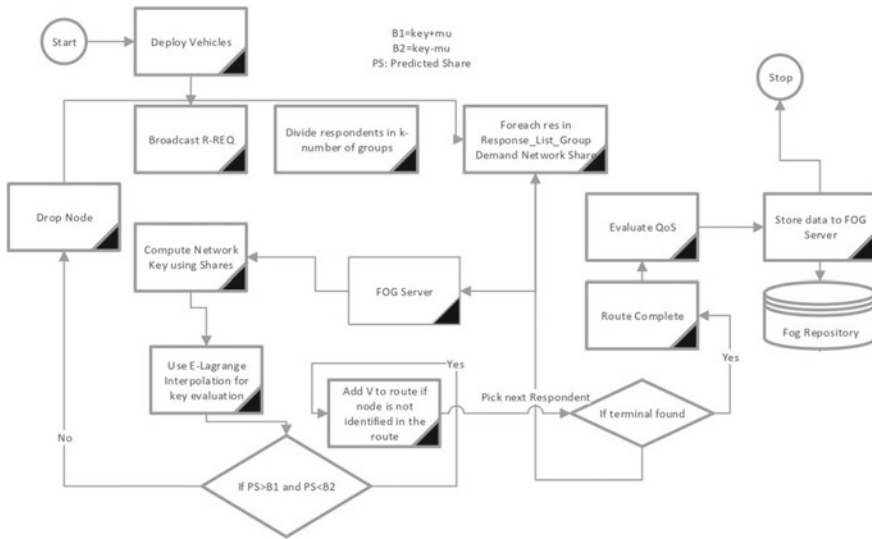


Fig. 1 Analysing a real-life node

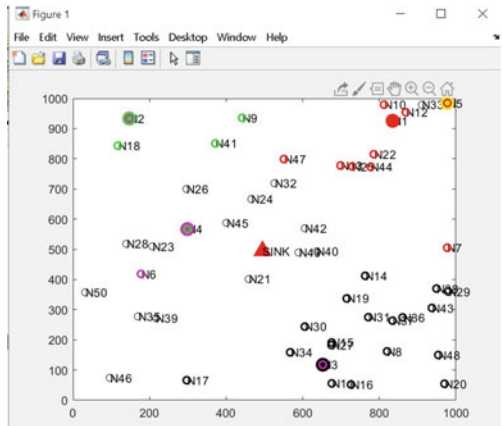
server in order to calculate the values shown below. Once this is completed, compute the projected share and compare it to B1 and B2. If the share falls within the range, route nodes are added, and if the share does not fall within the range, route nodes are deleted from the network. Following that, you will select the next responder and repeat the process until the destination has been reached; if the destination has been reached, there is no need to select the next responder. As a result, the route has reached its conclusion. Following that, the parameters that influence the overall quality of the service will be evaluated. It has now been kept in the Fog repository database as a parameter for the quality of the service.

4 Result Analysis

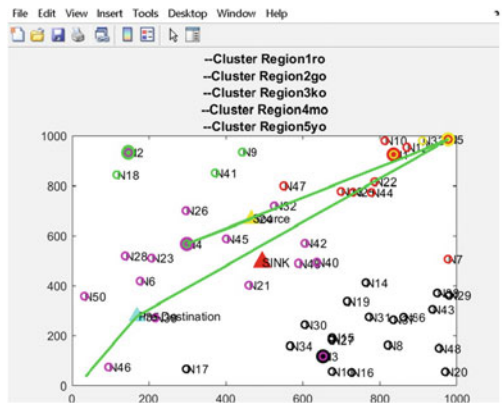
According to the proposed Authentic Vehicle Node Using FOG Computing (AVNFC), information aggregation is performed by the fog server (FS) and the road-side unit (RSU) using two layers of authentication. Specifically, fog and the RSU are both authenticated. When configuring a VANET, the RSU takes into account metrics such as throughput, power consumption, and packet delivery ratio; however, when constructing a VANET, the FL uses the Lagrange polynomial to identify nodes that are not trusted by the rest of the network.

In Fig. 2a, each vehicle is uniquely identifiable by the fog level global key, which is stored in a single location for the whole network. Insecure distribution of the global

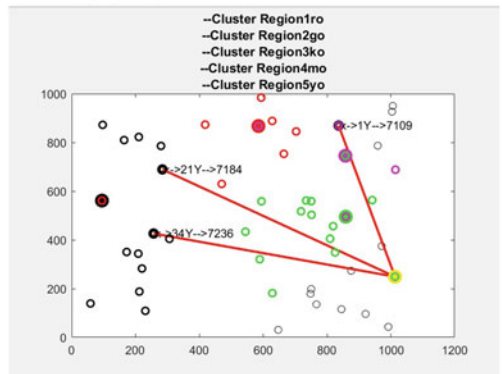
Fig. 2 Automated cluster head selection



(a) Finding node through Network Share Key



(b) Packet transferring among nodes



(c) Selection of cluster head

key across the vehicles is the reason why the vehicles adhere to a common scheme. A shared value is created by each vehicle.

Fig. 2b depicts a network of nodes that have been installed at various locations. As soon as an RSU or an RSU-connected vehicle requests information from a fog server, the fog server will demand three shares from any vehicle in the network or will randomly select two shares from a pool of available vehicles. Three total shares, including the demanding vehicle, will be examined for consideration. In Fig. 2c, the E-Lagrange technique was used to select the cluster head in this instance.

This is illustrated in Fig. 2 and can be seen in Table 1. We utilised the MATLAB network simulator to construct a network with 50 nodes that is distributed across a distance of 1000 m in length and a height of 1000 metres, as can be seen in Table 1. According to the best known estimations, the nodes travel at a speed of 50–100 m per second on average, depending on their location. As shown in Fig. 3, the following three quality of service parameters are measured and reported: throughput, energy consumption, and jitter. Specifically, five simulations were run, yielding an overall total of 11,000 packets in this particular instance. When finished evaluating each

Table 1 Network formation

Node count	50
Network elevation	1000m
Network width	1000m
Simulator	MATLAB
Node shifts	50–100 m/s
Packets injection rate	11,000/s

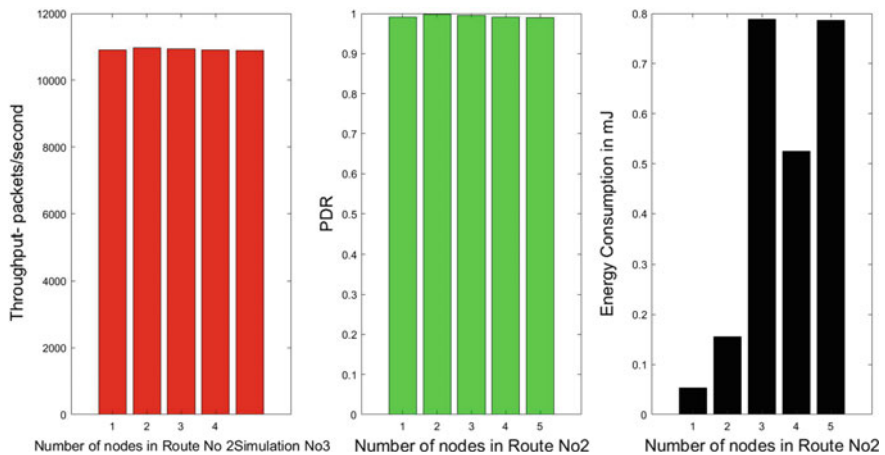


Fig. 3 Throughput, packet delivery ratio, energy consumption through machine learning

route, the throughput, jitter, and energy consumption of five different routes are calculated using the evaluation’s results.

It is possible to distinguish between different vehicles using the network shared key, which is managed by the fog server for the entire network and managed by the fog server for the entire network. Because of the security measures taken in the distribution of the network shared key in the vehicles, the vehicles are in compliance with the shared system as a result each vehicle possesses a particular shared value that is unique to it.

When an automobile requests the receipt of information obtained from a server, either directly or indirectly through the use of an RSU, the fog server (cluster head) will request the contribution of three shares from any vehicle in the network or will choose two at random from a pool of available shares when the automobile requests the receipt of information obtained from a server. Three total shares, including the demanding vehicle, will be considered in the decision-making process. For the following values, the Lagrange polynomial will be employed by the fog server (cluster head) to compute them on the fly:

The Lagrange polynomial $L(X)$ with *degree* $\leq (n - 1)$ requires n vehicles with respect to coordinates $x_1, y_1 = f(x_1), x_2, y_2 = f(x_2), \dots, x_m, y_m = f(x_m)$ and is provided by:

$$NS(x) = \sum_{i=1}^m PS_i(x)$$

where PS_i (Predicated Share) is provided by:

$$PS_i(x) = y_i \frac{x - x_k}{x_i - x_j} \text{ where } 1 \leq i, 1 \leq m \text{ and } k \neq j.$$

If $m = 3$ cars are specified explicitly, the equation will work.

$$NS(x) = y_1 \frac{(x-x_2)(x-x_3)}{(x_1-x_2)(x_1-x_3)} + y_2 \frac{(x-x_1)(x-x_3)}{(x_2-x_1)(x_2-x_3)} + y_3 \frac{(x-x_1)(x-x_2)}{(x_3-x_1)(x_3-x_2)}.$$

where NS represents Network Share.

To separate the polynomial, use the Lagrange equation’s fundamental interpolation.

$$NS_1(x) = y_1 \frac{x_2 * x_3}{(x - x_2)(x - x_3)} \text{ for the very first automobile}$$

$$NS_2(x) = y_2 \frac{x_1 * x_3}{(x - x_1)(x - x_3)} \text{ for the very second automobile}$$

$$NS_3(x) = y_3 \frac{x_1 * x_2}{(x - x_1)(x - x_2)} \text{ for the very third automobile.}$$

When several polynomials are integrated, the network key that comes from this integration is denoted by

NGK stands for network global key

$$NGK(x) = \sum_{k=1}^m NS_i(x).$$

Vehicles are added to the pool of vehicles and transmit data to the fog server only when the network global key (NGK) matches the predicted Share (PS).

5 Conclusion

It is proposed in this article that Authentic Vehicle Node utilising FOG Computing (AVNFC) be used as a VANET architecture that is fog-enabled. When developing the proposed solution, it is important to consider both the security of individual nodes and the overall network security at the same time. Node security encourages trust and collaboration among all network neighbours by providing a safe environment. By ensuring that packets are delivered as soon as feasible, the integrity of the network system at the node level is maintained and maintained. Keeping outsiders from accessing the network improves latency and increases the amount of packets sent per second, both of which are beneficial.

Besides that, the model is also used to communicate real-time network information to the TMO's RSU, which helps to reduce delays while simultaneously increasing the network packet delivery ratio by increasing the network packet delivery ratio. The proposed AVNFC method is evaluated based on the PDR and jitter QoS criteria in order to determine the overall performance of the system.

References

1. Devi A, Kait R, Ranga V (2019) Security challenges in fog computing. In: Handbook of research on the IoT, cloud computing, and wireless network optimization. IGI Global, pp 148–164
2. Dora DP et al (2016) Secured time stable geocast (S-TSG) routing for VANETs. In: Proceedings of 3rd international conference on advanced computing, networking and informatics. Springer, pp 161–167
3. Gao J et al (2019) A blockchain-SDN-enabled Internet of vehicles environment for fog computing and 5G networks. *IEEE IoT J* 7(5):4278–4291
4. Grover J et al (2018) Real-time vanet applications using fog computing. In: Proceedings of first international conference on smart system, innovations and computing. Springer, pp 683–691
5. Hasrouny H et al (2017) VANet security challenges and solutions: a survey. *Veh Commun* 7:7–20
6. Kaiwartya O et al (2018) Geometry-based localization for GPS outage in vehicular cyber physical systems. *IEEE Trans Veh Technol* 67(5):3800–3812
7. Kasana R et al (2017) Location error resilient geographical routing for vehicular ad-hoc networks. *IET Intel Transp Syst* 11(8):450–458
8. Lai Y et al (2017) Data gathering framework based on fog computing paradigm in vanets. In: Asia-Pacific web (APWeb) and web-age information management (WAIM) joint conference on web and big data. Springer, pp 227–236
9. Panayappan R et al (2007) VANET-based approach for parking space availability. In: Proceedings of the fourth ACM international workshop on vehicular ad hoc network, pp 75–76
10. Parveen RS, Kumar S (2021) IoV based intelligent vehicle tracker using FoG computing with supervised machine learning techniques. *J Discrete Math Sci Cryptogr* 24(5):393–1413
11. Pereira J et al (2019) Assessing the reliability of fog computing for smart mobility applications in VANETs. *Future Gen Comput Syst* 94:317–332
12. Qureshi KN et al (2018) A dynamic congestion control scheme for safety applications in vehicular ad hoc networks. *Comput Electric Eng* 72:774–788
13. Sheet DK et al (2015) Location information verification cum security using TBM in geocast routing. *Procedia Comput Sci* 70:219–225

14. Shehab M, Khader AT, Al-Betar MA (2017) A survey on applications and variants of the cuckoo search algorithm. *Appl Soft Comput* 61:1041–1059
15. Tangade S, Manvi SS, Lorenz P (2018) Decentralized and scalable privacy-preserving authentication scheme in VANETs. *IEEE Trans Veh Technol* 67(9):8647–8655
16. Truong Nguyen B, Myoung LG, Yacine G-D (2015) Software defined networking-based vehicular adhoc network with fog computing. In: *IFIP/IEEE (IM) international symposium on integrated network management*. IEEE, pp 1202–1207

Neural Network-Based BLDC Motor Drive for Electric Vehicle Application



Kishore Kumar Pedapenki

Abstract This paper presents an investigation of speed control of BLDC drive with VSI between PI and neural network controllers. Zeta converter is utilized for controlling the DC interface voltage since it is having buck-support capacity and better improvement of power factor contrasted with boost converter. The power factor correction (PFC)-based zeta converter is intended to work in discontinuous inductor current mode. The essential regulator utilized for speed control of engine utilized is PI controller. In this work, PI regulator-based BLDC drive and neural-based BLDC drive are performed in MATLAB and compared. The re-enactment results show neural network controller-based speed regulator lessens the ascent time and gives quick speed reaction. The created neural network controller has capacity to adapt promptly and adjust its own control parameters dependent on aggravations with least state error, overshoot, and rise time. The exhibition of the proposed drive is approved with outcomes got on a MATLAB/Simulink.

Keywords PI controller · Neural network controller · BLDC motor · Zeta converter

1 Introduction

BLDC motor is an ideal machine for all powers except high power because of its big torque/inertia ratio, big effectiveness, big energy thickness, wide scope of speed control, and little support necessity. It is a 3- ϕ synchronous machine with extremely durable magnets on the rotatory part and 3- ϕ windings on the stationary part. It is otherwise called an electronically exchanged motor on the grounds that there are no brushes and commutator, rather an electronic commutator is utilized depending on the position of the rotor detected by the Hall effect sensor. Applications include a wide scope of home devices, office computerization, modern instruments, ventilation, auto, cooling, and so on.

K. K. Pedapenki (✉)

Electrical and Electronics Engineering, Jain (Deemed to be University), Bengaluru, India
e-mail: iitr.kis1@gmail.com

Owing to the vigorous presentation over a wide scope of working conditions and utilitarian effortlessness, the PI controller is very popular. Utilizing PI, the drive controlled on the grounds that it used to arrive at speed accurately and diminishes consistent state blunder. The speed reaction of the drive is slow because of integral part of the controller. Since PI regulator does not anticipate the future blunders of the framework, subsequently it cannot dispose of consistent state motions and decreases settling time. Neural network has quickly become one of the best of the present innovation for creating complex control frameworks. A few investigations show, the outcomes that neural network control yields better outcomes with deference than those got by traditional control calculations. Along these lines, in modern applications, the neural network control has turned into an appealing arrangement in controlling the electrical drives. Neural network controller [1, 2] is one of the capable controllers [3] in reactive power compensation [3–9] also. The experimental arrangement dependent on the neural network was likewise evolved around, here of shunt active filters [8, 9]. In this paper, MATLAB/Simulink-based-simulated model with the neural network controller and zeta converter is introduced.

2 Power Factor Corrected Zeta Converter

The power factor corrected (PFC) zeta converter [10–12] is a four quadrant DC-DC converter that works in both current conduction modes. It operates as a buck boost feature that is a non-inverted response. The schematic diagram is shown in Fig. 1, and its waveforms are shown in Fig. 2.

The schematic diagram (Fig. 1) contains a MOSFET which acts as a switch, two inductors and two capacitors with a diode at the middle of combination of L and C. The waveforms of switching voltage, DC voltage, and voltage across C1, Currents passing through L1, L2, and Diode are shown in Fig. 2.

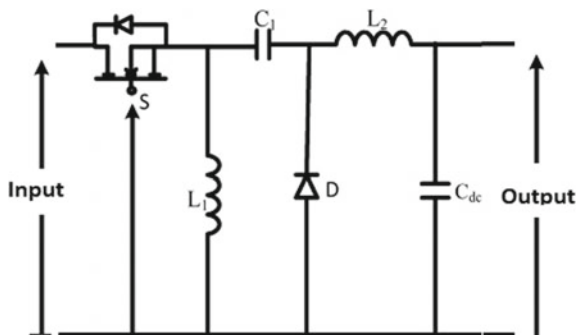


Fig. 1 Schematic diagram of zeta PFC converter

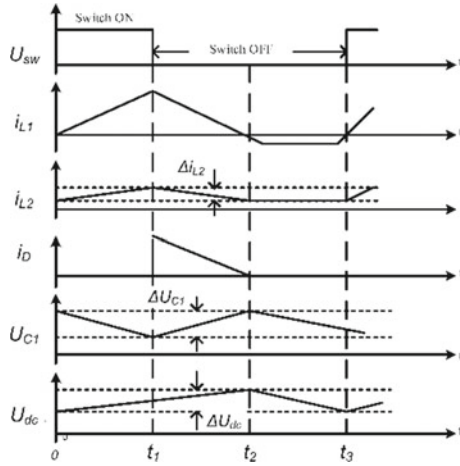


Fig. 2 Wave forms of zeta PFC converter

Steady State Analysis [13]

Utilizing the inductors [14, 15] voltage and time balance equations, the relationship between stimulus and response voltage can be obtained as shown in Fig. 3a, b.

Let the on period of switch is DT_s in the total period of T .

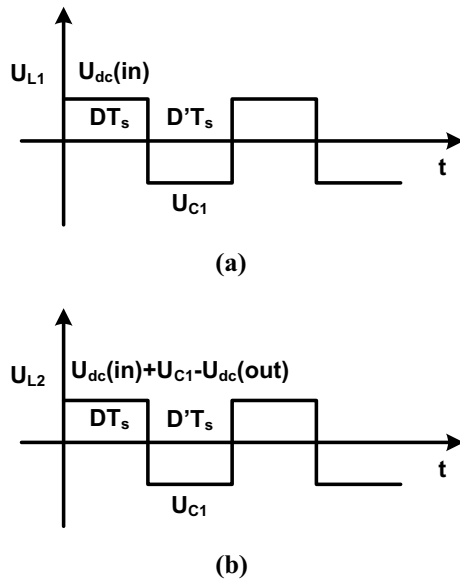


Fig. 3 a Voltage appeared at inductor L_1 [16] and b voltage appeared at inductor L_2 [17]

Inductor Volt-Sec Balance Equation

From Fig. 3a

$$U_{\text{dc(in)}}DT_s - U_{C1}D^1T_s = 0 \quad (1)$$

$$U_{\text{dc(in)}}DT_s = U_{C1}D^1T_s \quad (2)$$

$$U_{C1} = \frac{U_{\text{dc(in)}}D}{D^1} \quad (3)$$

From Fig. 3b

$$(U_{\text{in(dc)}} + U_{C1} - U_{\text{dc(out)}})DT_s - U_{\text{dc(out)}}D^1T_s = 0 \quad (4)$$

$$(U_{\text{in(dc)}} + U_{C1} - U_{\text{dc(out)}})DT_s = U_{\text{dc(out)}}D^1T_s$$

$$U_{C1} = \frac{U_{\text{dc(out)}}}{D} - U_{\text{dc(in)}} \quad (5)$$

From Eqs. (3) and (5)

$$\frac{U_{\text{dc(out)}}}{D} - U_S = \frac{U_{\text{dc(in)}}D}{D^1} \quad (6)$$

$$\frac{U_{\text{dc(out)}}}{D} = U_{\text{dc(in)}} + \frac{U_{\text{dc(in)}}D}{D^1} \quad (7)$$

$$U_{\text{dc(out)}} = \frac{U_{\text{dc(in)}}D}{(1-D)} \quad (8)$$

$$\frac{D}{1-D} = \frac{U_0}{U_{\text{in}}} = \frac{I_{\text{in}}}{I_o} \quad (9)$$

$$D = \frac{U_{\text{dc(out)}}}{U_{\text{dc(out)}} + U_{\text{dc(in)}}} \quad (10)$$

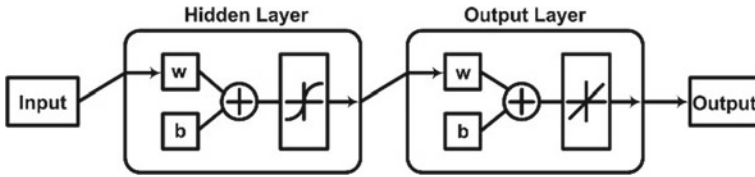


Fig. 4 Architecture of artificial neural network

3 Neural Network Controller (NNC)

Neural network [4] is one of the bio-motivated frameworks. It is motivated from organic neuron. The organic neuron works dependent on the preparation given by the parents and the climate. Likewise, the neuron in the NN must be prepared appropriately to get the necessary outcome. The information which are given to the neural network is partitioned into three sorts: training, validating, and testing.

- (1) *Architecture of NNC* [18]: The structure of the two layered neural network used in this work is show in Fig. 4. The neural network controller used here is having two computation layers. One is hidden layer and second one is the output layer. The sigmoid function is used as the active function in the hidden layer, and a linear function is used as the activation function for the output layer.
- (2) *Implementation of NNC* [19, 20]: The flowchart describing the step by step procedure to implement the neural network controller in the MATLAB is explained in Fig. 5. This procedure is followed for utilizing the neural network tool box in MATLAB/Simulink.

The various parameters utilized in this work are tabulated in Table 1 which contains the inductors, capacitors as shown in Fig. 6.

4 Methodology and Result Analysis

The BLDC drive includes rectifier, direct current filter, zeta converter, VSI, and brushless DC motor [14, 22]. The 1- ϕ AC supply is fed to zeta converter, after it has been converted to DC by using a diode bridge rectifier and DC filter. The output of diode bridge rectifier contains harmonics. To eradicate these harmonics, DC filter is utilized. Now the input to the zeta converter is a rectifier fed DC Voltage. Zeta converter can improve the power factor of the system when compared to the counter part of the boost converter.

Zeta converter is utilized to provide the input to VSI which feeds the BLDC motor. The control signals for switch in the zeta converter are provided by the speed of the BLDC motor. The speed of the motor is the actual speed which is compared with the reference speed given by the mathematical modelling. The difference between

Fig. 5 Flowchart for the implementation of neural network in MATLAB

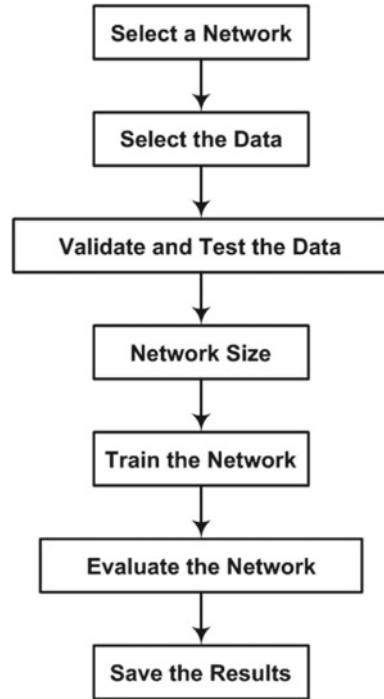


Table 1 Various parameters utilized in this work [21]

S. No.	Formulae	Values
1	$L_1 = \frac{U_s^2}{2P_{\max} f_s} \left(\frac{U_{dc \max}}{U_{dc \max} + \sqrt{2}U_s} \right)$	1.29 mH
2	$C_1 = \frac{P_{\max}}{\eta (\sqrt{2}U_s + U_{dc \max})^2 f_s}$	423.06 nF
3	$L_o = \frac{U_s^2}{P_{\max}} \frac{U_{dc \max}}{\lambda f_s (\sqrt{2}U_s)} \left(\frac{U_{dc \max}}{U_{dc \max} + \sqrt{2}U_s} \right)$	3.89 mH
4	$C_{dc} = \frac{P_{\min}}{2\omega k U_{dc \min}^2}$	5 mF
5	$C_{f \max} = \frac{I_m}{\omega_L U_m} \tan \theta$	378 nF
6	$L_f = \frac{1}{4\pi^2 f_c^2 C_f}$	1.03 mH

them is the error signal which is processed in an ANN controller. The output f ANN controller is again compared with the saw tooth generator in the PWM generator. The final output of the PWM generator is the gating signals to the switch in zeta converter. So, the zeta converter will provide the power factor improved input to VSI.

The position of the rotor of BLDC motor is extracted and sent to electronic commutator to provide the gating signals to VSI. In this way, the output of the VSI is purified twice with two control strategies, viz. One is giving signals to switch

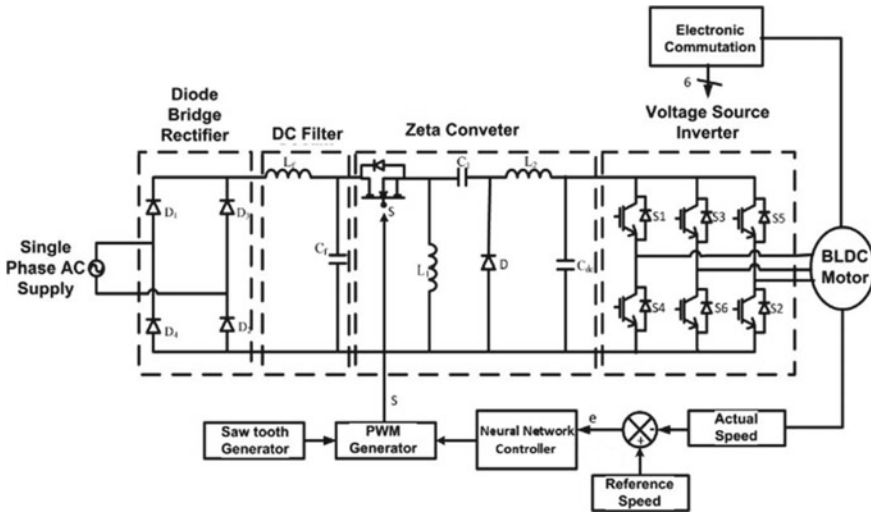


Fig. 6 Zeta PFC converter for BLDC drive with neural network controller

of zeta converter, and the other is giving the gating signals to VSI. For the first one, the input is speed of the BLDC, and for the second one, the input is the position of the rotor. It is clearly shown in Fig. 6. The neural network is used as shown in Fig. 6. The input for the neural networks is the error generated by the difference of the actual speed and the reference speed. The corresponding signals generated by the NN controller is fed to the PWM generator which can generate the signal to the MOSFET by comparing the same with a saw tooth generator. The difference between the PI and NNC are clearly visible in all the diagrams with various parameters. The correlation of the PI and NNC in numbers are displayed in Table 2.

Figure 7 shows the DC interface voltage taken after the DC filter shown in Fig. 6. The first diagram is shown with PI controller (conventional controller), and the second diagram is with NN controller (bioinspired controller). The difference is clearly shown in the diagram in settling the waveform at the steady state value of 250 V. With the PI controller, it takes 2.10 s to reach the steady state value whereas with the NN controller, it only takes 0.18 s.

Figure 8 shows the speed response of BLDC motor as shown in Fig. 6. The first diagram is shown with PI controller (conventional controller) and the second diagram

Table 2 Comparison between PI and neural network controller

Parameters	PI controller	Neural network controller
Power factor	0.91	0.98
Settling time (s)	2.10	0.18
Rise time (s)	2.10	0.06
Voltage ripple (V)	2.50	2.30

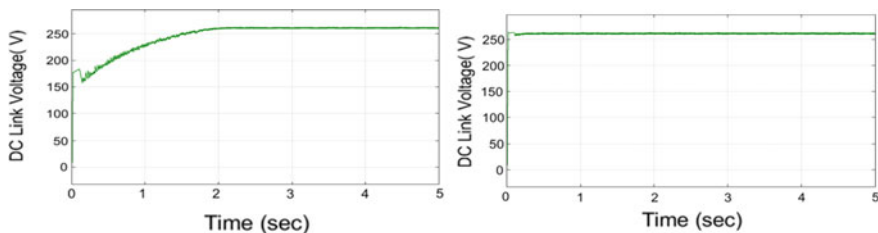


Fig. 7 DC interface voltage (V) using proportional–integral controller and neural network controller

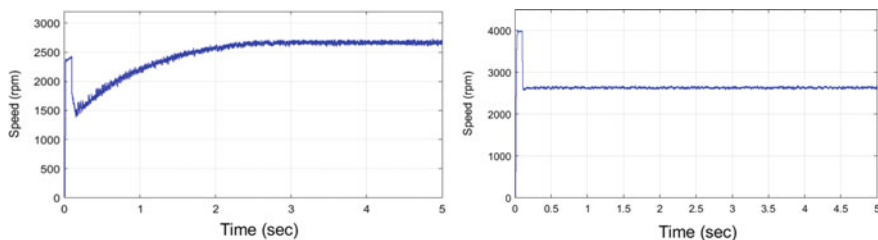


Fig. 8 Speed response of BLDC motor using proportional–integral controller and neural network controller

is with NN controller (bioinspired controller). With PI controller, the speed reaches 2500 rpm after 2.5 s and with NN controller, it reaches 2500 rpm after 0.15 s only.

Figure 9 shows the stator current of BLDC motor as shown in Fig. 6. The first diagram is shown with PI controller (conventional controller) and the second diagram is with NN controller (bioinspired controller). With PI controller, it has two cycles for 0.15 s whereas with NN controller, it has four cycles in 0.15 s. The frequency of the current is doubled with NN controller compared to PI controller. The magnitude with PI controller is maximum around 4.8 A, and it is around 6 A with NN controller.

Figure 10 shows the electromagnetic torque. The first diagram is shown with PI controller (conventional controller) and the second diagram is with NN controller (bioinspired controller). The frequency of the electromagnetic torque is doubled

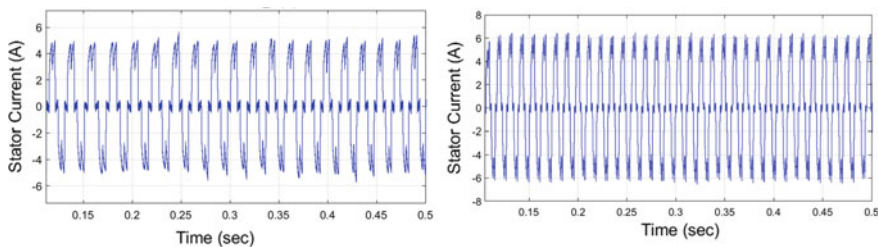


Fig. 9 Stator current using proportional–integral controller and neural network controller

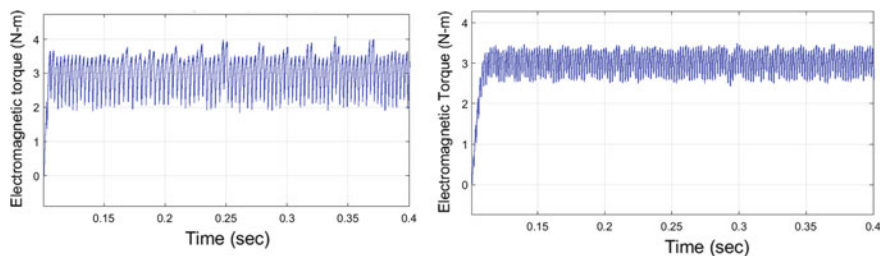


Fig. 10 Electromagnetic torque using proportional–integral controller and neural network controller

with NN controller compared to PI controller. The oscillations are very huge in first diagram (PI controller) when compared to second one (NN controller).

5 Conclusion

The PF corrected zeta converter-based BLDC utilizing NNC has simulated using MATLAB/Simulink. This work presents a relative examination of speed adjustment of the BLDC drive. In spite of using boost converter, the zeta converter is utilized which has buck boost property. So, it can provide better PF improvement. The speed of the BLDC utilizing PI controller takes more time to rise and settle to the steady state condition. In this work, the NNC is proposed to eradicate this problem. It has less rise and settling time with further developed PF. The speed adjustment of the BLDC drive with zeta converter controlled by PI and NN controllers are examined, and the simulation results were obtained in MATLAB/Simulink. By using the results obtained by these two controllers, the neural network has given a very good results in power factor, settling time, rise time, and voltage ripple.

References

1. Saleh AL, Obed AA, Qasim HH, Breesam WH, Al-Yasir YA, Parchin NO, Abd-Alhameed RA (2020) Wavelet neural networks for speed control of BLDC motor, automation and control. In: Voloşencu C, Küçük S, Guerrero J, Valero O (eds) IntechOpen
2. Pedapenki KK, Swathi G (2017) Application of genetic algorithm in electrical engineering. Int J Pure Appl Math 114(8):35–43
3. Pedapenki KK, Anumeha KJ (2021) Fuzzy logic controller-based BLDC motor drive. In: Kumar J, Jena P (eds) Recent advances in power electronics and drives. Lecture notes in electrical engineering, vol 707. Springer, Singapore. https://doi.org/10.1007/978-981-15-8586-9_34
4. Pedapenki KK, Gupta SP, Pathak MK (2016) Shunt active power filter with MATLAB and d'SPACE 1104 verification. Int J Appl Eng Res 11:4085–4090

5. Pedapenki KK, Gupta SP, Patha MK (2015) Comparison of fuzzy logic and neuro fuzzy controller for shunt active power filter. In: International conference on computational intelligence and communication networks (CICN), pp 1247–1250
6. Gobinath S, Madheswaran M (2020) Deep perceptron neural network with fuzzy PID controller for speed control and stability analysis of BLDC motor. *Soft Comput* 24:10161–10180
7. Shijith CC, Shijith S (2014) PFC and speed control of BLDC motor using zeta converter. *Int J Adv Res Electr Electron Eng* 2(2):52–58
8. Janardhan EG (2007) Special electrical machines. PHI Publications
9. Pedapenki KK, Gupta SP, Pathak MK (2015) Shunt active power filter with artificial intelligent controllers. In: IEEE—international conference on control, instrumentation, communication and computational technologies (ICCICCT), pp 74–77. <https://doi.org/10.1109/ICCICCT.2015.7475252>
10. Singh B, Bist V (2015) Power quality improvements in a zeta converter for brushless DC motor drives. *IET Sci Meas Technol* 9(3):351–361
11. Bist V, Singh B (2014) A brushless DC motor drive with power factor correction using isolated-zeta converter. *IEEE Trans Industr Inf* 9(2):1–6
12. Singh S, Singh B, Bhuvaneswari G, Bist V (2015) Power factor corrected zeta converter based improved power quality switched mode power supply. *IEEE Trans Industr Electron* 62(9):5422–5433
13. Kumar R, Singh B (2016) BLDC motor-driven solar PV array-fed water pumping system employing zeta converter. *IEEE Trans Industr Appl* 52(3):2315–2322
14. Tibor B, Fedák V, Durovský F (2011) Modeling and simulation of the BLDC motor n MATLAB GUI. In: IEEE international symposium on industrial electronics, pp 1403–1407
15. Singh B, Singh S (2011) Isolated zeta PFC converter-based voltage controlled PMBLDCM drive for air-conditioning application. In: India international conference on power electronics, pp 1–5
16. Mamadapur A, Unde Mahadev G (2019) Speed control of BLDC motor using neural network controller and PID controller. In: 2nd international conference on power and embedded drive control (ICPEDC), pp 146–151
17. Singh S, Singh B (2010) Voltage controlled PFC zeta converter based PMBLDCM drive for an air-conditioner. In: 5th international conference on industrial and information systems. IEEE, pp 550–555
18. Pedapenki KK, Gupta SP, Pathak MK (2015) Comparison of PI and neural network based controllers for shunt active power filter. In: International conference on control, instrumentation, communication and computational technologies (ICCICCT), pp 214–218
19. Pedapenki KK (2019) Power quality with ANN and hysteresis controllers. In: IEEE—international conference on intelligent computing, instrumentation and control technologies (ICICT), pp 328–334. <https://doi.org/10.1109/ICICT46008.2019.8993197>
20. Pedapenki KK, Gupta SP, Pathak MK (2015) Soft control techniques for shunt active power filter. In: IEEE Power, communication and information technology conference (PCITC), pp 60–65. <https://doi.org/10.1109/PCITC.2015.7438075>
21. Singh B, Bist V (2012) A single sensor-based PFC zeta converter FED BLDC motor drive for fan applications. In: IEEE fifth power India conference. IEEE, pp 1–6
22. Pedapenki KK, Gupta SP, Pathak MK (2015) Application of neural networks in power quality. In: International conference on soft computing techniques and implementations (ICSCTI), pp 116–119. <https://doi.org/10.1109/ICSCTI.2015.7489615>

Rule Placement-Based Energy-Aware Routing in SDN: Review



Rachid Ben Said , Sakirin Tam , and Omer Ozgur Tanriover 

Abstract The widespread use of cloud computing infrastructure increases energy consumption issue in data centres. Many network providers and researchers investigated and implemented different approaches to optimise energy in networks. Meanwhile, software-defined network (SDN) has emerged as a new possibility to save energy because it can separate control plane and data plane in order to optimise routing and enables flexible control in the network. To maintain administrative and technical requirements, defining and placing rules for routing policies in the network are an important concern. In this study, the authors conducted a systematic literature review of rule placement approaches for energy-aware routing (EAR) in SDNs. The search identified 965 articles from Web of Science, Scopus, IEEE Xplorer, ScienceDirect, Springer Linked, EBSCOhost, Emerald and Google Scholar, published since 2013. After multi-stages of selection process, 23 articles are selected for the review study. The study aims to discuss approaches and their corresponding methods used in rule placement for EAR. EAR approaches in implementing rule placement achieved good result but with limitations. The authors analyse how the rule placement approaches affect the performance of the network to address future research attentions for further EAR improvement.

Keywords Rule placement · Energy-aware routing · Software-defined networks · OpenFlow

R. B. Said (✉) · S. Tam · O. O. Tanriover
Graduate School of Natural and Applied Science, Ankara University, Ankara 06860, Turkey
e-mail: bensaid@ankara.edu.tr

S. Tam
Faculty of Science and Information Technology, Phnom Penh International University, Phnom Penh 12253, Cambodia

1 Introduction

Today's computer networks are made up of a range of heterogeneous devices (e.g. switch, router and middlebox) from many vendors, all of which are running a variety of complex and dispersed protocols. In traditional networks, router used to distribute packets by routing protocols such as Open Shortest Path First (OSPF) to determine which interface messages should be passed on [1].

Software-defined networking (SDN) promotes separating forwarding devices from the software that controls them to reduce reliance on a single equipment manufacturer and simplify network management. OpenFlow, in particular, implements a section of the SDN approach through a simple but powerful protocol that abstracts network communications into flows to be analysed by interim network elements using only a fixed number of primitives [1].

SDNs also strive to use flow-based forwarding rule rather than destination-based rule (as in classic router) to enable finer network traffic control. Rule placement is a technique which focuses on how to place the rules in the switches, effectively. Routing in SDNs is defined by control plane and forward data plane. The first packet of each new flow, the switch request triggers flow initiation to the controller. Then, the controller defines a path for the flow packet, and the decision of routing is based on the flow table entries. There is a rule, and its corresponding action is used to match with each flow table entry. The primary two approaches of rule placement are reactive or proactive. However, all rules in rule placement assume a global view of the network, and the efficiency of rules is depending on the complexity of the constraints such as rule meaning, optimise routing, rule space reduction and wildcard.

In this review study, the authors survey the approaches that were deploying SDN-based energy-aware routing (EAR) respecting both constraints: the capacity of the link not lost and the size of forwarding tables reduced so that energy consumption of the network is minimised. In addition, the authors discuss EAR methods and potential future researches.

The paper is organised as follows: Section 2 presents description on materials and methods. Section 3 provides results and discussion. Section 4 provides discussion on open issues on rule placement, QoS in SDN and direction for further works. Section 5 presents the paper's conclusion by remarks.

2 Materials and Methods

The authors conducted systematic literature review based on the original guidelines described by Kitchenham et al. [2]. The aims of a systematic literature review are: "... to identify, evaluate and interpret all available research relevant to a particular research question, or topic area or phenomenon of interest. Individual studies contributing to a systematic review are called primary studies; a systematic review

is a form of secondary study”. Based on guidelines of Kitchnham et al. [2], the steps of the study are conducted as below.

2.1 Research Question

SDNs have been attracting a growing attention in the network community in recent years. It is a new networking paradigm that decouples the control plane from the data plane. It provides a flexibility to develop, test new network protocols and policies in real networks. But in practice, TCAMs are an expensive and power hungry. To tackle this problem, many existing solutions implement rule placement in OpenFlow. Therefore, the study aims to provide a comprehensive understanding of rule placement, how rule placement algorithms are implemented and what are the potential of new research in rule placement for energy-aware routing. The authors’ study address research questions as follows:

- RQ1: What are the available rule placement approaches used for EAR in SDNs?
- RQ2: What algorithms are implemented in rule placement?
- RQ3: What are the impacts of rule placement approach on Quality of Service (QoS)?

To address RQ1, the authors conducted search strategy to identify promising research in EAR using rule placement approach from recognised online databases including Web of Science, Scopus, IEEE Xplorer, ScienceDirect, Springer Linked, EBSCOhost, Emerald and Google Scholar. Within the scope of the rule placement-based EAR, the authors answer RQ2 and RQ3 in order to analyse the algorithms used in rule placement approaches and how it affects the performance of network. With respect to RQ4, the authors address the limitation of current studies and define possible improvement for EAR in SDNs.

2.2 Search Terms

With respect to RQ1, the authors created primary search keywords: “rule placement, energy-aware routing and software-defined network”. Then, a search string is constructed based on the primary search keywords to search from online databases. The search string is formulated with different synonyms for each keyword. Table 1 shows the keywords and its synonyms.

Table 1 shows list of search keywords and synonyms.

To design the search string, the authors use the Boolean operators (“AND”, “OR”) to connect the keywords together. The search string is as follows:

(Rule placement OR rule placement approach OR rule placement algorithms), AND (for energy-aware routing OR energy consumption OR energy reduction), AND (in software-defined networks OR data centre).

Table 1 Search keywords for search strategy

Category	Keywords and synonyms
Network	Software-defined networks,
Energy aware routing	Energy-aware routing, energy consumptions and energy reduction
Rule placement	Rule placement, rule placement approach and rule placement algorithm

2.3 Search Strategy

Kitchenham et al. [2] recommended conducting search from different online databases. Therefore, the authors perform the search on eight different online databases including both the journal and conference articles as follows: Web of Science, Scopus, IEEE Xplorer, ScienceDirect, Springer Linked, EBSCOhost, Emerald and Google Scholar. These databases cover the majority of journal and conference proceeding, technical and books that allow the authors to perform and locate potentially relevant studies. Trial searches were also performed for a number of search strings that were constructed with a combination of keywords and synonyms as given in Table 1.

2.4 Study Selection Criteria and Procedure

Metaheuristic search was applied based on the search string on different online databases. In order to cover wide range of the study, initially the search string is defined widely. The search found many studies including out of the scope of energy aware routing using rule placement in SDNs. To exclude irrelevant articles those which do not provide information for research question, the authors developed inclusion/exclusion criteria.

Inclusion criteria

- Articles those are published since 2013.
- Articles those provide information for the research questions.

Exclusion criteria

- Papers those primary focus were not discussing about rule placement for energy aware routing.
- Papers those discuss energy aware routing but do not discuss rule placement approach.
- Papers those full text were not available.
- Document which is in the form of slide presentation.
- Workshop summary, patents and unpublished reports.

2.5 Search Quality Assessment

To guide the interpretation of finding for data analysis and synthesis, the authors have preliminary applied study quality assessment [3]. The quality of each selected study was evaluated against a quality criteria checklist. The criteria use binary scale, which is a “Yes” and “No” to avoid misinterpretation of the result [4]. It is because the aim was not to rank the studies according to the overall quality score. Quality criteria checklist as the following display:

- A1: Are the aims of the research sufficiently explained?
- A2: Is the study context adequately described?
- A3: Is method adequately described?
- A4: Is measure use in the study well defined?
- A5: Is research finding clearly stated?
- A6: Does the conclusion relate the aims of proposed research defined?

2.6 Data Extraction

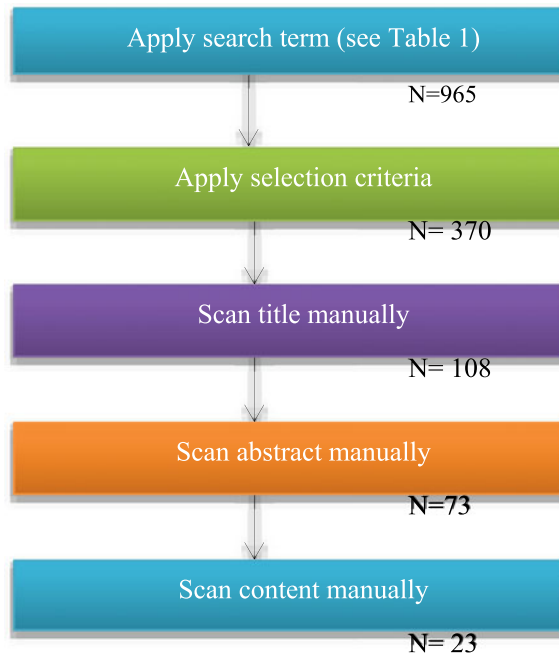
Data extraction is executed based on the study selection criteria mentioned in the section above. The papers which provided adequate discussion on energy aware routing using rule placement techniques and published since 2013 are included for further analysis. On the other hand, the papers which do not discuss the rule placement for energy aware routing or discuss only energy aware routing but do not discuss rule placement, or papers those full text are not available or full text those are in the form of slide presentation or workshop summary or patents or unpublished reports are excluded. Figure 1 illustrates the stages to execute the search. Each stage shows how the papers are filtered and the quantity of papers are counted.

There were 965 articles found when search strings were applied at the initial stage of the search process. After applying study selection criteria, there were 108 articles recorded. Title and abstract skimming were conducted to find the relevant studies which addressing rule placement for energy aware routing. There were 72 articles related about the issue. Then, moving one more stage to review detail of the articles manually in order to get the most relevant studies of rule placement used for energy optimisation in SDNs. There were 23 articles considered for the study.

3 Results and Discussion

In this section, the authors provided an evaluation of energy aware routing using rule placement in SDNs. The result of the study is structured based on the research questions as the following:

Fig. 1 Paper selection process



RQ1: Rule placement approaches

Rule placement is a technique, which focuses on how to place the rules in the switches, effectively. Routing in SDNs is defined by control plane and forward data plane. The first packet of each new flow, the switch request triggers flow initiation to the controller. Then, the controller defines a path for the flow packet, and the decision of routing is based on the flow table entries. There is a rule, and its corresponding action is used to match with each flow table entry. The primary two approaches of rule placement are reactive or proactive. However, all rules in rule placement assume a global view of the network, and the efficiency of rules is depending on the complexity of the constraints such as rule meaning, optimise routing, rule space reduction and wildcard [5–10].

Overall, there are 23 relevant studies from the sources are identified to provide rule placement approaches for energy optimisation in SDNs, as given in Table 2. There are 12 studies discussed proactive approach, six studies discussed reactive approach, five studies discussed rule meaning approach, six studies discussed compromised routing approach, 19 studies discussed rule space reduction approach, and ten studies discussed the wildcard approach.

Proactive rule placement: In this approach, the controller populates the rules ahead of time before the traffic matches of the new packet flow arrive into the switch. Figure 2 [7] shows the process of how a flow is placed and forwarded, proactively. The rules are pre-installed in the controller and pushed to respective switches (Arrow

Table 2 Rule placement approaches for energy efficiency

ID	Approaches	Proactive	Reactive	Rule space reduction	Rule meaning	Compromised Routing	Wildcard
1	MINNIE-Extended [5]	✓	✓	✓			✓
2	MINNIE [6]	✓		✓			✓
3	OFFICER [7]	✓		✓	✓	✓	
4	Two-dimensional compression [8]	✓			✓		✓
5	Optimising rule placement [9]	✓		✓	✓	✓	
6	Big Switch [10]	✓	✓	✓			
7	NNIRSS [11]			✓		✓	
8	Optimised flow management mechanism [12]		✓				✓
9	Rule placement schemes [13]			✓			
10	OPTree [14]	✓		✓		✓	
11	FlowStat [15]			✓			✓
12	Compression in EAR [16]			✓			✓
13	Reducing reconfiguration cost of flow tables [17]			✓			
14	Raptor [18]	✓	✓	✓		✓	
15	TRPS [18]			✓			✓
16	Reduce flow configuration [20]			✓			
17	Hybrid rule placement [21]			✓	✓		✓
18	RPCFE [22]			✓	✓		
19	PARD [23]	✓	✓				
20	FTRS [24]	✓		✓			
21	Wildcard [25]	✓		✓		✓	✓
22	MIRA-RA [26]		✓				
23	Palette [27]	✓		✓			

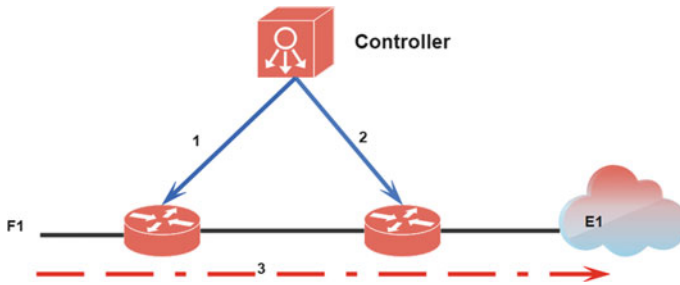


Fig. 2 Proactive rule placement

1 and Arrow 2) before the arrival of flow F1 (Arrow 3). This approach does not have setup delay which enable forwarding traffic to become faster. The most studies of proactive approach is mainly focused on access control [7, 8, 10, 18, 23–25, 27]. In access control, the rules are predefined by the operator, independently of the traffic. However, proactive approach is also used to decide forwarding rule, but it requires prior paths calculation and accurate traffic prediction [9, 14].

Reactive rule placement: In this approach, the rules are generated to react on demand to flow events. The switch enqueues the packet and looks up the rules in the flow table for every new flow entry. Then, the switch informs the new arrival flow to the controller. The controller generates rule corresponding to the instruction and push it back to respective switches. Afterwards, the switch dequeues and forwards the packet in the network. The process of packet handling in reactive approach is described in Fig. 3 [7]; the flow F1 is queued at two switches (Arrow 1 and Arrow 4), the switch sends new flow message to the controller (Arrow 2 and Arrow 5). Once the controller creates the rule and installs it on the network, the controller pushes the rule to the switch (Arrow 3 and Arrow 6). Finally, the packet of new flow is forwarded to end-point E1 (Arrow 7). Reactive rule placement requires to adjust configuration of the network continuously which increases flow path establishment latency [12, 26]. The controller creates path for every new coming flow and reroute for all effected flow.

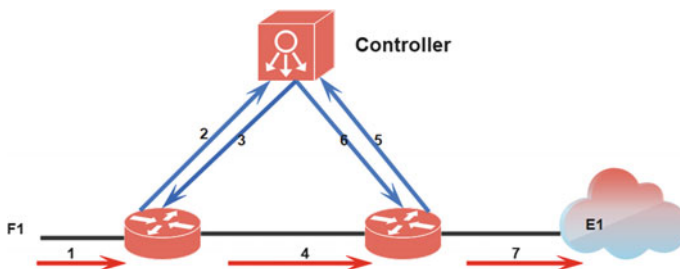


Fig. 3 Reactive rule placement

Hybrid rule placement: Using proactive approach in rule placement requires prior to paths calculation and traffic prediction, while using reactive approach for all flows increases flow path establishment latency and memory constraint. Hybrid approaches [5, 10, 18, 23] are proposed to create new flow entry in a proactive–reactive manner. Proactive handles the flow traffic at aggregate level, and reactive creates flow table entries with higher priorities.

Rule space reduction: Another problem is the routing table of SDN is stored on TCAM which causes power hungry and memory constraints. Rule space reduction is an essential technique to generate, compress and push the rules to forwarding switch to reduce the table size to improve routing traffic while optimising energy consumption. Many existing researches have addressed and proposed rule space reduction approach to solve the problem of rule space limitation. The authors in [9, 13–21, 24, 25] proposed aggregation technique to compress the flow table. Kanan et al. tried to compress by adding tag to packet header [18], and Rifai et al. proposed packet header compression to compact the rule [5, 6].

The authors in [7, 10, 22, 27] proposed forwarding policies distribution as another solution to manage rule space constraints. Unlike aggregation technique, forwarding policies do not apply any compression mechanisms to the flow. For example, Nguyen et al. proposed to deviate traffic path to alternative paths with different policies to reach destination without violating end-point policy [7]. However, the mechanism may introduce penalty on QoS of flows when forwarding table are rarely full. Chuangchuang et al. introduced intelligent routing base on neural network routing scheme. The principle is to reduce TCAM space and to improve data flow routing and forwarding. Flow table is replaced by neural network, and the routing of data flow can be predicted based on application type that meets the QoS requirement of the network.

Wildcard: is also called compression or aggregation technique which is used to reduce the required rules in the flow table. The number of original rules will be compacted and replaced by smaller size of rule of flow table. The principle of wildcard is to compress flows that have the same action into a single rule to reduce rule space while preserving the original semantic. Authors in [9, 22] tried to compress and prioritise access control rules. The access control rule compression has multiple header field and a binary decision action which indicate that a packet is dropped or permitted. Authors in [3, 5, 12, 13] proposed forwarding rule compression to decide exact forwarding rules for the flow on the chosen path. The access control compression is more complicated than forwarding compression; the forwarding compression has stricter time constraints for fast rerouting for the case of failure. Zhou et al. proposed a hybrid of wildcard approach by joining per-flow routing and tag-based routing problem. Even though wildcard rule placement can reduce number of rules to optimise memory constraints, it makes flow become less visible which make the controller cannot control the flow without impacting other flows. In addition, allocating rules with high ratio of compression require more computation time and slow down the network configuration update. Therefore, to design a wildcard, computation ratio and update time must be considered in order to achieve a trade-off solution.

Routing policy optimisation: The ability of rule placement to view the network globally is tackled. Rule placement acquires the information of entire network, routing policy and end-point policy to decide which rule should be placed on forwarding switch. Nguyen et al. [7] and Giroire et al. [9] implemented efficient approach for forwarding rules in the switch, reselecting end-point policy and relaxing routing policy. The idea of relaxing routing policy could achieve efficient use of network and obtain maximum of end-point policy. However, the relaxing routing policy has a drawback of longer paths. On the other hand, Zhao et al. [21] implemented hybrid rule placement for fine-grained flow management by installing wildcard rules and exact match rules. The model of Zhao achieved fine-grained management for all flows comparing to default path scheme model of Nguyen.

Compromised Routing: Routing has the ability to determine and choose shortest path to optimise routing traffic [7, 9, 11, 14, 18, 25]. However, to find alternative energy efficient path may require extra longer paths performance which may affect the QoS of the network. Therefore, to design compromised routing rule placement, impact of path on QoS must be considered.

RQ2: Algorithms used in EAR and rule placement

As the general problem of EAR is known to be NP hard [28], integer linear programs such as greedy-based heuristic algorithm has been the most commonly method applied to solve the problem of energy optimisation in SDN [8–10, 13, 15, 16, 18–21, 23–27]. However, greedy-based heuristics do not guarantee optimal solution for flow routing. Jimenes et al. [17] and Qi et al. [22] proposed better energy consumption in SDN using genetic-based heuristic algorithm, and Chuangchuang et al. [11] proposed neural network-based intelligent routing scheme for SDN using neural network to replace flow table with well-trained neural network. Details of each algorithm are described as follows:

Greedy-based heuristic algorithm: the primary objective of rule placement is to maximise the number of flows in the network while minimising associated cost. Greedy-based heuristic algorithm is started from the whole of network to compute routing feasibility respecting to rule space constraints. Rules are assigned to each flow freely until routing table is full. Rules are then shrink by forming a default port from the most occurring flow. The links with less loaded are removed, installed rules are reduced and space is available for the new rule. With a smaller number of active links allow the network to save the energy.

Genetic-based heuristic algorithm: is used to solve the problem of routing as dynamic shortest path routing [29, 30], broadcast routing [31] and cluster-based routing [32]. It can find the near-optimal solution. In genetic-based heuristic algorithm, the network is mapped to a chromosome which consists of one rule solution. The rule of certain flow is placed into software or hardware switch based on binary value (0 or 1) of certain bit of a chromosome. The genetic heuristic algorithm then evaluates the chromosomes in the population based on arrival curves and the service curves of the

flow. Maximum delay of traffic flow in the fast and slow channel is calculated, and the fitness of chromosome is evaluated.

Neural network-based algorithm: self-learning algorithm to use and predict the route through neural network (NN). In NN algorithm, when the flow of data goes to SDN, the controller extracts the sources, destination, application type, record transmission paths and construct sample set. The sample set is then normalised, and the controller uses the training set to train the NN model. The NN training process is completed until the prediction success rate reaches the present level. When the NN training is completed, the controller creates the NN packet according to the well-trained NN and distributes the NN packet to all the underlying switches. Once the NN packet distributed by the controller is received, the underlying switch constructs a NN according to the structure parameters contained in the NN packet, and the constructed NN is used for route prediction.

RQ3: Rule placement approaches and QoS

For many years, QoS routing has been a major concern in several studies. The main objective of such research is to propose routing algorithms that can provide the best paths to satisfy customer service level agreement (SLA). To help reducing TCAM storage space and meet the QoS, many rules placement approaches [5–9, 11–27] have been done on flow table compression and aggregation such as wildcard and rule space reduction. These approaches could decrease the TCAM storage space and improve routing efficiency to meet the requirement of QoS. However, they cannot solve the problem of flow table expansion and cannot satisfy the requirement of QoS. On the other hand, Chuangchuang introduced neural network-based approach for SDN routing scheme [11] and designing neural network packet to replace flow table. His model could reduce storage space of TCAM, routing time overhead and improve routing efficiency. However, his model is at early stage of investigating neural network packet instead of flow table and need more improvement and verification on its implementation.

4 Further Research

The study has raised several research potentials that could be possible for further research. The methods were used to optimise storage space of TCAM. These could reduce TCAM, routing time overhead and improve routing efficiency. Secondly, the installed rules in TCAM need to be updated in order to adapt with different scenarios of network requirement such network topology, user mobility and changes in policies. Rule space reduction enabling fast push of the rules to forwarding switch to improve routing while optimising energy consumption is one of possible directions.

To guarantee the performance of network, QoS with different handling and capacity allocation to specific flows in the traffic could be done. Therefore, to design

any rule placement approaches and algorithms, there is a must to meet the measurements concern of QoS to ensure the best performance of the network. Unfortunately, most of the studies do not fully comply their approaches and algorithms with QoS measurements. Consequently, the efficiency of routing is reduced. Therefore, a combination of new controller and OpenFlow rules placement would be interesting to optimise the network.

There is only one study which applied neural network algorithm as self-learning algorithm to predict the route in SDN [11]. However, this study was only a preliminary and requires further study to verify the efficiency of neural network model. In addition, applying advance neural network such as deep learning in routing might improve the network performance.

5 Conclusion

SDNs enable energy saving via use of efficient routing in data centre networks, applying flow base forwarding rules. In this study, we provided a review of rule placement for EAR in SDNs. There is a need to focus on methods combining controllers and OpenFlow with rule placement to optimise the network traffic with energy efficiency and QoS. Secondly, rule space reduction enabling fast push of the rules to forwarding switches to improve routing while optimising energy consumption is one of the recurrent research directions. Finally, self-learning algorithms to be used to predict the route with a neural network is determined as a promising research direction.

References

1. Moy J, OSPF Version 2 (1998) RFC 2328 (Standard). <http://www.ietf.org/rfc/rfc2328.txt>
2. Kitchenham B, Brereton OP, Budgen D, Turner M, Bailey J, Linkman S (2009) Systematic literature reviews in software engineering: a systematic literature review. *Inf Softw Technol* 51:7–15
3. Kitchenham B (2004) Procedures for performing systematic review, joint technical report, software engineering group. Department of Computer Science Keele University, United Kingdom and Empirical Software Engineering, National ICT Australia
4. Dybå T, Dingsøy T, Hanssen GK (2007) Applying systematic reviews to diverse study types: an experience report. In: ESEM 2007. International symposium on empirical software engineering and measurement, pp 225–234
5. Rifai M, Huin N, Caillouet C, Giroire F, Moulhierac J, Pacheco DL, Urvoy-Keller G (2017) Minnie: an SDN world with few compressed forwarding rules. *Comput Network* 121:185–207
6. Rifai M, Huin N, Caillouet C, Giroire F, Pacheco DML, Moulhierac J, Urvoy-Keller G (2015) Too many SDN rules? Compress them with MINNIE. In: GLOBECOM
7. Nguyen X-N, Saucez D, Barakat C, Turlitti T (2015) OFFICER: a general optimization framework for OpenFlow rule allocation and endpoint policy enforcement. In: The 34th annual IEEE international conference on computer communications (INFOCOM), Hongkong, China

8. Giroire F, Havet F, Moulrierac J (2016) Compressing two-dimensional routing tables with order. *Electron Notes Discr Math* 52:351–358
9. Giroire F, Moulrierac J, Phan TK (2014) Optimizing rule placement in software-defined networks for energy-aware routing. In: *IEEE Global communications conference*, Austin, TX, pp 2523–2529. <https://doi.org/10.1109/GLOCOM.2014.7037187>
10. Kang N, Liu Z, Rexford J, Walker D (2013) Optimizing the “one big switch” abstraction in software-defined networks. In: *Ninth ACM conference on emerging networking experiments and technologies*. CoNEXT, ACM, New York, NY, USA, pp 13–24
11. Chuangchuang Z, Xingwei W, Fuliang L, Min H (2019) NNIRSS: neural network-based intelligent routing scheme for SDN. *Neural Comput Appl* 31(6):189–6205
12. Hong ETB, Wey CY (2017) An optimized flow management mechanism in OpenFlow network. In: *International conference on information networking (ICOIN)*, Da Nang, pp 143–147. <https://doi.org/10.1109/ICOIN.2017.7899493>
13. Chen Y, Lin Y (2018) Study of rule placement schemes for minimizing TCAM space and effective bandwidth utilization in SDN. In: *6th international conference on future internet of things and cloud workshops (FiCloudW)*, Barcelona, pp 21–27. <https://doi.org/10.1109/W-FiCloud.2018.00010>
14. Li W, Qin Z, Li K, Yin H, Ou L (2019) A novel approach to rule placement in software-defined networks based on OPTree. *IEEE Access* 7:8689–8700. <https://doi.org/10.1109/ACCESS.2018.2889194>
15. Bouzidi EH, Outtagarts A, Langar R (2019) Deep reinforcement learning application for network latency management in software defined networks. In: *IEEE global communications conference (GLOBECOM)*, Waikoloa, HI, USA, pp 1–6. <https://doi.org/10.1109/GLOBECOM38437.2019.9013221>
16. Giroire F, Huin N, Moulrierac J, Phan TK (2018) Energy-aware routing in software-defined network using compression. *Comput J* 61(10):1537–1556. <https://doi.org/10.1093/comjnl/bxy029>. (hal-01920970)
17. Jimenez JG, Polverini M, Cianfrani A (2018) Reducing the reconfiguration cost of flow tables in energy-efficient software-defined networks. *Comput Commun* 128:95–105
18. Kannan PG, Chan MC, Ma RTB, Chang E (2017) Raptor: scalable rule placement over multiple path in software defined networks. In: *IFIP networking conference (IFIP networking) and workshops*, Stockholm, pp 1–9. <https://doi.org/10.23919/IFIPNetworking.2017.8264843>
19. Zhao G, Huang L, Li Z, Xu H (2018) Hybrid routing by joint optimization of per-flow routing and tag-based routing in software-defined networks. *Tsinghua Sci Technol* 23(4):440–452. <https://doi.org/10.26599/TST.2018.9010084>
20. Mimidis-Kentis A, Pilimon A, Soler J, Berger M, Ruepp S (2018) A novel algorithm for flow-rule placement in SDN switches. In: *4th IEEE conference on network softwarization and workshops (NetSoft)*, Montreal, QC, pp 1–9. <https://doi.org/10.1109/NETSOFT.2018.8459979>
21. Zhao G, Xu H, Fan J, Huang L, Qiao C (2020) Achieving fine-grained flow management through hybrid rule placement in SDNs. *IEEE Trans Parallel Distrib Syst* 32(3):728–742. <https://doi.org/10.1109/TPDS.2020.3030630>
22. Qi Q, Wang W, Gong X, Que X (2017) Rules placement with delay guarantee in combined SDN forwarding element. *KSII Trans Internet Inf Syst* 11:2870–2888. <https://doi.org/10.3837/tiis.2017.06.004>
23. Rzepka M, Borylo P, Lason A, Szymanski A (2020) PARD: hybrid proactive and reactive method eliminating flow setup latency in SDN. *J Netw Syst Manage* 28:1547–1574. <https://doi.org/10.1007/s10922-020-09550-z>
24. Leng B, Huang L, Qiao C, Xu H, Wang X (2017) FTRS: a mechanism for reducing flow table entries in software defined networks. *Comput Netw* 122:1–15
25. Huin N (2016) Energy-aware routing in software-defined networks with table compression (using wildcard rules). Ph.D. thesis. INRIA Sophia Antipolis-13S
26. Ashraf U (2017) Rule minimization for traffic evolution in software-defined networks *IEEE Commun Lett* 21(4):793–796

27. Kanizo Y, Hay D, Keslassy I (2013) Palette: distributing tables in software-defined networks. In: IEEE INFOCOM, pp 545–549
28. Giroire F, Mazauric D, Moulhierac J, Onfroy B (2010) Minimizing routing energy consumption: from theoretical to practical results. In: IEEE/ACM GreenCom
29. Ahn CW, Ramakrishna RS (2002) A genetic algorithm for shortest path routing problem and the sizing of populations. *IEEE Trans Evol Comput* 6(6):566–579. <https://doi.org/10.1109/TEVC.2002.804323>
30. Yang S, Cheng H, Wang F (2010) Genetic algorithms with immigrants and memory schemes for dynamic shortest path routing problems in mobile adhoc networks. *Trans Syst Man Cybern Part C* 40(1):52–63. <https://doi.org/10.1109/TSMCC.2009.2023676>
31. Salcedo-Sanz S, Bousoño-Calzon C, Figueiras-Vidal AR (2003) A mixed neural-genetic algorithm for the broadcast scheduling problem. *IEEE Trans Wirel Commun* 2(2):277–283. <https://doi.org/10.1109/TWC.2003.808967>
32. Al-Ghazal M, El-Sayed A, Kelash H (2007) Routing optimization using genetic algorithm in ad hoc networks. In: 2007 IEEE international symposium on signal processing and information technology, pp 497–503. <https://doi.org/10.1109/ISSPIT.2007.4458010>

Strengthening Auto-Feature Engineering of Deep Learning Architecture in Protein–Protein Interaction Prediction



Bhawna Mewara and Soniya Lalwani

Abstract Protein–protein interactions (PPIs) are known to play a crucial role in system biology by predicting the protein function of a given target. Machine learning (ML) could provide valuable information on PPI and its various effects. However, the complexity of the prediction task makes it more challenging and time-consuming. Deep learning (DL) technology is rapidly becoming useful in various applications. It can extract features automatically by analyzing the network itself. This paper presents a deep neural network (DNN) framework that learns features only from the protein primary sequences. It can predict polypeptides (PPIs) using automated features that were learned automatically from the primary sequences. The proposed model is assessed using two PPI datasets and compared the performances with the existing competitive approaches. It can be witnessed from the outstanding performance of the proposed approach that DL architectures are much capable to work smarter to filter out potential information from the raw data without the inclusion of additional feature engineering as in ML.

Keywords Protein–protein interactions · Deep learning · Auto-feature engineering · Deep neural network

Abbreviations

AA	Amino Acid
AC	Auto-Covariance
ACC	Auto-Cross-Covariance
CT	Conjoint Triad
DL	Deep Learning
DNA	Deoxyribonucleic Acid
DNN	Deep Neural Networks

B. Mewara (✉) · S. Lalwani
Department of Computer Science and Engineering, Career Point University, Kota, India
e-mail: mewara.bhawna2203@gmail.com

FN	False Negatives
FP	False Positives
LD	Local Descriptors
LR	Learning Rate
MCD	Multi-scale Continuous and Discontinuous
ML	Machine Learning
MLD	Multi-Scale Local Descriptor
NN	Neural Network
PPI	Protein–Protein Interaction
ReLU	Rectified Linear Unit
RF	Random Forest
RNA	Ribonucleic Acid
SGD	Stochastic Gradient Descent
SVM	Support Vector Machine
TN	True Negatives
TP	True Positives

1 Introduction

Proteins play several vital roles in the living organism with their large and complex structure. Maximum functions in the cell are done by proteins and act as multi-tasker to perform the role of enzymes, antibodies, messenger, and so on. However, to carry out this wide range of functions, they require a partner like DNA, RNA, or even proteins. The process of becoming the partner is termed as interactions and if the protein interacts with other protein by some signaling process, then is said to have protein–protein interactions (PPIs). Protein control and mediate many of the biological activities of the cell by these interactions. For example, muscle contraction is possible due to PPI between active myosin filaments, cell signaling, cellular transport (molecule coming out and going inside the cell using PPI) [1]. So, PPIs play a vital role in many cellular processes. But, the occurrence of any disruption and development of unusual interactions can lead to disease states like Alzheimer’s disease [2, 3] and in Huntington’s disease [4].

Therefore, prior information about PPIs can offer a clear idea to identify drug targets, further biological processes, and new remedies for diseases [5]. Computational approaches [6] have now become an alternate solution of time-consuming and expensive experimental methods [7] for the identification of PPIs. One dominating computational method is ML that is enlightening better coverage in PPIs prediction by fabricating a suitable feature set and selecting an appropriate ML algorithm for classification [8, 9]. The process of scrutiny of relevant features from the input data is called the feature extraction method and plays a vital role in prediction performance. Numerous researches have been done regarding this and some popular works includes AC, ACC [10], CT [11], LD [12], MCD [13], MLD [14], and many more.

Every feature extraction algorithm entails a favorable classifier to properly categorize the interaction or no interaction according to the feature sets obtained. Various classification algorithms have been developed are RF [14, 15], SVM [10] and their derivatives [12, 16], gradient boosting decision trees [17, 18], and ensemble classifier [19].

In recent years, DL technology is booming in many research areas including bioinformatics that has eased human efforts [20]. Specifically, in PPI prediction tasks, various DL architectures have been used to ease the complex and tiresome feature extraction methods. The architecture of DL solely takes the responsibility of extracting the relevant features from the input data. Numerous researches have been published in PPI prediction using DL approaches. Some of the published works have included handcrafted features along with the DL methods [21, 22] while some have proved better prediction performance using the capabilities of DL architecture alone [23–25]. It can be observed that thorough knowledge of this emerging area is essential to take full advantage of its characteristics.

In this paper, an autonomous framework is presented underlying the architecture of DNN (an architecture of DL) for the prediction of PPIs. The main motivation behind this research is to make a deep and clear concept of auto-feature engineering abilities of DL architecture. The highlights of this prediction work using the DNN framework are:

- DNN model is framed in a way to utilize its autonomous property as much as possible in selecting the relevant features.
- Not inclusion of handcrafted features. The only requirement of architecture is the raw data.
- To obtain a better understanding of considered DNN in every perspective like layers, functions, optimizers, etc.
- The proposed approach is evaluated on two different datasets using six performance measures.

The chapter is systematized as follows: Sect. 1 presents a brief idea behind this proposed approach. Section 2 details the proposed work with a possible explanation of every parameter taken into consideration. Section 3 illustrates the datasets; evaluation measures are taken for PPI prediction and demonstrates the results accordingly. Lastly, a conclusion is made with the future aspects intended to fully utilize the DL technologies in near-term researches in Sect. 4.

2 Methodology

This section demonstrates the working of the proposed approach from scratch. The prediction model, i.e., the DNN model, takes the input data in a suitable format which is obviously not expected to be in character form. The numeric form is the ideal input data format for any DL architecture while the raw data in the PPI prediction task considered here are in sequence and string form by nature; therefore, in order

to represent protein sequence in a suitable numerical form, some operations are followed that can appropriately transform the input sequence into a numeric form with equivalent length. It is notable that this process is not considered a feature extraction method, and this simply an alias representation that makes the input to be eligible for the DNN model. After the successful transformation, the obtained sequence is fed to the neural network for training and testing where optimization of network parameters is done for relevant feature extraction and classification task. The subsequent parts of this section present the aforementioned concept in detail.

1. **Data Remodeling Process.** The PPI interaction dataset and respective AA sequence are required for this step. The details of the dataset taken in this paper are described in the ‘Dataset’ section. The interaction dataset is composed of protein pairs ID and their respective status of interaction. The status of interaction is either 1 (interacting pair) or 0 (non-interacting pair). For every protein in a protein pair, there is a signified AA sequence which is obtained from the database. Once the data collection is completed, the process of remodeling the data start. For this, an AA sequence is considered as a sentence, and instead of taking each AA as a distinct unit, a group of contiguous three AAs is deliberated as a word. Innately, the whole task can be considered as a Natural Language Problem (NLP) [27]. Similarly, the entire sequence is converted into multiple groups of three words or it is said triplet or three-gram without replacement as shown in Fig. 1. And if any sequence length is not divisible by three, the first AA is ignored. This scheme is adopted from the inspiration performance by other works like in [28].
2. **Tokenization process.** Each unique triplet obtained from step 1 is assigned a unique integer (token) number in order to make the sequence adaptable by the neural network for further processing. The token ‘0’ is not considered in the tokenization process because it is used for padding purposes to represent a fixed-length sequence vector (in case of a shorter sequence length). In this research work, the length of the sequence vector is taken as 1000 elements. The longer sequences are truncated from the left side, and pre-padding is used in case of the shorter sequences. The resultant fixed-length numerical representation obtained from this process is now ready for feature extraction and prediction procedure by the DNN model.
3. **Development of DNN model.** The motivation behind the selection of this DL architecture for current research work is its emerging success in many similar applications [29, 30] as well as its capability of selecting potential features from the low-level input by digging out the rules of data through objective function. A DNN, in simple words, is a network that is deep in nature, i.e., which has four or more hidden layers along with the input layer and an output layer. The neurons present in each hidden layer are responsible for the computation of the weighted sum of current input and transfer the output to the next level via an activation function. The proposed DNN model is shown in Fig. 2. For the current research work, *ReLU* [31] and *cross-entropy* [32] are used, respectively, as activation function and loss function which helps in accelerating the training

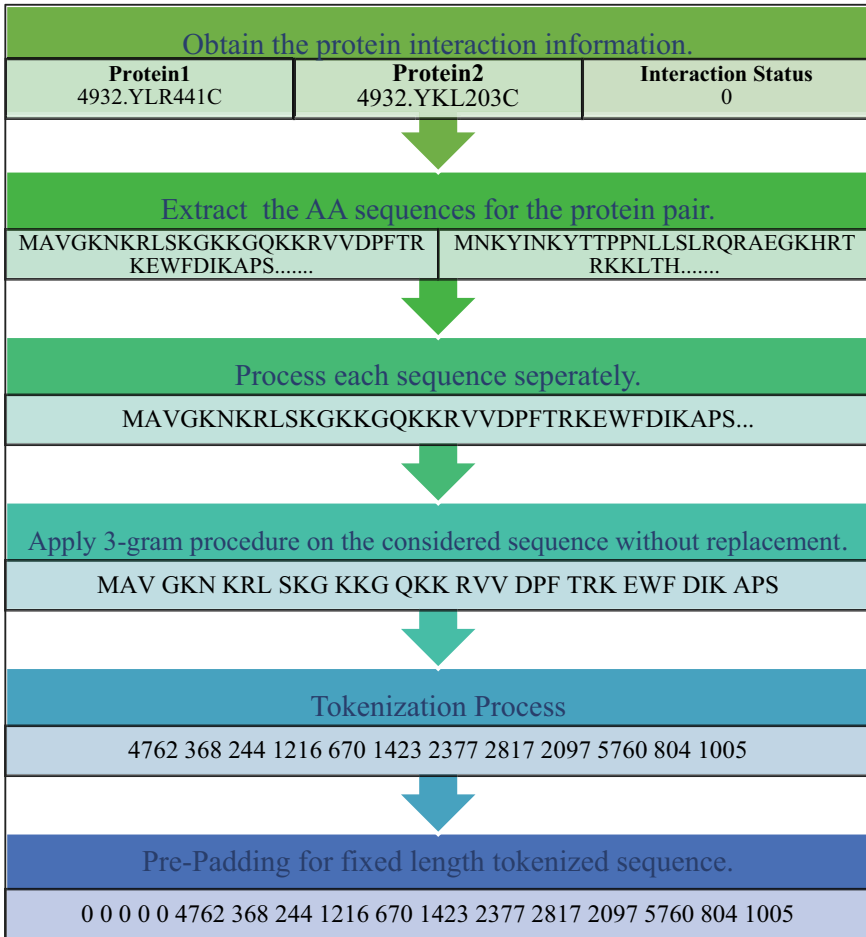
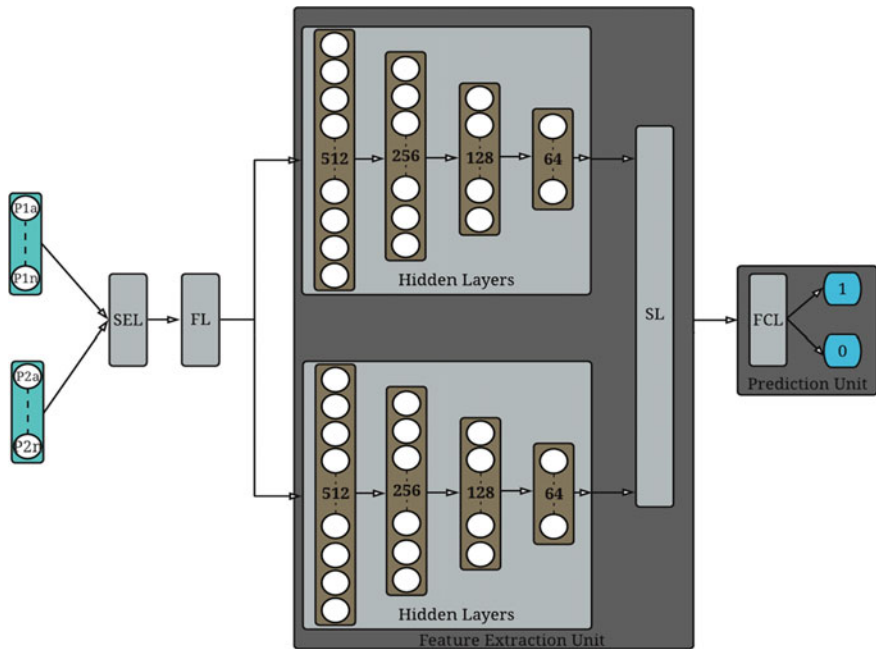


Fig. 1 Overview of data pre-processing from raw data to the fixed-length tokenized sequence representation with example

process. The network parameters used for the proposed work are presented in Table 1.

This can clearly be outlined from Fig. 2, that the proposed DNN model is comprised of multiple segments: input layer, shared embedding layer, feature extraction layer, synthesis layer, and classification layer.

- *Input Layer*: The input of the proposed model is the tokenized protein representation obtained from the previous step. The two separate inputs (each protein in a pair) are taken by the model, and for each protein in a pair, the DNN receives two attributes (one is the data representation and another one is the interaction status) as mentioned above. Therefore, underlying the same



SEL: Shared Embedding layer, FL: Fusion Layer, SL: Synthesis Layer, FCL: Fully Connected Layer

Fig. 2 Architecture of proposed DNN model. P1a–P1n and P2a–P2n are the fixed-length tokenized input sequence

Table 1 Parameter values of DNN model

S. No.	Parameter	Value
1	Batch size	64
2	LR	0.01
3	Momentum value	0.9
4	Dropout rate	20%
5	Epoch size	5
6	Weight decay	0.0001
7	Activation function	ReLU
8	Weight regularization	L2
9	Optimizer	SGD
10	Loss function	Binary cross-entropy

architecture, the proteins are discretely processed in two divisions and learn the complex features from the input.

- *Embedding Layer*: This can be clarified from Fig. 2, that a shared embedding layer is used in the proposed work intended to the contextual dependencies

among the protein pairs. The responsibility of this particular layer is to carefully observe the tokens and outputs the representation based on the occurrences of the tokens in the same sequence. This process is depicted by a toy example in Fig. 2. The output of this layer is another vector representation of the considered input which is a more refined depiction of context in terms of the token information and the positional information. Though this adds another dimension in the output, a *flatten layer* is employed to effectively pass the data into every single neuron of the next layers [33].

- *Feature Extraction Layer*: This layer is tasked to learn the rules of the input context and scrutinize the pertinent configuration from the data. It can be observed from Fig. 2 that this layer consists of two parallelized divisions working on similar network parameters. Each division has four successive *dense* layers having 512, 256, 128, and 64 units, respectively. The term ‘unit’ means the number of neurons present in the particular layer which are responsible for all the complex computation considering the parameter settings and result out the high-level representation as features. The activation function considered here is *ReLU* whose task is to perform the nonlinear operation on the output generated by the *dense* layers. Along with *ReLU*, *dropout* techniques are applied to every layer, in which some neurons and their links are ignored or assigned their values to 0. This is a usual procedure (if opted) [34] to circumvent the over-fitting issue.
- *Synthesis Layer*: As the name indicates, this layer merged the abstract features generated by both divisions.
- *Classification Section*: This section includes the fully connected layers with the *ReLU* function followed by the last layer with the sigmoid function having two units for the prediction results (interaction and no interaction). The loss function considered here is *binary cross-entropy* which takes the interaction probability values evaluated by the sigmoid function and correspondingly determines the interaction status.
- Additionally, at the time of training, the schemes like early stopping [35] and dropping the LR on inactivity are also considered in order to elude wasting resources and to achieve better local minima.

3 Experiment Details

3.1 Datasets

Two different datasets are considered for the evaluation of the proposed approach: Human and *S. cerevisiae*. The protein interaction information is downloaded from the STRING [36] database of version 11.0. For the selection of potential interactions, the positive samples and negative samples for this work are generated using the same approach as in [25]. Their respective AA sequence is obtained from the UniProt database [26].

Table 2 Network dimensions of each layers of DNN model

S. No.	Layer	Output shape of Protein 1	Output shape of Protein 2
1	Input layer	(None, 1000)	(None, 1000)
2	Embedding layer (shared)	(None, 1000, 512)	(None, 1000, 512)
3	Flatten layer	(None, 51,200)	(None, 51,200)
4	DeL1	(None, 256)	(None, 256)
5	DeL2	(None, 128)	(None, 128)
6	DeL3	(None, 64)	(None, 64)
7	Synthesis layer	(None, 128)	
8	Merged_feature1 (FCL)	(None, 64)	
9	Merged_feature2 (FCL)	(None, 32)	
10	Merged_feature3 (FCL)	(None, 16)	
11	DeL4	(None, 2)	

DeL dense layer; *FCL* fully connected layer

3.2 Network Parameters

The overall architecture of the proposed DNN model is explained in the previous sections. The network parameter settings and the output shape of each layer with other essential details are briefly outlined in Tables 1 and 2. The same parameters are used for all the datasets considered in the experiment.

3.3 Result and Discussions

To evaluate the feasibility and effectiveness of the proposed work, fivefold cross-validation is applied on all the datasets considered in this work and legitimately standard evaluation measures [37] are calculated, namely accuracy (Accu.), specificity (Sp), Precision (*P*), Recall (*R*), Matthews's correlation coefficient (MCC), and *F1*-score (*F1*). Then, the results are compared with other existing methods that are introduced in previous researches and are compatible or similar to the proposed model.

Performance of the proposed model on *S. cerevisiae* dataset. The result of PPI prediction using the proposed approach on *S. cerevisiae* dataset is presented in Table 3 and Fig. 3. A-E denotes approaches by proposed model, [23–25, 38] respectively.

The resultant of PPI prediction is also compared with previous works in which the author made use of CNN and LSTM layers [23], GRU [24], Siamese-based

Table 3 Evaluation measures of proposed approach on *S. cerevisiae* dataset

Approach	Accu.	Spc	<i>P</i>	<i>R</i>	<i>F1</i>	MCC
A	99.79	100	99.37	100	99.68	99.53
B	76.61	73.59	75.1	79.63	77.29	53.32
C	92.59	91.59	93.65	91.4	92.51	85.2
D	94.55	0	96.68	92.24	94.41	0
E	97.09	97	97	97.17	97.09	94.17

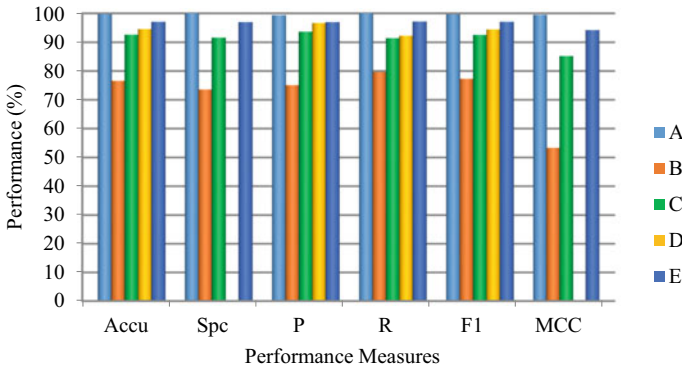


Fig. 3 Performance analysis of proposed approach with existing approaches on *S. cerevisiae* dataset. A–E denotes approaches by proposed model, [23–25, 38] respectively

convolutional NN [25], and convolutional layer with bidirectional GRU [38] as DL approach to predict PPIs as shown in Fig. 4. These competitive works considered are

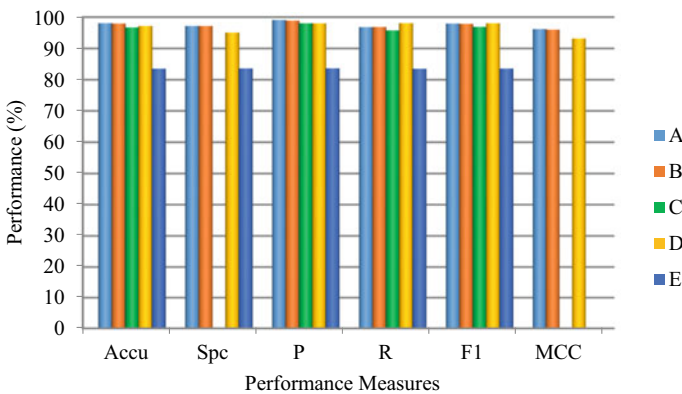


Fig. 4 Performance analysis of proposed approach with existing approaches on human dataset. A–E denotes approaches by proposed model, [24, 39–41] respectively

Table 4 Evaluation measures of proposed approach on human dataset

Approach	Accu.	SpC	<i>P</i>	<i>R</i>	<i>F1</i>	MCC
A	98.08	97.18	99.11	96.8	97.96	96.2
B	97.98	97.17	98.9	96.85	97.86	96
C	96.69	0	98.05	95.76	96.89	0
D	97.2	95.04	97.99	98.07	98.03	93.16
E	83.55	83.61	83.64	83.49	83.56	0

based on the auto-feature engineering approach, and from the table, it is observed that the proposed approach scales up the effectiveness of the DNN model.

Performance of proposed model on Human dataset: The performance of PPI prediction using the proposed approach on the Human dataset is shown in Table 4 and Fig. 4. A-E denotes approaches by proposed model, [24, 39–41] respectively. As shown in Fig. 4, the resultant of PPI prediction is also compared with previous works that used DL architectures such as GRU [24], 2D-Convolution with ResNet model [39], LSTM classifier using multimodal features [40], and stacked auto-encoder [41]. The proposed approach is performing well when compared with the competitive approaches as in Fig. 4.

4 Conclusion and Future Aspects

This research chapter presents an autonomous DNN model for the prediction of PPIs using sequence information only. Though DNNs, which are a type of DL architectures, are known for their auto-feature engineering capability, still there are a lot more to discover because numerous researchers are taking the help of handcrafted features with NN for improving the performance. The objective of the proposed approach is to make a deep and clear concept of auto-feature engineering abilities of DL architecture. The efficacy of the proposed work is evaluated using two different dataset and the obtained prediction performances proved better results than existing approaches.

To prove the generalization of the proposed approach, testing can be done on independent datasets. Moreover, some other similar application may also target via proposed model like identification of interactions of proteins with other possible molecules, prediction of interaction sites, and so on.

References

1. Cui WJ, Gong XJ, Yu H, Zhang XC (2015) Mining topological structures of protein-protein interaction networks for human brain-specific genes. *Genet Mol Res* 14(4):12437–12445
2. Smith MA, Perry G (1996) Alzheimer disease: protein-protein interaction and oxidative stress. *Bol Estud Med Biol* 44(1–4):5–10
3. Thompson TB, Chaggar P, Kuhl E, Gorieli A, Alzheimer's Disease Neuroimaging Initiative (2020) Protein-protein interactions in neurodegenerative diseases: a conspiracy theory. *PLoS Comput Biol* 16(10):e1008267
4. Wanker EE, Ast A, Schindler F, Trepte P, Schnoegl S (2019) The pathobiology of perturbed mutant huntingtin protein–protein interactions in Huntington's disease. *J Neurochem* 151(4):507–519
5. Petta I, Lievens S, Libert C, Tavernier J, De Bosscher K (2016) Modulation of protein–protein interactions for the development of novel therapeutics. *Mol Ther* 24(4):707–718
6. Skrabanek L, Saini HK, Bader GD, Enright AJ (2008) Computational prediction of protein–protein interactions. *Mol Biotechnol* 38(1):1–17
7. Szilagyai A, Grimm V, Arakaki AK, Skolnick J (2005) Prediction of physical protein–protein interactions. *Phys Biol* 2(2):S1
8. Sarkar D, Saha S (2019) Machine-learning techniques for the prediction of protein–protein interactions. *J Biosci* 44(4):1–12
9. Zhang M, Su Q, Lu Y, Zhao M, Niu B (2017) Application of machine learning approaches for protein–protein interactions prediction. *Med Chem* 13(6):506–514
10. Guo Y, Yu L, Wen Z, Li M (2008) Using support vector machine combined with auto covariance to predict protein–protein interactions from protein sequences. *Nucleic Acids Res* 36(9):3025–3030
11. Shen J, Zhang J, Luo X, Zhu W, Yu K, Chen K, Li Y, Jiang H (2007) Predicting protein–protein interactions based only on sequences information. *Proc Natl Acad Sci* 104(11):4337–4341
12. Yang L, Xia JF, Gui J (2010) Prediction of protein-protein interactions from protein sequence using local descriptors. *Protein Pept Lett* 17(9):1085–1090
13. You ZH, Zhu L, Zheng CH, Yu HJ, Deng SP, Ji Z (2014) Prediction of protein-protein interactions from amino acid sequences using a novel multi-scale continuous and discontinuous feature set. *BMC Bioinform* 15(15):1–9
14. You ZH, Chan KC, Hu P (2015) Predicting protein-protein interactions from primary protein sequences using a novel multi-scale local feature representation scheme and the random forest. *PLoS ONE* 10(5):e0125811
15. Ding Y, Tang J, Guo F (2016) Identification of protein–protein interactions via a novel matrix-based sequence representation model with amino acid contact information. *Int J Mol Sci* 17(10):1623
16. Zhou YZ, Gao Y, Zheng YY (2011) Prediction of protein–protein interactions using local description of amino acid sequence. In: *Advances in computer science and education applications*. Springer, Berlin, Heidelberg, pp 254–262
17. Friedman JH (2001) Greedy function approximation: a gradient boosting machine. *Ann Stat*:1189–1232
18. Zhou C, Yu H, Ding Y, Guo F, Gong XJ (2017) Multi-scale encoding of amino acid sequences for predicting protein interactions using gradient boosting decision tree. *PLoS ONE* 12(8):e0181426
19. Wei L, Xing P, Zeng J, Chen J, Su R, Guo F (2017) Improved prediction of protein–protein interactions using novel negative samples, features, and an ensemble classifier. *Artif Intell Med* 83:67–74
20. Min S, Lee B, Yoon S (2017) Deep learning in bioinformatics. *Brief Bioinform* 18(5):851–869
21. Yao Y, Du X, Diao Y, Zhu H (2019) An integration of deep learning with feature embedding for protein–protein interaction prediction. *PeerJ* 7:e7126

22. Mahapatra S, Gupta VRR, Sahu SS, Panda G (2021) Deep neural network and extreme gradient boosting based Hybrid classifier for improved prediction of Protein–Protein interaction. *IEEE/ACM Trans Comput Biol Bioinform*
23. Li H, Gong XJ, Yu H, Zhou C (2018) Deep neural network based predictions of protein interactions using primary sequences. *Molecules* 23(8):1923
24. Gonzalez-Lopez F, Morales-Cordovilla JA, Villegas-Morcillo A, Gomez AM, Sanchez V (2018) End-to-end prediction of protein-protein interaction based on embedding and recurrent neural networks. In: 2018 IEEE international conference on bioinformatics and biomedicine (BIBM). IEEE, pp 2344–2350
25. Hashemifar S, Neyshabur B, Khan AA, Xu J (2018) Predicting protein–protein interactions through sequence-based deep learning. *Bioinformatics* 34(17):i802–i810
26. UniProt: the universal protein knowledgebase. *Nucl Acids Res* 45(D1):D158–D169
27. Young T, Hazarika D, Poria S, Cambria E (2018) Recent trends in deep learning based natural language processing. *IEEE Comput Intell Mag* 13(3):55–75
28. Asgari E, Mofrad MR (2015) Continuous distributed representation of biological sequences for deep proteomics and genomics. *PLoS ONE* 10(11):e0141287
29. Samek W, Montavon G, Lapuschkin S, Anders CJ, Müller KR (2021) Explaining deep neural networks and beyond: a review of methods and applications. *Proc IEEE* 109(3):247–278
30. Song M, Zhao J, Hu Y, Zhang J, Li T (2018) Prediction based execution on deep neural networks. In: 2018 ACM/IEEE 45th annual international symposium on computer architecture (ISCA). IEEE, pp 752–763
31. Eckle K, Schmidt-Hieber J (2019) A comparison of deep networks with ReLU activation function and linear spline-type methods. *Neural Netw* 110:232–242
32. Ruby U, Yendapalli V (2020) Binary cross entropy with deep learning technique for image classification. *Int J Adv Trends Comput Sci Eng* 9(10)
33. Ketkar N (2017) Introduction to Keras. In: Deep learning with python. Apress, Berkeley, CA, pp 97–111
34. Srivastava N (2013) Improving neural networks with dropout. *Univ Toronto* 182(566):7
35. Prechelt L (1998) Early stopping-but when? In: Neural networks: tricks of the trade. Springer, Berlin, Heidelberg, pp 55–69
36. Szklarczyk D, Gable AL, Lyon D, Junge A, Wyder S, Huerta-Cepas J, Simonovic M, Mering CV (2019) STRING v11: protein–protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. *Nucl Acids Res* 47(D1):D607–D613
37. Vihinen M (2012) How to evaluate performance of prediction methods? Measures and their interpretation in variation effect analysis. *BMC Genom* 13(4):1–10
38. Chen M, Ju CJT, Zhou G, Chen X, Zhang T, Chang KW, Zaniolo C, Wang W (2019) Multi-faceted protein–protein interaction prediction based on Siamese residual RCNN. *Bioinformatics* 35(14):i305–i314
39. Lu S, Hong Q, Wang B, Wang H (2020) Efficient ResNet model to predict protein-protein interactions with GPU computing. *IEEE Access* 8:127834–127844
40. Jha K, Saha S (2020) Amalgamation of 3D structure and sequence information for protein–protein interaction prediction. *Sci Rep* 10(1):1–14
41. Jha K, Saha S, Tanveer M (2021). Prediction of protein–protein interactions using stacked auto-encoder. *Trans Emerg Telecommun Technol*:e4256

Author Index

A

Abhishek P. Jiju, 1031
Abirami, A. M., 817
Ajay K. Sharma, 909
Alexander Krutikov, 675
Ali Dehghani, 1083
Amal Pavithran, 1031
Anirudh V. Ragam, 273
Anshu Devi, 1169
Anupama, V., 523
Anupam Kumar, 771
Anusha Gadgil, 139
Anusha, M., 119
Anushka Xavier, K., 223, 371
Arjun Thakur, 139
Arpith G. Naik, 1031
Arshi Naim, 245
Aruna, R., 119
Asha Ambhaikar, 411
Ashwanth, V., 201
Aswin Unnikrishnan, 1031

B

Baibaswata Bhattacharjee, 1
Baraq Ghaleb, 463
Bhavisha, G., 119
Bhawna Mewara, 1205
Bhusan Zope, 1015
Biswajit Kar, 95
Bruno Mendes, 421
Bryan Sandoval-Maiza, 1107
Bui Thi Bich Ngoc, 129

C

Chaitanya Kharche, 1043
Chandana, G., 119
Chandan K. Das, 745
Chetradevee, S. L., 223, 371
Chinmay Gosavi, 883
Chinnadurai, M., 53
Chinnam S. V. Maruthi Rao, 853
Craig Thomson, 463

D

Daniel Yanez-Bravo, 1107
Dário Passos, 421
David Rivas-Lalaleo, 1107
Deepak Mane, 1015, 1043
Deepa S. Kumar, 1053
Devadas Kuna, 829
Devashri Raich, 411
Devika Menon, M. K., 897
Dhanya, P. R., 759
Dipanshu Rautela, 169
Dmitry Strabykin, 675
Durgadevi, M., 393

E

Easwaramoorthy Rangaswamy, 489
Ebrahim Hirani, 319
Esther Rani, P., 27
Eugene Fedorov, 447
Ezzaldden Mahyoub, 179

G

GBS Akhil, 273

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022

H. Sharma et al. (eds.), *Communication and Intelligent Systems*, Lecture Notes in Networks and Systems 461, <https://doi.org/10.1007/978-981-19-2130-8>

Gitanjali Mehta, 557
 Gokula Krishnan, V., 327
 Grazyna Dzwigala, 463
 Gurupandi, D., 327
 Gyanendra Sheoran, 909

H

Hamed Khodadadi, 1083
 Hamid Ghadiri, 1083
 Hanseon Joo, 713
 Hayoung Choi, 713
 Hector Lasluisa-Naranjo, 1107
 Hetal Panchal, 661
 Hoang Trung Hieu, 235
 Hyodong Ha, 713

I

Igor Fedorko, 1067
 Indira, K., 817
 Isam Wadhaj, 463
 Ivana Strumberger, 947

J

Jagadeeswara Rao, E., 433
 Jagadisha, N., 523
 Jaisakthi, S. M., 759
 Jayapandian, N., 223, 371
 Jayavel Kanniappan, 587
 Jithin Gangadharan, 587
 Joseph Rodrigues, 897
 Jules-Raymond Tapamo, 39
 Jyotsna Singh, 383

K

Kalpana Naidu, 637
 Kamil Skarzynski, 973
 Karthiga Shankar, 817
 Karthika, S., 393
 Kazuo Shiokawa, 481
 Ketan Kotecha, 139
 Kishore Kumar Pedapenki, 1181
 Kunju Lekshmi, A. S., 1053
 Kutubuddin Ansari, 481, 725

L

Lakshmi Akshitha, Y., 119
 Lalit Kumar Awasthi, 933
 Lavika Goel, 883
 Leena Ladage, 963

Le Xuan Huy, 129
 Lohitha, B., 119
 L'udovít Nastišin, 685
 Luiz Guerreiro Lopes, 1093
 Luka Jovanovic, 947

M

Madhuri Saha, 989
 Majid Zaman, 619
 Maksim Kovalchuk, 603
 Mallesham, G., 295
 Mamatha, H. R., 273
 Mani Joseph, P., 947
 Mani Kumar Jogi, 873
 Manisha Chaudhary, 789
 Manoj Kumar, 107
 Marcelo Alvarez-Veintimilla, 1107
 Marcin Stepniak, 973
 Meharunnisa, M., 1137
 Mihir Gohad, 139
 Milica Ivanovic, 947
 Minjong Cheon, 713
 Miodrag Zivkovic, 947
 Mohammad Rashid Hussain, 245
 Mohammed Saleh Alsaqer, 245
 Mohammed Tawfik, 179
 Mrunal Mendgudle, 1157
 Mrunal Shidore, 1157
 Muheet Ahmed Butt, 619
 Mukund Prasad Sah, 343
 Mukunthan, B., 79
 Murthy, B. S. N., 1119

N

Naga Raju, M., 1119
 Nagadeepa, N., 79
 Naishadh Mehta, 11
 Naresh Nadipilli, 489
 Nasser M. Al-Zidi, 179, 463
 Naveen Kumar Perumalla, 829
 Nebojsa Bacanin, 947
 Neetu Singla, 383
 Nidarshan Kumar, 273
 Nihar Ranjan, 1015
 Nilesh Kumar, 771
 Nisha Angeline, C. V., 817
 Nishad Nawaz, 489
 Nitai Pal, 989
 Noélia Correia, 421

O

Olga Nechyporenko, 447
 Omer Ozgur Tanriover, 1191
 Ook Lee, 713

P

Palanivelan, M., 533
 Pappu Soundarya Lahari, 807
 Parama Bhaumik, 61
 Pauroosh Kaushal, 573
 Pawan Dubey, 909
 Pepe Ibañez-Jacome, 1107
 Petr Gorbunov, 863
 Pham Ngoc Hai, 235
 Phan Duy Hung, 129, 235
 Philane Tshabalala, 651
 Pooja Dehraj, 283
 Pooja Shah, 11
 Prabhat Kumar, 919
 Pranjal Bahore, 169
 Pranshav Gajjar, 11
 Preeti Kathiria, 545
 Priya Gautam, 261
 Priya, L., 533
 Puniyawi Jamjareegulgarn, 481, 725

R

Rachid Ben Said, 1191
 Rahee Walambe, 139
 Rahul Chaurasiya, 169
 Rajashree Taparia, 261
 Rajesh Kumar Jayavel, 587
 Rajiv Ranjan, 919
 Rajneesh Rani, 343
 Ramabalan, S., 53
 Ramakrishna Akella, 853
 Ramesh Kait, 1169
 Rangith B. Kuriakose, 651
 Ranjeet Bidwe, 1015
 Regidi Suneetha, 735
 Richard Fedorko, 685, 1067
 Ritvik Shrivastava, 283
 Rodmonga Potapova, 863
 Rodrigo Possidônio Noronha, 703
 Rosa Granizo-López, 1107
 Ruqaiya Khanam, 557

S

Saam Prasanth Dheeraj, 27
 Sabeena Beevi, K., 1053
 Sachin Gajjar, 661

Sakirin Tam, 1191
 Salem Alelyani, 245
 Samruddhi Anikhindi, 573
 Sanjay Singh, 843
 Sankar, K., 327
 Saranya, J., 533
 Saravanan, T., 79
 Sariah López-Fierro, 211
 Sathiya, V., 53
 Sayan Das, 95
 Sayani Mondal, 771
 Sekhar R. Aravind, 509
 Sérgio Ribeiro, 1093
 Shailza Kanwar, 933
 Shani du Plessis, 421
 Sharad Garg, 283
 Sheena Christabel Pravin, 533
 Sheikh Amir Fayaz, 619
 Shilpa Mehta, 157
 Shreyansh Paliwal, 169
 Shreyas Patil, 573
 Shrisha, H. S., 523
 Shubhavya, K., 817
 Shweta Bankar, 1043
 Shyamala Devi, M., 119
 Siva Kumar, CH., 295
 Smit Patel, 545
 Sneha Sreedevi, 201
 Soniya Lalwani, 1205
 Sornam, M., 1137
 Sreeni, K. G., 509
 Sreenivasulu Reddy, D., 807
 Sreerama Murthy, K., 327
 Sridevi, P. V., 735
 Srinivasa Rao, Y., 357, 873, 963
 Srinivas, M. A. S., 1119
 Srinivas, M. N., 1119
 Stanislaw Ambroszkiewicz, 973
 Štefan Kráľ, 1067
 Sudeep, P. V., 1031
 Sudhkar, J., 433
 Sunil Nimbhore, 179
 Sunkaraboina Sreenu, 637
 Surajit Bosu, 1
 Suraksha Suryawanshi, 1043
 Surekha Dholay, 319
 Surendaranath, K., 533
 Suresha, D., 523
 Sushama Nagpal, 383
 Svitlana Smerichevska, 447
 Swarna Kamal Paul, 61
 Swati V. Shinde, 1043

T

Talal A. Aldhaheri, [463](#)
Tanu Singh, [107](#)
Tarun Biswas, [771](#)
Tarun Kumar, [343](#)
Tetyana Utkina, [447](#)
Tilak Raju, D., [357](#)

U

Usha Patel, [545](#)

V

Varaprasad Janamala, [807](#)
Varun Magotra, [319](#)
Vasily Meltsov, [675](#)
Vasyl Tereshchenko, [603](#)
Vedant Mehta, [319](#)
Veena Puri, [789](#)
Veni, T., [309](#)
Venkata Rao, K., [327](#)
V́ctor Bautista-Naranjo, [1107](#)
Vijay Kumari, [883](#)
Vijay Ukani, [11](#)
Vikram Singh, [843](#)

Vineeta Kumari, [909](#)
Vinod Kumar Yadav, [557](#)
Virender Ranga, [1169](#)
Vishal, A., [533](#)
Vishal Balaji Sivaraman, [533](#)
Viswanathasarma, Ch., [327](#)
Vivek Shrivastava, [933](#)
Vsevolod Potapov, [863](#)

W

Waldemar Bartyna, [973](#)
Worachai Srisamoodkham, [481, 725](#)

Y

Yadav Maharaj, [39](#)
Yaroslav Tereshchenko, [603](#)
Yashpal Singh, [411](#)
Yashvardhan Sharma, [883](#)
Yuichi Otsuka, [481](#)
Yuliia Remyha, [447](#)

Z

Zeyad A. T. Ahmed, [179](#)