

Lecture Notes in Networks and Systems 336

S. Smys
Valentina Emilia Balas
Ram Palanisamy *Editors*

Inventive Computation and Information Technologies

Proceedings of ICICIT 2021

 Springer

Lecture Notes in Networks and Systems

Volume 336

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.

More information about this series at <https://link.springer.com/bookseries/15179>

S. Smys · Valentina Emilia Balas · Ram Palanisamy
Editors


Inventive Computation and Information Technologies

Proceedings of ICICIT 2021

 Springer

Editors

S. Smys
Department of Computer Science
and Engineering
RVS Technical Campus
Coimbatore, Tamil Nadu, India

Valentina Emilia Balas 
Aurel Vlaicu University of Arad
Arad, Romania

Ram Palanisamy
Business Administration Department
Gerald Schwartz School of Business
St. Francis Xavier University
Antigonish, NS, Canada

ISSN 2367-3370

ISSN 2367-3389 (electronic)

Lecture Notes in Networks and Systems

ISBN 978-981-16-6722-0

ISBN 978-981-16-6723-7 (eBook)

<https://doi.org/10.1007/978-981-16-6723-7>

© 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

ICICIT is highly privileged to dedicate this book to the young and aspiring researchers in the “Computation and Informatics” domain. We also dedicate this book to all the reviewers, authors, and committee members who contributed actively during the whole conference program.

Preface

This conference proceedings volume contains the written versions of most of the contributions presented during the conference of ICICIT 2021. The conference provided a setting for discussing recent developments in a wide variety of topics including cloud computing, artificial intelligence, and fuzzy neural systems. The conference has been a good opportunity for participants coming from various destinations to present and discuss topics in their respective research areas.

This conference tends to collect the latest research results and applications on computation technology, information and control engineering. It includes a selection of 67 papers from 232 papers submitted to the conference from universities and industries all over the world. All the accepted papers were subjected to strict peer reviewing by 2–4 expert referees. The papers have been selected for this volume because of quality and the relevance to the conference.

We would like to express our sincere appreciation to all the authors for their contributions to this book. We would like to extend our thanks to all the referees for their constructive comments on all papers; especially, we would like to thank to organizing committee for their hard working. Finally, we would like to thank Springer publications for producing this volume.

Coimbatore, India
Arad, Romania
Antigonish, Canada

S. Smys
Valentina Emilia Balas
Ram Palanisamy

Acknowledgements

ICICIT 2021 would like to acknowledge the excellent work of our conference organizing the committee and keynote speakers for their presentation on August 12–13, 2021. The organizers also wish to acknowledge publicly the valuable services provided by the reviewers.

On behalf of the editors, organizers, authors, and readers of this conference, we wish to thank the keynote speakers and the reviewers for their time, hard work, and dedication to this conference. The organizers wish to acknowledge Dr. Y. Robinson, Dr. S. Smys, Dr. Valentina Emilia Balas, and Dr. Ram Palanisamy for the discussion, suggestion, and cooperation to organize the keynote speakers of this conference. The organizers also wish to acknowledge for keynote speaker Dr. Shajulin Benedict, Indian Institute of Information Technology, Kottayam, India, and participants who attend this conference. Many thanks are given for all persons who help and support this conference. We would like to acknowledge the contribution made to the organization by its many volunteers. The members contribute their time, energy, and knowledge at a local, regional, and international level.

We also thank all the chair persons and conference committee members for their support.

Contents

Sign Language Recognition: A Comparative Analysis of Deep Learning Models	1
Aswathi Premkumar, R. Hridya Krishna, Nikita Chanalya, C. Meghadev, Utkrist Arvind Varma, T. Anjali, and S. Siji Rani	
Hardware Trojan Detection at Behavioral Level Using Inline Assertions and Verification Using UVM	15
Aki Vamsi Krishna and E. Prabhu	
A Tool to Extract Onion Links from Tor Hidden Services and Identify Illegal Activities	29
Varun Nair and Jinesh M. Kannimoola	
Adaptive IoT System for Precision Agriculture	39
V. Geetha Lekshmy, P. A. Vishnu, and P. S. Harikrishnan	
Web Design Focusing on Users Viewing Experience with Respect to Static and Dynamic Nature of Web Sites	51
R. M. Balajee, M. K. Jayanthi Kannan, and V. Murali Mohan	
Image-Based Authentication Security Improvement by Randomized Selection Approach	61
R. M. Balajee, M. K. Jayanthi Kannan, and V. Murali Mohan	
Automatic Content Creation Mechanism and Rearranging Technique to Improve Cloud Storage Space	73
R. M. Balajee, M. K. Jayanthi Kannan, and V. Murali Mohan	
Voter ID Card and Fingerprint-Based E-voting System	89
Rajesh Kannan Megalingam, Gaurav Rudravaram, Vijay Kumar Devisetty, Deepika Asandi, Sai Smaran Kotaprolu, and Vamsy Vivek Gedela	

Intelligent CCTV Footage Analysis with Sound Source Separation, Object Detection and Super Resolution 107
Yash Khare, Abhijit Ramesh, Vishwaak Chandran,
Sevagen Veerasamy, Pranjali Singh, S. Adarsh, and T. Anjali

A Real-Time Approach of Fall Detection and Rehabilitation in Elders Using Kinect Xbox 360 and Supervised Machine Learning Algorithm 119
V. Muralidharan and V. Vijayalakshmi

A New Approach for Optical Image Encryption Standard Using Bit Swapping and Fractional Fourier Transform 139
L. Anusree and M. Abdul Rahiman

Ensemble Model Ransomware Classification: A Static Analysis-based Approach 153
Shanoop Johnson, R. Gowtham, and Anand R. Nair

Flood Prediction Using Hybrid ANFIS-ACO Model: A Case Study 169
Ankita Agnihotri, Abinash Sahoo, and Manoj Kumar Diwakar

Evaluation of Different Variable Selection Approaches with Naive Bayes to Improve the Customer Behavior Prediction 181
R. Siva Subramanian, D. Prabha, J. Aswini, and B. Maheswari

Personalized Abstract Review Summarization Using Personalized Key Information-Guided Network 203
Nidhin S. Dharan and R. Gowtham

PKI-Based Security Enhancement for IoT in 5G Networks 217
Nayeem Ahmad Khan

Wearable Tag for Human Health Monitoring System 227
A. Jhansi Sri Latha, Ch. NagaSai Manojna,
Ch. N. L. Padma Ashalesha, and K. S. Balamurugan

Energy Efficient Advancement-Based Dive and Rise Localization for Underwater Acoustic Sensor Networks 241
R. Bhairavi and Gnanou Florence Sudha

Performance Comparison of Machine Learning Algorithms in Identifying Dry and Wet Spells of Indian Monsoon 257
Harikumar Rajaguru and S. R. Sannasi Chakravarthy

Automated Hardware Recon—A Novel Approach to Hardware Reconnaissance Process 267
Kalpesh Gupta, Aathira Dineshan, Amrita Nair, Jishnu Ganesh,
T. Anjali, Padmamala Sriram, and J. Harikrishnan

Randomised Analysis of Backtracking-based Search Algorithms in Elucidating Sudoku Puzzles Using a Dual Serial/Parallel Approach 281
 Pramika Garg, Avish Jha, and Kumar A. Shukla

High Speed VLSI Architecture Design Using FFT for 5G Communications 297
 P. Lakshmi Devi, Somashekhar Malipatil, and P. S. Surekha

A Literature Review on Bidirectional Encoder Representations from Transformers 305
 S. Shreyashree, Pramod Sunagar, S. Rajarajeswari, and Anita Kanavalli

Localization and Multi-label Classification of Thoracic Diseases Using Deep Learning 321
 Atique Siddiqui, Sudhanshu Chavan, Sana Fatima Ansari, and Prasenjit Bhavathankar

Experimental Evaluation of Adder Circuits on IBM QX Hardware 333
 Divyanshu Singh, Simran Jakhodia, and Babita Jajodia

Development of the InBan_CIDO Ontology by Reusing the Concepts Along with Detecting Overlapping Information 349
 Archana Patel and Narayan C. Debnath

Prevention of Phishing Attacks Using QR Code Safe Authentication 361
 M. Taraka Rama Mokshagna Teja and K. Praveen

Machine Learning Approach to Recognize and Classify Indian Sign Language 373
 Smriti Pillai, Adithya Anand, M. Sai Jishnu, Siddarth Ganesh, and S. Thara

Comparison of Concurrent Program Behavior Using Java Interactive Visualization Environment 383
 M. Shobitha, R. Prakash Sidharth, P. K. Sreesruthi, P. Varun Raj, and Jayaraman Swaminathan

Message Forwarding Scheme with Max-Delivery and Min-Delay for Delay Tolerant Network 395
 Sudhakar Pandey, Nidhi Sonkar, Sanjay Kumar, Danda Pravija, and Sanchit Mahto

Design and Development of Access Control and Face Mask Detector in Real Time Using Deep Learning to Prevent COVID-19 403
 Manu Gupta, Gadhiraju Hari Priya, Nandikonda Archana Reddy, and A. Sanjana

Hierarchical Language Modeling for Dense Video Captioning 421
 Jaivik Dave and S. Padmavathi

Resource Provisioning in Fog-Based IoT 433
Daneshwari I. Hatti and Ashok V. Sutagundar

DOMAIN-Based Intelligent Network Intrusion Detection System 449
Nithil Jose and J. Govindarajan

Movie Recommendation System Using Color Psychology Based on Emotions 463
G. R. Ramya and Priyadarshini Bhatnagar

Global Positioning System (GPS) and Internet of Things (IOT) Based Vehicle Tracking System 481
V. Baby Shalini

Machine Learning-Driven Algorithms for Network Anomaly Detection 493
Md. Sirajul Islam, Mohammad Abdur Rouf, A. H. M. Shahariar Parvez, and Prajoy Podder

Performance Analysis of a FSO Link Considering Different Atmospheric Turbulence 509
Md. Rawshan Habib, Ahmed Yousuf Suhan, Abhishek Vadher, K. M. Monzur Rahaman, A. M. Rubayet Hossain, Md. Rashedul Arefin, Md Shahnewaz Tanvir, and Shuva Dasgupta Avi

Sentiment Analysis of Unstructured Data Using Spark for Predicting Stock Market Price Movement 521
Miss Dhara N. Darji, Satyen M. Parikh, and Hiral R. Patel

Intrusion Detection and Prevention Using RNN in WSN 531
Ashok Yadav and Arun Kumar

Error Evaluation of Short-Term Wind Power Forecasting Models 541
Upma Singh and M. Rizwan

Vision-Based Personal Face Emotional Recognition Approach Using Machine Learning and Tree-Based Classifier 561
R. Sathya, R. Manivannan, and K. Vaidehi

Design and Development of Smart Charger for Automotive Application 575
K. Vinutha, A. Usha, and Poonthugilan Jayaraman

Scalability Challenges and Solutions in Blockchain Technology 595
K. Harshini Poojaa and S. Ganesh Kumar

An Open-Source Framework Unifying Stream and Batch Processing 607
Kiran Deshpande and Madhuri Rao

Smart Mirror Information System Using Iot 631
 B. Praveena, K. R. Chairma Lakshmi, S. Vijayalakshmi,
 and K. Vijay Anand

**A Hybrid Model for Prediction and Progression of COVID-19
 Using Clinical Text Data and Chest X-rays** 641
 Swetha V. Devan and K. S. Lakshmi

**High-Precision Indoor Tracking Using Ultra-Wide Band Devices
 and Open Standards** 655
 K. Deepika and B. Renuka Prasad

Design and Analysis of Single-Phase Inverter for Avionics System 673
 K. Lavenya and M. Umavathi

**IoT-Based Novel Framework for Solid Waste Management
 in Smart Cities** 687
 Mohd Anjum, M. Sarosh Umar, and Sana Shahab

**A Novel Algorithm to Withstand Attacks on Blockchain Using
 Transaction History and Block Publishing Time** 701
 Anjaneyulu Endurthi, Aparna Pyarapu, Gayathri Jagiri,
 and SaiSuma Vennam

Mental Health Prediction Using Data Mining 711
 I. G. Hemanandhini and C. Padmavathy

**Decision Rules Generation Using Decision Tree Classifier
 and Their Optimization for Anemia Classification** 721
 Rajan Vohra, Anil Kumar Dudyala, Jankisharan Pahareeya,
 and Abir Hussain

Design of Low Power Sequential Circuits Using GDI Cells 739
 Sujatha Hiremath and Deepali Koppad

Automatic Drainage Monitoring and Alert System Using IoT 747
 K. R. Chairma Lakshmi, B. Praveena, K. Vijayanand,
 and S. Vijayalakshmi

**Earliest Deadline First (EDF) Algorithm Based Vaccination
 Management System** 759
 M. Karthigha, R. Pavithra, and C. Padmavathy

**Using Computer Vision to Detect Violation of Social Distancing
 in Queues** 771
 Muhammed Ismail, T. Najeeb, N. S. Anzar, A. Aditya, and B. R. Poorna

**An Efficient Approach Toward Security of Web Application Using
 SQL Attack Detection and Prevention Technique** 781
 Vishal Bharati and Arun Kumar

String Matching Algorithm Based Filter for Preventing SQL Injection and XSS Attacks 793
Abhishek Kumar Yadav and Arun Kumar

Modelling the Inhibitors of Online Learning Over 4G Networks: ISM-MICMAC and FMICMAC Analysis 809
L. Kala, T. A. Shahul Hameed, and V. R. Pramod

An Improved Model for Clarification of Geospatial Information 827
Khudov Hennadii, Butko Igor, Makoveichuk Oleksandr, Khizhnyak Irina, Khudov Vladyslav, Yuzova Iryna, and Solomonenko Yuriy

Face Recognition in Different Light Conditions 839
Waseem Rana, Ravikant Pandey, and Jaspreet Kaur

Music Genre Transfer Using TransGAN 851
Sandeep Kumar, Jerin Verghese, and Ritesh Dutta

Diskless Booting: A Hybrid Computing Technology 859
S. Suriya Prasath, Shwetha Ravikumar, and Anindita Khade

Analysis of Deep Learning Models for Early Action Prediction Using LSTM 879
D. Manju, M. Seetha, and P. Sannulal

Identification of Masked Face Using Deep Learning Techniques 889
M. Sheshikala, P. Praveen, and B. Swathi

An Intrusion Detection Approach for Small-Sized Networks 899
Phong Cao Nguyen, Van The Ho, Dong Hai Duong, Thinh Truong Nguyen, Luan Anh Luong, Huong Hoang Luong, and Hai Thanh Nguyen

Author Index 915

Editors and Contributors

About the Editors

Dr. S. Smys received his M.E. and Ph.D. degrees all in Wireless Communication and Networking from Anna University and Karunya University, India. His main area of research activity is localization and routing architecture in wireless networks. He serves as Associate Editor of Computers and Electrical Engineering (C&EE) Journal, Elsevier and Guest Editor of *MONET* Journal, Springer. He is served as a reviewer for IET, Springer, Inderscience and Elsevier journals. He has published many research articles in refereed journals and IEEE conferences. He has been the General chair, Session Chair, TPC Chair and Panelist in several conferences. He is member of IEEE and senior member of IACSIT wireless research group. He has been serving as Organizing Chair and Program Chair of several International conferences, and in the Program Committees of several International conferences. Currently he is working as a professor in the Department of CSE at RVS Technical Campus, Coimbatore, India.

Dr. Valentina Emilia Balas is currently Full Professor at “Aurel Vlaicu” University of Arad, Romania. She is author of more than 300 research papers. Her research interests are in Intelligent Systems, Fuzzy Control, Soft Computing. She is Editor-in-Chief to *International Journal of Advanced Intelligence Paradigms* (IJAIP) and to *IJCSE*. Dr. Balas is member of EUSFLAT, ACM and a SM IEEE, member in TC—EC and TC-FS (IEEE CIS), TC—SC (IEEE SMCS), Joint Secretary FIM.

Prof. Ram Palanisamy is a Professor of Enterprise Systems in the Business Administration Department at the Gerald Schwartz School of Business, St. Francis Xavier University. Dr. Palanisamy teaches courses on Foundations of Business Information Technology, Enterprise Systems using SAP, Systems Analysis and Design, SAP Implementation, Database Management Systems, and Electronic Business (Mobile Commerce). Before joining StFX, he taught courses in Management at the Wayne

State University (Detroit, USA), Universiti Telekom (Malaysia) and National Institute of Technology (NITT), Deemed University, India. His research interest Enterprise Systems (ES) implementation; ES acquisition; ES Flexibility, ES Success; knowledge management systems; Healthcare Inter-professional Collaboration.

Contributors

M. Abdul Rahiman LBSCST, Thiruvananthapuram, Kerala, India

S. Adarsh Department of Computer Science and Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

A. Aditya Mar Baselios College of Engineering and Technology, Thiruvananthapuram, Kerala, India

Ankita Agnihotri Department of Civil Engineering, MNIT Jaipur, Jaipur, Rajasthan, India

Adithya Anand Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

T. Anjali Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

Mohd Anjum Department of Computer Engineering, Aligarh Muslim University, Aligarh, India

L. Anusree LBSITW, Thiruvananthapuram, Kerala, India

N. S. Anzar Mar Baselios College of Engineering and Technology, Thiruvananthapuram, Kerala, India

Deepika Asandi Department of Electronics and Communication Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

J. Aswini Sree Vidyanikethan Engineering College, Tirupati, India

Shuva Dasgupta Avi Ahsanullah University of Science & Technology, Dhaka, Bangladesh

V. Baby Shalini Department of Information Technology, Kalasalingam Academy of Research and Education, Krishnankoil, India

R. M. Balajee Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur, Andhra Pradesh, India

K. S. Balamurugan Department of Electronics and Communication Engineering, Sasi Institute of Technology and Engineering, Tadepalligudem, Andhra Pradesh, India

R. Bhairavi Department of Electronics and Communication Engineering, Pondicherry Engineering College, Puducherry, India

Vishal Bharati Centre for Advanced Studies, AKTU, Lucknow, India

Priyadarshini Bhatnagar Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India

Prasenjit Bhavathankar Bharatiya Vidya Bhavan's Sardar Patel Institute of Technology, Mumbai, India

K. R. Chairma Lakshmi Department of Electronics and Instrumentation Engineering, R.M.K. Engineering College, Gummidipoondi, Chennai, India

Nikita Chanalya Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

Vishwaak Chandran Department of Computer Science and Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

Sudhanshu Chavan Bharatiya Vidya Bhavan's Sardar Patel Institute of Technology, Mumbai, India

Miss Dhara N. Darji DCS, Ganpat University, Mehsana, India

Jaiwik Dave Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India

Narayan C. Debnath Department of Software Engineering, School of Computing and Information Technology, Eastern International University, Binh Duong, Vietnam

K. Deepika Department of MCA, RV College of Engineering, Bengaluru, Karnataka, India

Kiran Deshpande Thadomal Shahani Engineering College, University of Mumbai, Mumbai, Maharashtra, India

Swetha V. Devan Department of Information Technology, Rajagiri School of Engineering and Technology, Ernakulam, Kerala, India

Vijay Kumar Devisetty Department of Electronics and Communication Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

Nidhin S. Dharan Department of Computer Science Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India

Aathira Dineshan Department of Computer Science and Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

Manoj Kumar Diwakar Department of Civil Engineering, MNIT Jaipur, Jaipur, Rajasthan, India

Anil Kumar Dudyala Department of Computer Science, National Institute of Technology Patna (NIT Patna), Patna, India

Dong Hai Duong FPT University, Can Tho, Vietnam

Ritesh Dutta Department of Computer Science, Maharaja Surajmal Institute of Technology, New Delhi, India

Anjaneyulu Endurthi Department of Computer Science and Engineering, Rajiv Gandhi University of Knowledge Technologies—IIIT, Basar, Telangana, India

Sana Fatima Ansari Bharatiya Vidya Bhavan's Sardar Patel Institute of Technology, Mumbai, India

Jishnu Ganesh Department of Computer Science and Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

Siddarth Ganesh Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

S. Ganesh Kumar SRM Institute of Science and Technology, Chennai, India

Pramika Garg SCOPE, Vellore Institute of Technology, Vellore, TN, India

Vamsy Vivek Gedela Department of EE, University of Cincinnati, Cincinnati, OH, USA

V. Geetha Lekshmy Department of Computer Science and Applications, Amrita Vishwa Vidyapeetham, Amritapuri, India

J. Govindarajan Department of Computer Science Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India

R. Gowtham Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India

Kalpesh Gupta Department of Computer Science and Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

Manu Gupta Department of ECM, Sreenidhi Institute of Science and Technology, Hyderabad, India

T. A. Shahul Hameed Thangal Kunju Musaliar College of Engineering, APJ Abdul Kalam Technological University, Kerala, India

J. Harikrishnan Cisco Systems, Bangalore, India

P. S. Harikrishnan Department of Computer Science and Applications, Amrita Vishwa Vidyapeetham, Amritapuri, India

K. Harshini Poojaa SRM Institute of Science and Technology, Chennai, India

Daneshwari I. Hatti Department of Electronics and Communication, BEC, Bagalkot, BLDEA's V.P. Dr. P.G.H. College of Engineering and Technology, Vijayapur, Karnataka, India

I. G. Hemanandhini Department of Computer Science and Engineering, Sri Ramakrishna Engineering College, Coimbatore, India

Khudov Hennadii Ivan Kozhedub Kharkiv National Air Force University, Kharkiv, Ukraine

Sujatha Hiremath Department of Electronics and Communication Engineering, RVCE, Bengaluru, India

Van The Ho FPT University, Can Tho, Vietnam

R. Hridya Krishna Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

Abir Hussain Department of Computer Science, Liverpool John Moores University, Liverpool, UK

Butko Igor State Enterprise “State Land Cadastral Center”, Kyiv, Ukraine

Khizhnyak Irina Ivan Kozhedub Kharkiv National Air Force University, Kharkiv, Ukraine

Yuzova Iryna Ivan Kozhedub Kharkiv National Air Force University, Kharkiv, Ukraine

Muhammed Ismail Mar Baselios College of Engineering and Technology, Thiruvananthapuram, Kerala, India

Gayathri Jagiri Department of Computer Science and Engineering, Rajiv Gandhi University of Knowledge Technologies—IIIT, Basar, Telangana, India

Babita Jajodia Indian Institute of Information Technology Guwahati, Guwahati, India

Simran Jakhodia Indian Institute of Information Technology Guwahati, Guwahati, India

M. K. Jayanthi Kannan Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur, Andhra Pradesh, India;

Department of Computer Science and Engineering, JAIN (Deemed to be University), Bangalore, India

Poonthugilan Jayaraman CMS Department, Molex India Business Services Pvt Ltd., Bengaluru, India

Avish Jha SCOPE, Vellore Institute of Technology, Vellore, TN, India

A. Jhansi Sri Latha Department of Electronics and Communication Engineering, Sasi Institute of Technology and Engineering, Tadepalligudem, Andhra Pradesh, India

Shanoop Johnson Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India

Nithil Jose Department of Computer Science Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India

L. Kala NSS College of Engineering, APJ Abdul Kalam Technological University, Kerala, India

Anita Kanavalli Department of Computer Science and Engineering, M S Ramaiah Institute of Technology (Affiliated To VTU), Bengaluru, India

Jinesh M. Kannimoola Department of Computer Science and Applications, Amrita Vishwa Vidyapeetham, Amritapuri, India

M. Karthiga Sri Ramakrishna Engineering College, Coimbatore, India

Jaspreet Kaur Department of Computer Science and Engineering, JSS Academy of Technical Education, Noida, India

Anindita Khade SIES GST, University of Mumbai, Navi Mumbai, India

Nayeem Ahmad Khan Faculty of Computer Science and Information Technology, AlBaha University, AlBaha, Saudi Arabia

Yash Khare Department of Computer Science and Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

Deepali Koppad Department of Electronics and Communication Engineering, RIT, Bengaluru, India

Sai Smaran Kotaprolu Department of Electronics and Communication Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

Aki Vamsi Krishna Department of Electronics and Communication Engineering, Amrita School of Engineering, Coimbatore, Amrita Vishwa Vidyapeetham, Coimbatore, India

Arun Kumar Department of Computer Science and Engineering, Centre for Advanced Studies, Dr. A.P.J. Abdul Kalam Technical University, Lucknow, Uttar Pradesh, India

Sandeep Kumar Department of Computer Science, Maharaja Surajmal Institute of Technology, New Delhi, India

Sanjay Kumar Department of Information Technology, National Institute of Technology Raipur, Raipur, India

K. S. Lakshmi Department of Information Technology, Rajagiri School of Engineering and Technology, Ernakulam, Kerala, India

P. Lakshmi Devi Department of Electronics and Communication Engineering, St. Peeter's Engineering College (A), Hyderabad, Telangana, India

K. Lavenya Department of Electrical and Electronics Engineering, B.M.S. College of Engineering, Bengaluru, India

Huong Hoang Luong FPT University, Can Tho, Vietnam

Luan Anh Luong FPT University, Can Tho, Vietnam

B. Maheswari Department of Computer Science and Engineering, Rajalakshmi Institute of Technology, Chennai, India

Sanchit Mahto Department of Information Technology, National Institute of Technology Raipur, Raipur, India

Somashekhar Malipatil Department of Electronics and Communication Engineering, Malla Reddy Engineering College and Management Sciences, Medchal, Telangana, India

R. Manivannan Department of Computer Science and Engineering, SCETW (Autonomous), Hyderabad, India

D. Manju G. Narayanamma Institute of Technology and Science, Hyderabad, India

Rajesh Kannan Megalingam Department of Electronics and Communication Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

C. Meghadev Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

K. M. Monzur Rahaman United International University, Dhaka, Bangladesh

V. Murali Mohan Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur, Andhra Pradesh, India

V. Muralidharan Research Scholar, Department of Computer Science, Government Arts College (Grade-I), (Affiliated to Bharathidasan University), Ariyalur, India

Ch. NagaSai Manojna Department of Electronics and Communication Engineering, Sasi Institute of Technology and Engineering, Tadepalligudem, Andhra Pradesh, India

Amrita Nair Department of Computer Science and Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

Anand R. Nair Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India

Varun Nair Department of Computer Science and Applications, Amrita Vishwa Vidyapeetham, Amritapuri, India

T. Najeeb Mar Baselios College of Engineering and Technology, Thiruvananthapuram, Kerala, India

Hai Thanh Nguyen Can Tho University, Can Tho, Vietnam

Phong Cao Nguyen FPT University, Can Tho, Vietnam

Thinh Truong Nguyen FPT University, Can Tho, Vietnam

Makoveichuk Oleksandr Kharkiv National University of Radio Electronics, Kharkiv, Ukraine

Ch. N. L. Padma Ashalesha Department of Electronics and Communication Engineering, Sasi Institute of Technology and Engineering, Tadepalligudem, Andhra Pradesh, India

S. Padmavathi Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India

C. Padmavathy Department of Computer Science and Engineering, Sri Ramakrishna Engineering College, Coimbatore, India

Jankisharan Pahareeya Department of Information Technology, Rustamji Institute of Technology, BSF Academy, Tekanpur, Gwalior, India

Ravikant Pandey Department of Computer Science and Engineering, JSS Academy of Technical Education, Noida, India

Sudhakar Pandey Department of Information Technology, National Institute of Technology Raipur, Raipur, India

Satyen M. Parikh FCA Ganpat University, Mehsana, India

Archana Patel Department of Software Engineering, School of Computing and Information Technology, Eastern International University, Binh Duong, Vietnam

Hiral R. Patel DCS, Ganpat University, Mehsana, India

R. Pavithra Sri Ramakrishna Engineering College, Coimbatore, India

Smriti Pillai Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

Prajoy Podder Institute of Information and Communication Technology (IICT), Bangladesh University of Engineering and Technology (BUET), Dhaka, Bangladesh

B. R. Poorna Mar Baselios College of Engineering and Technology, Thiruvananthapuram, Kerala, India

D. Prabha Department of Computer Science and Engineering, Sri Krishna College of Engineering and Technology, Coimbatore, India

E. Prabhu Department of Electronics and Communication Engineering, Amrita School of Engineering, Coimbatore, Amrita Vishwa Vidyapeetham, Coimbatore, India

V. R. Pramod NSS College of Engineering, APJ Abdul Kalam Technological University, Kerala, India

K. Praveen TIFAC-CORE in Cyber Security Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India

P. Praveen School of CS and AI, Department of CS and AI, SR University, Warangal, India

B. Praveena Department of Electronics and Instrumentation Engineering, R.M.K. Engineering College, Gummidipoondi, Chennai, India

Danda Pravija Department of Information Technology, National Institute of Technology Raipur, Raipur, India

Aswathi Premkumar Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

Gadhiraju Hari Priya Department of ECM, Sreenidhi Institute of Science and Technology, Hyderabad, India

Aparna Pyarapu Department of Computer Science and Engineering, Rajiv Gandhi University of Knowledge Technologies—IIIT, Basar, Telangana, India

Harikumar Rajaguru Department of Electronics and Communication Engineering, Bannari Amman Institute of Technology, Sathyamangalam, India

S. Rajarajeswari Department of Computer Science and Engineering, M S Ramaiah Institute of Technology (Affiliated To VTU), Bengaluru, India

Abhijit Ramesh Department of Computer Science and Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

G. R. Ramya Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India

Waseem Rana Department of Computer Science and Engineering, JSS Academy of Technical Education, Noida, India

Madhuri Rao Thadomal Shahani Engineering College, University of Mumbai, Mumbai, Maharashtra, India

Md. Rashedul Arefin Ahsanullah University of Science & Technology, Dhaka, Bangladesh

Shwetha Ravikumar SIES GST, University of Mumbai, Navi Mumbai, India

Md. Rawshan Habib Murdoch University, Murdoch, Australia

Nandikonda Archana Reddy Department of ECM, Sreenidhi Institute of Science and Technology, Hyderabad, India

B. Renuka Prasad Department of MCA, RV College of Engineering, Bengaluru, Karnataka, India

M. Rizwan Delhi Technological University, Maharaja Surajmal Institute of Technology, Delhi, India

Mohammad Abdur Rouf Department of Computer Science and Engineering, Dhaka University of Engineering & Technology, Gazipur, Bangladesh

A. M. Rubayet Hossain Ahsanullah University of Science & Technology, Dhaka, Bangladesh

Gaurav Rudravaram Department of Electronics and Communication Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

Abinash Sahoo Department of Civil Engineering, NIT Silchar, Silchar, Assam, India

M. Sai Jishnu Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

P. Sammual JNTUH CEJ, Jagtial, India

A. Sanjana Department of ECM, Sreenidhi Institute of Science and Technology, Hyderabad, India

S. R. Sannasi Chakravarthy Department of Electronics and Communication Engineering, Bannari Amman Institute of Technology, Sathyamangalam, India

M. Sarosh Umar Department of Computer Engineering, Aligarh Muslim University, Aligarh, India

R. Sathya Kongunadu College of Engineering and Technology (Autonomous), Trichy, India

M. Seetha G. Narayanamma Institute of Technology and Science, Hyderabad, India

Sana Shahab College of Business Administration, Princess Nourah Bint Abdulrahman University, Riyadh, Saudi Arabia

A. H. M. Shahariar Parvez Department of Computer Science and Engineering, Dhaka University of Engineering & Technology, Gazipur, Bangladesh

Md Shahnewaz Tanvir Ahsanullah University of Science & Technology, Dhaka, Bangladesh

M. Sheshikala School of CS and AI, Department of CS and AI, SR University, Warangal, India

M. Shobitha Department of Computer Science Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

S. Shreyashree Department of Computer Science and Engineering, M S Ramaiah Institute of Technology (Affiliated To VTU), Bengaluru, India

Kumar A. Shukla SCOPE, Vellore Institute of Technology, Vellore, TN, India

Atique Siddiqui Bharatiya Vidya Bhavan's Sardar Patel Institute of Technology, Mumbai, India

R. Prakash Sidharth Department of Computer Science Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

S. Siji Rani Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

Divyanshu Singh Gautam Buddha University, Greater Noida, Uttar Pradesh, India

Pranjal Singh Department of Computer Science and Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

Upma Singh Delhi Technological University, Maharaja Surajmal Institute of Technology, Delhi, India

Md. Sirajul Islam Department of Computer Science and Engineering, Dhaka University of Engineering & Technology, Gazipur, Bangladesh

R. Siva Subramanian Anna University, Chennai, India

Nidhi Sonkar Department of Information Technology, National Institute of Technology Raipur, Raipur, India

P. K. Sreesruthi Department of Computer Science Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

Padmamala Sriram Department of Computer Science and Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

Gnanou Florence Sudha Department of Electronics and Communication Engineering, Pondicherry Engineering College, Puducherry, India

Ahmed Yousuf Suhan Curtin University, Bentley, Australia

Pramod Sunagar Department of Computer Science and Engineering, M S Ramaiah Institute of Technology (Affiliated To VTU), Bengaluru, India

P. S. Surekha Department of Electronics and Communication Engineering, Malla Reddy Engineering College and Management Sciences, Medchal, Telangana, India

S. Suriya Prasath SIES GST, University of Mumbai, Navi Mumbai, India

Ashok V. Sutagundar Department of Electronics and Communication, Basaveshwar Engineering College, Bagalkot, Karnataka, India

Jayaraman Swaminathan Department of Computer Science Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

B. Swathi School of CS and AI, Department of CS and AI, SR University, Warangal, India

M. Taraka Rama Mokshagna Teja TIFAC-CORE in Cyber Security Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India

S. Thara Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

M. Umavathi Department of Electrical and Electronics Engineering, B.M.S. College of Engineering, Bengaluru, India

A. Usha Department of Electrical and Electronics, B.M.S. College of Engineering, Bengaluru, India

Abhishek Vadher Murdoch University, Murdoch, Australia

K. Vaidehi Department of Computer Science and Engineering, SCETW (Autonomous), Hyderabad, India

Utkrist Arvind Varma Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

P. Varun Raj Department of Computer Science Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

Sevagen Veerasamy Department of Computer Science and Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

SaiSuma Vennam Department of Computer Science and Engineering, Rajiv Gandhi University of Knowledge Technologies—IIIT, Basar, Telangana, India

Jerin Verghese Department of Computer Science, Maharaja Surajmal Institute of Technology, New Delhi, India

K. Vijay Anand Department of Electronics and Instrumentation Engineering, R.M.K. Engineering College, Chennai, India

S. Vijayalakshmi Department of Electronics and Instrumentation Engineering, R.M.K. Engineering College, Gummidipoondi, Chennai, India

V. Vijayalakshmi Assistant Professor & Head, Department of Computer Science, Government Arts College (Grade-I), (Affiliated to Bharathidasan University), Ariyalur, India

K. Vijayanand Department of Electronics and Instrumentation Engineering, R.M.K. Engineering College, Gummidipoondi, India

K. Vinutha Department of Electrical and Electronics, B.M.S. College of Engineering, Bengaluru, India

P. A. Vishnu Department of Computer Science and Applications, Amrita Vishwa Vidyapeetham, Amritapuri, India

Khudov Vladyslav Kharkiv National University of Radio Electronics, Kharkiv, Ukraine

Rajan Vohra Department of Computer Science, Liverpool John Moores University, Liverpool, UK

Abhishek Kumar Yadav Department of Computer Science and Engineering,
Centre for Advanced Studies, Dr. A.P.J. Abdul Kalam Technical University,
Lucknow, Uttar Pradesh, India

Ashok Yadav Centre for Advanced Studies, AKTU, Lucknow, Uttar Pradesh, India

Solomonenko Yuriy Ivan Kozhedub Kharkiv National Air Force University,
Kharkiv, Ukraine

Sign Language Recognition: A Comparative Analysis of Deep Learning Models



Aswathi Premkumar, R. Hridya Krishna, Nikita Chanalya, C. Meghadev,
Utkrist Arvind Varma, T. Anjali, and S. Siji Rani

Abstract Sign language is the primary means of communication used by deaf and dumb people. Learning this language could be perplexing for humans; therefore, it is critical to develop a system that can accurately detect sign language. The fields of deep learning and computer vision with recent advances are used to make an impact in sign language recognition with a fully automated deep learning architecture. This paper presents two models built using two deep learning algorithms; VGG-16 and convolutional neural network (CNN) for recognition and classification of hand gestures. The project aims at analysing the models' performance quantitatively by optimising accuracy obtained using limited dataset. It aims at designing a system that recognises the hand gestures of American sign language and detects the alphabets. Both the models gave excellent results, VGG-16 being the better. VGG-16 model delivered an accuracy of 99.56% followed by CNN with an accuracy of 99.38%.

Keywords American sign language · Deep learning · Sign language · VGG-16 · Convolutional neural network · Neural network · Feature extraction

1 Introduction

Sign language is a method by which the deaf and/or dumb individuals communicate through visual gestures. It is expressed using hand gestures, movements, orientation of palm and face expressions executed by humans. It is the most expressible way for communication for the individuals with hearing or speech impairment. Sign languages have their sign grammar and lexicon. It is not common universally, and thus, each country has its own sign language system. There are about 150 sign languages as per the 2021 edition of Ethnologue. American sign language (ASL) is one of the most leading sign languages of the deaf and dumb individuals of USA as well as of Anglophone Canada. ASL uses ASL manual alphabet/ASL fingerspelled alphabet. Fingerspelling is the method in which a particular word is spelled out

A. Premkumar · R. Hridya Krishna (✉) · N. Chanalya · C. Meghadev · U. A. Varma · T. Anjali · S. Siji Rani
Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

using hand gestures. Each sign shown corresponds to a letter of the word. One of the challenges faced is that normal people find it difficult to understand the gestures of sign language, thus making communication burdensome. Deep learning models have given efficient results in sign language recognition from hand gesture images [1]. Much research has been done in deep learning to find an efficient method of sign language detection. There are various neural networks such as the convolutional neural network which comprises of various layers such as convolutional layers, pooling layers and fully connected layers, fully convolutional network (FCN) in which all learnable layers are convolutional, thus having lesser number of parameters and maintaining the spatial information of the input hand gestures images. Considering the importance of sign language and its efficient recognition to help deaf and dumb individuals to communicate with society, comparative research was conducted in two different deep learning modes, namely VGG-16 and a CNN model, by training and testing each model with hand gesture images.

2 Related Works

American sign language recognition and detection is not a novel concept. Over the past two decades, researchers have made use of various classifiers that belong to different categories such as linear classifiers, Bayesian, neural networks (Table 1).

3 Dataset

The initial step of the proposed system is to collect the data. The dataset consists of 17,113 American sign language hand gesture images from 27 classes (26 alphabets + 1 space class denoted as '0') out of which 12,845 images were used for training and 4268 images for validation (Fig. 1).

4 Data Augmentation

Image data augmentation technique is done using the ImageDataGenerator class imported from K. This is used in expanding the dataset in order to boost the performance and strengthen the model to generalise. Thus, more data result in better accuracy and efficiency of the model.

Table 1 Related works

References	Classification model	Focus
[2]	SqueezeNet architecture	The system uses RGB images which are then used to train the SqueezeNet architecture, so that it could be run on mobiles
[3]	SVM and ANN	Develops a system for Indian sign language recognition using feature extraction techniques, scale invariant feature transform (SIFT) and histogram of oriented gradients (HOG), and then classifies
[4]	Support vector machine (SVM)	Researches explain skin segmentation can be completed using YCbCr systems. This was then classified using SVM
[5]	Histogram matching and ORB algorithm	Developed an android application that captures the hand gesture and detects the sign into digits and alphabets. Proposed methodology involves preprocessing the real-time image, recognising gestures using histogram matching and ORB algorithm
[6]	PNN and KNN	Hand gestures are recognised and translated into text and speech (Hindi as well as English) by using MATLAB. The classification is done using two models and results compared
[7]	SVM and ANN	Involves a review of various steps involved in hand gesture recognition. The methods used for data acquisition, image processing, segmentation and feature extraction were compared. And, the models were then classified
[8]	HMM	Involves comparison of various vision based sign recognition systems, which mostly uses HMM as base. Detailed common challenges of these models
[9]	CNN (SignNet)	A CNN model, SignNet, is proposed by combining high-and low-resolution network CNNs. It works at various spatial resolutions and detects hand gesture images using a synthetic dataset

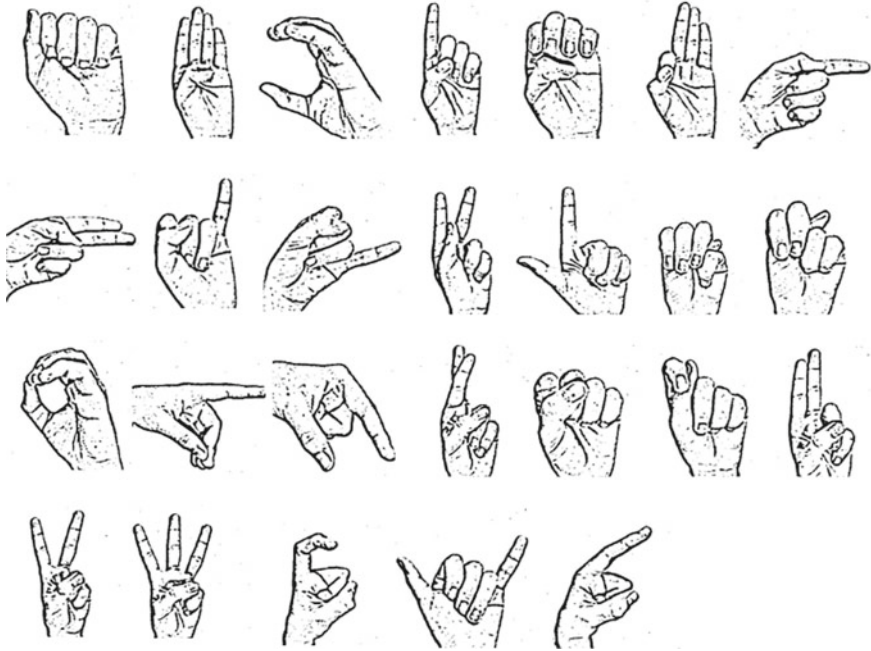


Fig. 1 Dataset

5 Methodology

Two deep learning models were developed for the problem statement, and a comparative analysis was performed on these models based on their training statistics and results.

5.1 Model 1: CNN Model

Introduction

Convolutional neural network (CNN) is a class of deep neural networks that is used in object detection, image recognition, image classification, etc. [10, 11] CNN architecture has a similar architecture to the nerve cells that communicate with the interconnected neurons in the body. The important core layers in the CNN architecture include Input, Padding, Convolution + Activation/ReLU, Pooling, Flatten/Dense, Fully Connected + Softmax. In CNN, the input image is assigned importance to certain features in it to differentiate one from another. Each input image was passed through a series of convolutional layers followed by chosen parameters, and filters

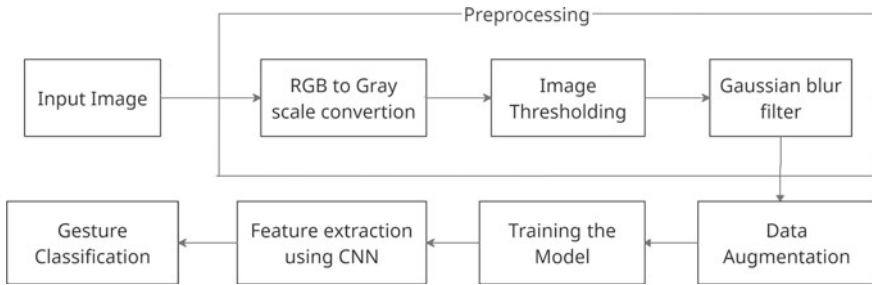


Fig. 2 CNN flow process

with strides were applied and padded whenever required. It gives the highest accuracy in comparison with other image processing algorithms.

Flow Process

A 2D convolutional neural network (CNN) with tensor flow library was developed for training the required models in order to do the detection.

Steps for classification (Fig. 2): Step I: Importing Keras libraries and packages. Step II: Data loading and preprocessing. Step III: Building CNN model with two convolutional layers with rectified linear unit (ReLU), which is the activation function that extracts different features of the input, two MaxPooling layers to gradually decrease the spatial size of the image representation in order to decrease the number of parameters, thus decreasing computational complexity in the model network, the flatten layer, and then, finally, a fully connected layer in which the last dense layer has Softmax as activation function which will execute the classification based on extracted features. Then, the CNN model was compiled with loss ‘categorical_crossentropy’ and ‘adam’ as optimiser. Step IV: Using ImageDataGenerator to apply transformations and augmentation on images for training.

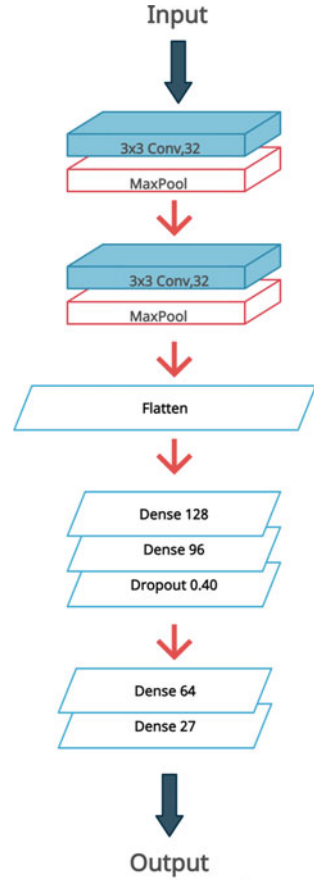
Preprocessing

Preprocessing of coloured images is necessary to extract features from the images. The coloured images were first converted to grayscale as the grey scale images are easier to process than coloured images which will take plenty of resources and time for training the data. Image thresholding was applied to detect the boundaries, thus separating the required object from the background pixels. This was followed by the application of a Gaussian blur filter, which helps to reduce the random noise and for cutting the extreme outliers [12].

Architecture

The images pass through the following layers (Fig. 3): (i) one convolutional layer of size $126 \times 126 \times 32$, succeeded by a pooling layer of size 2×2 that decreases the height and width of the image into $63 \times 63 \times 32$; (ii) one convolutional layer of size $61 \times 61 \times 32$ succeeded by a pooling layer of size 2×2 that further decreases the height and width of the image into $30 \times 30 \times 32$; (iii) flatten layer; (iv) one dense

Fig. 3 CNN network architecture



layer with 96 units and along with a dropout after that of 96 units; (v) two dense layers, first one with 64 units and the second one with 26 units, 1 for each class. The flatten layer and dense layer reduce the data into one dimension and identify the class into which it belongs.

5.2 Model 2: VGG-16

Introduction

VGG-16 (OxfordNet), which stands for Visual Geometry Group, is an object recognition model in deep learning [13]. It is a convolutional neural network architecture that is 16 layers deep. The default input size of this model is 224×224 pixels with three channels for the image in RGB format [14]. The receptive field used by the

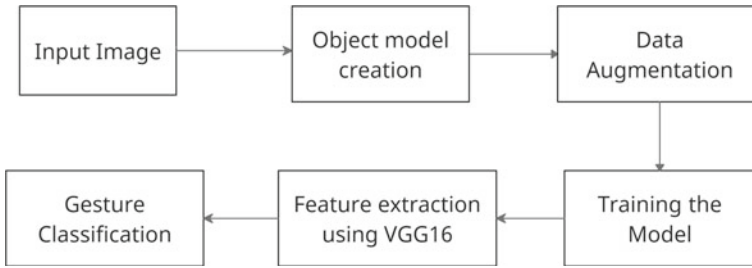


Fig. 4 VGG-16 flow process

convolutional layer in the VGG-16 model is very small, i.e., 3×3 . To retain the spatial resolution after convolution, the convolution stride is set to 1 pixel. It has pooling layers of size 2×2 . VGG-16 has three fully connected layers. The ReLU activation unit is used by all hidden layers.

Flow Process

The steps for classification are (Fig. 4): Step I: Imported Keras libraries and packages. Step II: Loaded the dataset. Step III: The VGG-16 model was built. Convolutional layers with small size convolution filters were added so as to have a large number of weighted filters. After each block of convolutional filters, one max pooling layer was added which helps to reduce the amount of data sent to the next layer by a factor of 4. After the 5 blocks of convolution and pooling layers, the flatten layer was added in order to change the data dimension into a one-dimensional input into the dense layers. Lastly the dense layers were added in which the neurons of different layers are connected into a network. A sigmoid activation unit is added to the last dense layer. Step IV: Use ImageDataGenerator to apply transformations and augmentation on images for training.

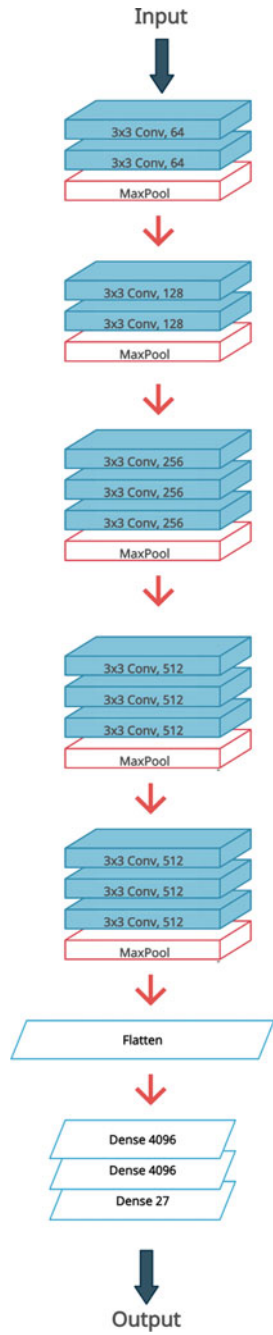
Architecture

VGG-16 architecture (Fig. 5) comprises 13 convolutional layers and 3 fully connected layers, hence not a fully convolutional network.

The images pass through the following layers:

(i) two convolutional layers each of size $224 \times 224 \times 64$ succeeded a pooling layer of size 2×2 , which thus decreases the image size into $112 \times 112 \times 64$; (ii) two convolutional layers each of size $112 \times 112 \times 128$, succeeded by a pooling layer of size 2×2 that further decreases the image size into $56 \times 56 \times 128$; (iii) three convolutional layers each of size $56 \times 56 \times 256$ succeeded by a pooling layer of size 2×2 that further decreases the image size into $28 \times 28 \times 256$; (iv) three convolutional layers each of size $28 \times 28 \times 512$ succeeded by a pooling layer of size 2×2 which further decreases the image size into $14 \times 14 \times 512$; (v) three convolutional layers each of size $14 \times 14 \times 512$ succeeded by a pooling layer of size 2×2 which further decreases the size of the image into $7 \times 7 \times 512$; (vi) flatten layer; and (vii) three dense layers, first two with 4096 units and the last with 26 units,

Fig. 5 VGG-16 network architecture



1 for each class. The flatten layer and dense layer reduce the data into one dimension and identify the class into which it belongs.

6 Result

After training the model, a set of validation images were passed to test the prediction. Then, the performances of both the models were optimised by selecting the appropriate number of epochs and steps per epochs. More epochs give better accuracies, but it could possibly increase the complexity of the model too. For the CNN model, 30 epochs and 200 steps per epoch were used, and for VGG-16 model, 10 epochs and 100 steps per epoch were applied which gave the best results that balanced between accuracy and complexity.

6.1 CNN Model

The test set consists of 4268 images. The proposed model was trained using 30 epochs. From the tests done, an accuracy of 99.38 was obtained for the validation set. The accuracies in various epochs varied from 57.10 to 99.95%. An evaluation on these gave an average of 99.38%.

The train and test accuracy as well as losses were plotted across the number of epochs (Figs. 6 and 7, respectively). Both the accuracies escalated as the number

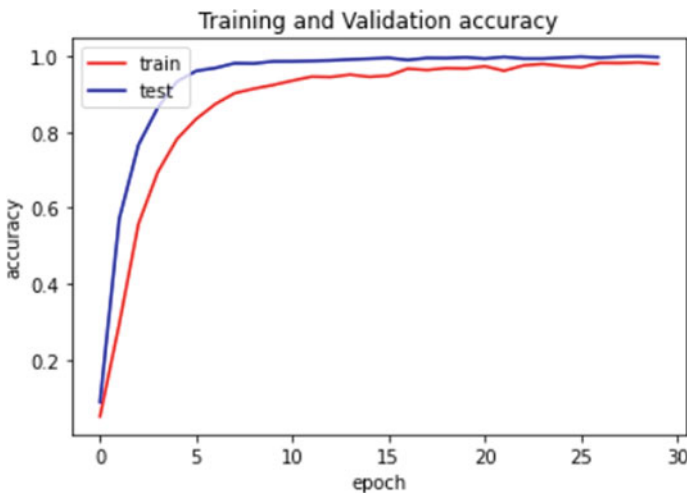


Fig. 6 Plot of estimated training and validation accuracy (CNN model)

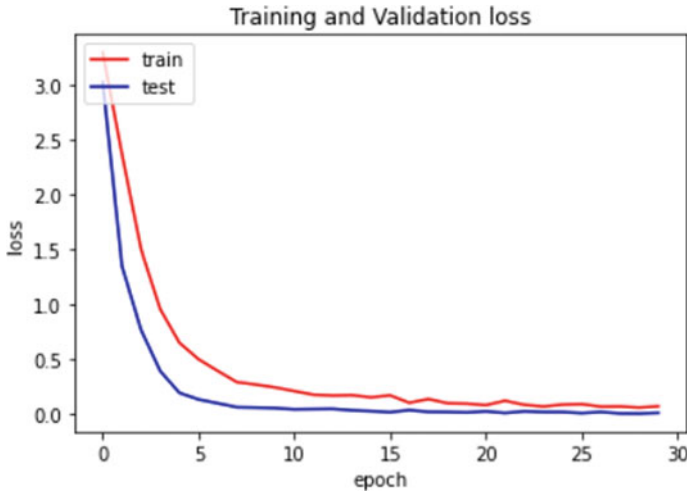


Fig. 7 Plot of estimated training and validation loss (CNN model)

of epochs progressed. The number of losses of both training and testing reduced in subsequent epochs.

The plot of training and testing accuracy (Fig. 6) showed that the accuracy of prediction got higher for higher epochs. The validation accuracy at the first epoch was found to be 57.10%, and it increased sharply till the 5th epoch, with an accuracy of 93.32% after which it increased smoothly till a maximum of 99.95%.

The validation loss decreased sharply (Fig. 7) from the first epoch till the 6th epoch after which it decreased smoothly.

6.2 VGG-16 Model

The validation set (400 images) was passed through the model, and accuracies were optimised by using 10 epochs. As a result, an accuracy of 99.56% was obtained. The accuracies in various epochs varied from 95.88 to 100%. An evaluation on these gave an average of 99.56%. The accuracies of the training and validation were plotted and evaluated (Figs. 8 and 9, respectively). The graphs accuracy of both training and testing increased with increase in the number of epochs. The losses occurred during training and testing were also plotted and evaluated. The losses of both training and testing decreased as the number of epochs got higher.

The plot (Fig. 8) shows that the validation accuracy at the first epoch was found to be 98.62%, and it increased till the 3rd epoch. It then gave an accuracy of 95.88% in the 4th epoch after which it increased sharply reaching a maximum of 100%. The validation loss decreased sharply after the first epoch (Fig. 9).

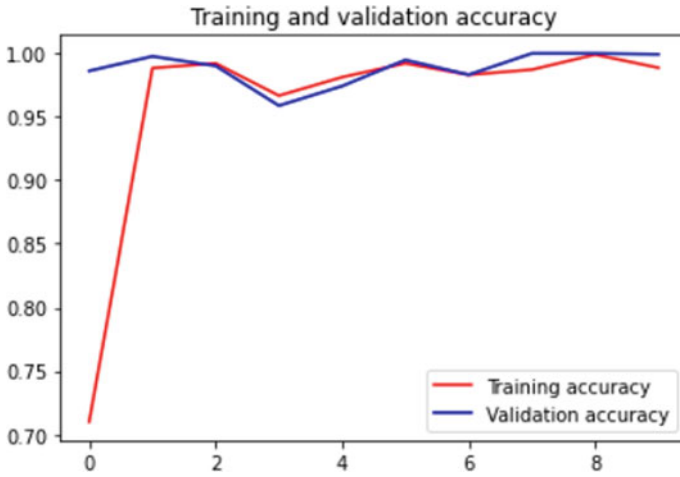


Fig. 8 Plot of estimated training and validation accuracy (VGG-16 model)

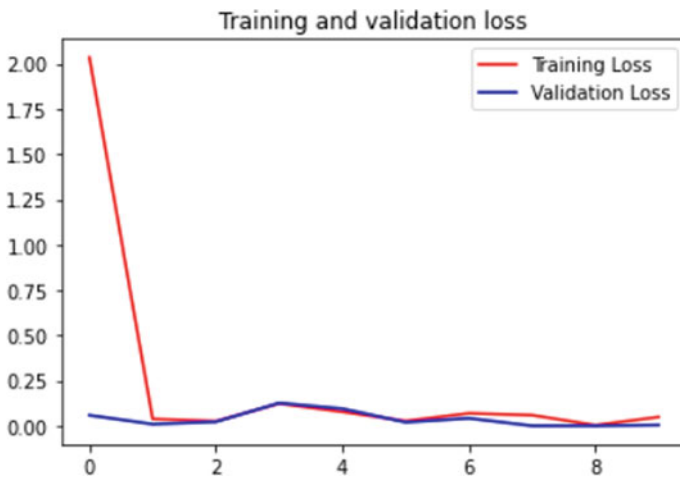


Fig. 9 Plot of estimated training and validation loss (VGG-16 model)

7 Conclusion and Future Works

Considering the complexities of various combinations of hand gestures and its understandability to normal people, there are many challenges in this domain. With this project, two efficient deep learning models were analysed for American sign language detection and the best one was thus recognised. The accuracy of both the models improved as the number of epochs got higher. This is because for each subsequent epoch, the neural network updates the weight estimated in the first epoch with the

values that reduce overall loss. Choosing an appropriate number of epochs, balancing complexity and accuracy produced excellent results. The two models gave excellent results in recognising the gestures correctly. The VGG-16 model with fixed residual blocks gave a better result compared to the CNN model built from scratch. The VGG-16 model fits the data more accurately as there are more weighted layers in VGG which thus had much more parameters which extracted features better than the CNN model, thus classifying better.

The project focuses on recognising the hand gestures that correspond to the alphabet from A to Z using the dataset containing hand gesture images. This project can be developed further to detect images in real time. This will involve the use of LeapMotion API that would help in real-time data generation from hand gestures of people. This could also be developed to recognise hand gestures for words and numbers along with the finger spelling. The project has a scope in further development by using various other models, such as ResNet, which has been proved to be faster and efficient due to its deeper layers.

Acknowledgements The authors feel obliged in taking the opportunity to sincerely thank Anjali T. (Assistant Professor, Computer Science Engineering, Amrita School of Engineering, Amritapuri) for assisting time to time throughout the project duration as well as Amrita Vishwa Vidyapeetham University to provide the golden opportunity to work on this outstanding project on the topic sign language recognition. The authors are overwhelmed with gratitude and humility and acknowledge gratitude to all those who have assisted in bringing these thoughts and ideas far beyond the simplicity and into something substantial.

References

1. R.C. Gonzalez, R.E. Woods, *Digital Image Processing*, 2nd edn. (Prentice Hall, New Jersey, 2008), p. 693
2. N. Kasukurthi, B. Rokad, S. Bidani, A. Dennisan, *American Sign Language Alphabet Recognition using Deep Learning*. [arXiv:1905.05487](https://arxiv.org/abs/1905.05487) [cs.CV] (2019)
3. J. Ekbote, M. Joshi, Indian sign language recognition using ANN and SVM classifiers, in *2017 International Conference on Innovations in Embedded and Communication System (ICIIECS)* (2017)
4. S. Lahoti, S. Kayal, S. Kumbhare, I. Suradkar, V. Pawar, Android based American sign language recognition system with skin segmentation, in *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)* (2017)
5. M. Mahesh, A. Jayaprakash, M. Geetha, Sign language translator for mobile platforms, in *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)* (IEEE, 2017), pp. 1176–1181
6. U. Patel, A.G. Ambedkar, Moment based sign language recognition for Indian language, in *2017 International Conference on Computing, Communication, Control and Automation (ICCUBEA)* (2017)
7. M.J.Z. Omar, M.H. Jaward, A review of hand gesture and sign language recognition techniques. *Int. J. Mach. Learn. Cyber.* **10**, 131–153 (2019)
8. N. Aloysius, M. Geetha, Understanding vision-based continuous sign language recognition. *Multimed. Tools Appl.* **79**, 22177–22209 (2020)

9. N. Aloysius, M. Geetha, An ensemble scale-space model of deep convolutional neural networks for sign language recognition, in *Advances in Artificial Intelligence and Data Engineering Advances in Intelligent Systems and Computing*, vol 1133, eds. by N. Chiplunkar, T. Fukao (Springer, Singapore, 2021)
10. J. Herazo, *Sign Language Recognition Using Deep Learning* (2020)
11. D.A. Sharath Kumar, Sign language recognition with convolutional neural network. *Int. Res. J. Eng. Technol. (IRJET)* (2020)
12. S.A. Khan, A.D. Joy, S.M. Asaduzzaman, M. Hossain, *An efficient sign language translator device using convolutional neural network and customized ROI segmentation*, in *2019 2nd International Conference on Communication Engineering and Technology (ICCET)*, pp. 152–156 (2019)
13. H. Qassim, A. Verma, D. Feinzimer, Compressed residual-VGG16 CNN model for big data places image recognition, in *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*, pp. 169–175 (2018)
14. S. Masood, H.C. Thuwal, A. Srivastava, American sign language character recognition using convolutional neural network, in *Smart Computing and Informatics. Smart Innovation, Systems and Technologies*, vol. 78, eds. by S. Satapathy, V. Bhateja, S. Das (Springer, Singapore, 2018)

Hardware Trojan Detection at Behavioral Level Using Inline Assertions and Verification Using UVM



Aki Vamsi Krishna and E. Prabhu 

Abstract Recently, hardware Trojan (HT) is posing a significant challenge to the integrated circuit (IC) industry and has inspired various improvements in the Trojan identification plans. This research study presents the inline assertions for the detection of hardware Trojan at the behavioral level of a system on chip (SoC). In the proposed RTL design, a modified circuit design flow is suggested to incorporate inline assertions into a SoC. Flexible inline assertions are developed in the RTL block within the design module. The router IP design and inline assertions are synthesized and implemented in Xilinx Vivado and Aldec Rivera Pro using Verilog HDL. The universal verification methodology (UVM) is also used to verify the proposed design with the different test case scenarios. The functional coverage and code coverage are analyzed in Aldec Rivera Pro. Parameters such as power and area are analyzed in the Synopsys design compiler (DC).

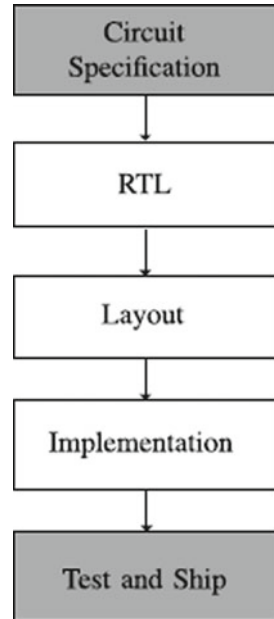
Keywords Hardware Trojan · Behavioral level · Inline assertions · Verilog HDL · UVM · Xilinx Vivado · Aldec Rivera pro · Synopsys design compiler

1 Introduction

With the recent increase in the IC production and the cost of profound sub-micrometer innovation, many IC design houses are currently importing some modules and outsourcing production to the third party (3P), which is considered as a typical practice in the chip improvement cycle. The 3P IP cores and design automation tools are extensively used to improve the circuits. Therefore, the integrity of manufactured product could be undermined. The likelihood that an IC will be susceptible to attack by HT has been increased. A chip, or otherwise a circuit, can be hacked and attacked, resulting in certain modifications if an attacker accesses specific stages of the IC design flow, as shown in Fig. 1 [1].

A. V. Krishna · E. Prabhu (✉)

Department of Electronics and Communication Engineering, Amrita School of Engineering, Coimbatore, Amrita Vishwa Vidyapeetham, Coimbatore, India
e-mail: e_prabhu@cb.amrita.edu

Fig. 1 IC design life cycle

A HT can be described as a malicious change or consideration to IC that changes its functions or cause it to perform an extra malicious function [3]. These malicious incorporations or modifications are mostly customized to activate under a specific set of conditions formulated by an attacker and are extremely difficult to identify or detect at the dormant state. Many different industries, for example, military, telecommunications, and business applications are continuously focused on expanding the challenges faced by malicious circuits that are included for the design [13]. The method proposed to overcome this vulnerability is to utilize HT detection methods at different levels of chip IC design process [4].

The proposed research work is concentrated on one of the level of chip IC design flow that is the RTL block. RTL is a hardware description language (HDL), which is basically the detailed logical implementation of the entire IC and detailed design or circuit specifications that are converted into Verilog or VHDL language. An RTL modeling style is a synthesizable coding style, which could be a data flow model or a behavioral model. The data flow model is a signal that is assigned through the data manipulating equations; all such assignments are concurrent in nature. The behavioral level is the highest level of design description that contains functions, procedural statements, and design modeling. RTL style of coding is widely used in synchronous designs, which involve both combinational and sequential designs. RTL basically represents the data flow between combinational clouds and registers.

This paper focuses on the designing and implementation of router IP protocol using Verilog HDL language. Also, the proposed research work introduces the application of inline assertions in order to detect HT at the behavioral level of the design

or system. The primary goal is to provide a dedicated RTL block that can be a module design with inline assertions in order to detect the HTs during runtime. The primary objectives of this research work are as follows. (1) The inline assertions are proposed within a design module dedicated to finding HT detection. (2) The proposed inline assertions technique is applied on industrial protocols like router IP which contains (FIFO, FSM, synchronizer, register). (3) The proposed router IP design and inline assertions have very little power and area. (4) With different test cases, the proposed design is verified with the latest methodology called UVM.

The paper shows some related works on this concept in Sects. 2 and 3 shows a brief introduction on router IP design and proposed detection method in Sect. 4, correspondingly; Sect. 5 illustrates the UVM test bench. Section 6 shows the simulation outputs, power, area values, and coverage report. Section 7 provides conclusion.

2 Background and Related Works

The authors stated that a better understanding of what HT may resemble and what influence they could have on an IC are necessary. HT is a malicious module, which is introduced inside the IC during the fabrication or design process. Further, they present eight particular attack procedures by utilizing RTL HTs to bargain the safety of an alpha encryption module [2].

HT is made out of a few gates and tries to change the functionality of the chip. These types of HTs are difficult to detect with offline HT detection methods, for example, digital systems tests and side-channel analysis techniques, and authors proposed methodology focuses on an online method for quickly HTs at the runtime [3]. The authors survey made on a different type of HTs present in IC, different Trojans insertions at various stages of IC, and different techniques for detection of the HTs [4].

The author proposed a secured netlist generation using obfuscation techniques without modifying the functionality of circuits with reduced area and power [5]. The authors provide a technique that involves inserting observation sites into the circuit to capture the most difficult-to-observe faults, which works in conjunction with off-chip and on-chip structural testing to provide greater coverage [6].

The authors have implemented the hardware router IP design with different protocol versions like IPv4 and IPv6 results in higher switching speeds of per packet routing for two protocols by applying VLSI architecture techniques [7]. The time of arrival is calculated by using two different time analysis STAs and statistical STAs. The implementation was carried out in ISCAS-89 benchmark circuits with results of the time improvement [8]. Proposed an efficient activity estimator which is fast and accurate and a survey paper on switching activity estimations techniques, power estimation was done in Synopsys DC tool gave a reduction in power for the circuits [9].

The authors proposed a technique for the automated checker generation of the PSL properties for the verification [10]. In this paper, assertions checkers are used

for security of the processors designs during memory instructions, and also survey made on PSL2HDL tool and code coverage techniques to detect malicious [11]. Ngo et al. proposed a built-in assertion checkers that integrate into the design of general useful designs to identify Trojan during runtime in which ACs selection happens pre-synthesis [12]. The author proposed a reconfigurable assertion checker to detection of the HTs at the SoC and demonstrated the mapping of ACs into RAC [13].

Demonstrating the UVM methodology for design verification, explain the UVM test bench hierarchy, registration of factory, components, TLM, mailbox, and callbacks. Different approaches are demonstrated for developing a test strategy and test cases for design verification [14]. This paper describes the implementation of various types of verification mechanisms that can be used with the UVM-based verification environment to improve the ability to protocol verify, hidden bugs, functional checking of design under verification (DUV) [15].

3 Router IP Design

A router IP protocol that forward data packets between computer networks. Packet header contains address based on that it drives the incoming packet to an output channel. At the same time, three parallel connections will support in the router. Router top-level block as shown in Fig. 2, which shows inputs and outputs signals from source network to three client networks.

Router interface of input and out signals defined the functionality of each signal is shown in Table 1. Router design features contain packet routing, parity checking, reset, header, payload, and parity. Packet routing is driven from the input port and

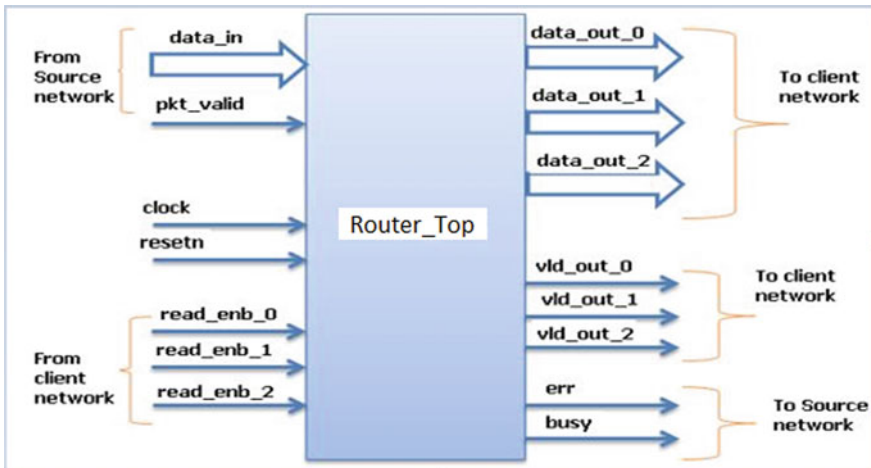


Fig. 2 Block diagram of router top level

Table 1 Router interface

Input/output	Functionality
Clock	Active high clocking event
pkt_valid	Pkt_valid is an active high input signal that detects an arrival of a new packet from a source network
Resetrn	Active low synchronous reset
data_in	Eight-bit input data bus that transmits the packet from source network to router
read_enb_0	Active high input signal for reading the packet through output data bus data_out_0
read_enb_1	Active high input signal for reading the packet through output data bus data_out_1
read_enb_2	Active high input signal for reading the packet through output data bus data_out_2
data_out_0	Eight-bit output data bus that transmits the packet from the router to destination client network 1
data_out_1	Eight-bit output data bus that transmits the packet from the router to destination client network 2
data_out_2	Eight-bit output data bus that transmits the packet from the router to destination client network 3
vld_out_0	Active high signal that detects that a valid byte is available for destination client network 1
vld_out_0	Active high signal that detects that a valid byte is available for destination client network 2
vld_out_0	Active high signal that detects that a valid byte is available for destination client network 3
Busy	Active high signal that detects a busy state for the router that stops accepting any new byte
Err	Active high signal that detects the mismatch between packet parity and internal parity

is routed to any output port, based on the address of the destination network. Parity checking is an error detection being transmitted between server and client. This technique guarantees that the data transmitted by the server network is received by the client network without getting corrupted. Reset is an active low-synchronous input that resets the router, and three FIFO are made empty, and the valid out signals go low indicating that no valid packet on the output data bus. Packet format consists three parts parity, header, and payload; each packet has eight bits width and 1 byte to 63 bytes of the pay load length as shown in Fig. 3. Header destination address has 2 bits of packet, and length has 6 bits of the data. Payload was the data; it is in format of bytes. Parity is used as security to verify of the packet.

The top-level block above as shown in Fig. 2 consists of 6 sub-blocks as shown in Fig. 4, as followed three FIFO, synchronizer, register, and finite state machine.

- FIFO: In router design, three FIFO are used; each one consists of 16 bytes depth and 8 bits width, depending on control signals given by FSM, and it stores the data coming from the input port. The FIFO can be reset by a soft_reset signal; that is, an internal signal is an active high signal of that block coming from the

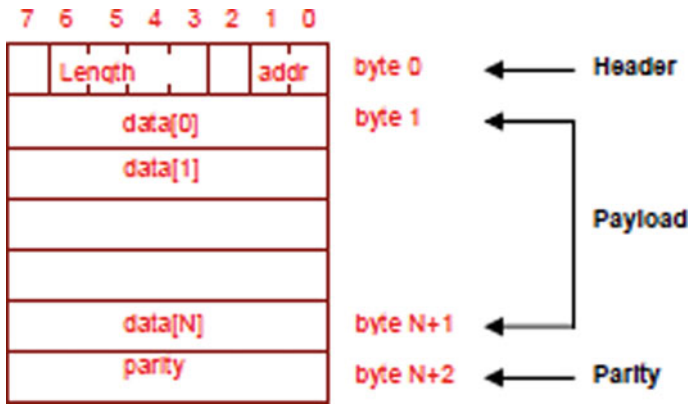


Fig. 3 Packet format

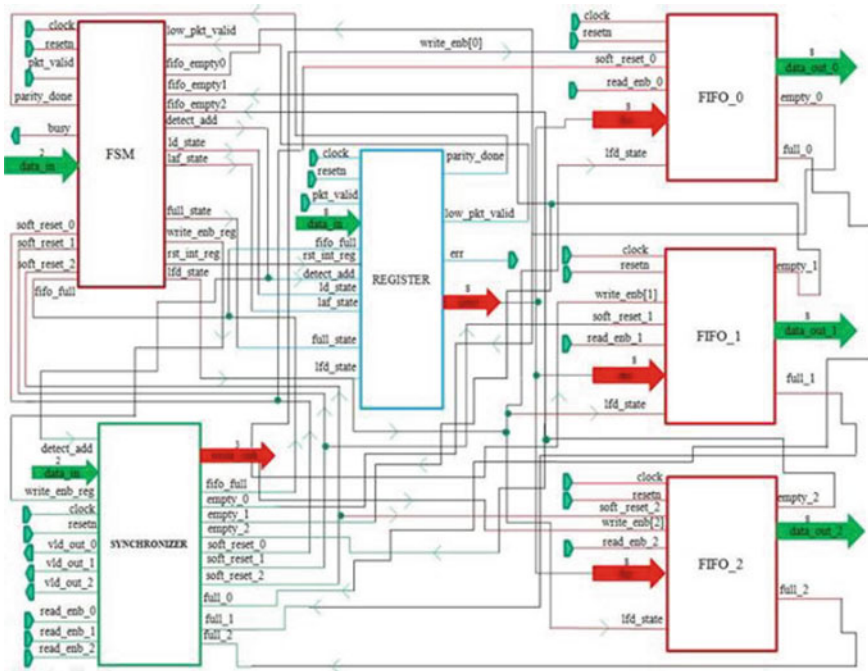


Fig. 4 Block diagram of top-level architecture

synchronizer module during a time out. Reset becomes low, then empty = 1, full = 0, and data_out = 0. Write operation and read operation occur when write_emb, read_end were high, the data_in, data_out sampled at the positive edge of clock, and FIFO is not become full, empty state. Write and read operations can be done at the same time. Full signal demonstrates that all the areas inside FIFO have been

written. Empty signal demonstrates that all the areas of FIFO are empty and have been read.

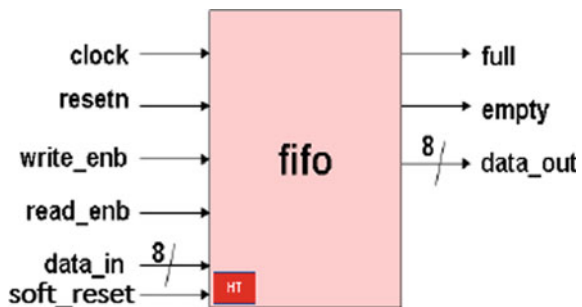
- Synchronizer: This block provides synchronization between FIFO and FSM modules. It also provides correct information between one input port and three output ports.
- Register: This block implements four internal registers to hold, that is, packet parity byte, internal parity, FIFO full state, header byte; all these are register latched on the positive edge of the clock.
- FSM: This block was the controlled design of the router IP. When router gets the new packet, this block generates all controlled signals; these signals are used to transfer the packet to output by other design components.

4 Proposed Detection Method

4.1 Trojan-Free Implementation

In this router IP design, we implemented a HT in a FIFO block as shown in Fig. 5, in order to demonstrate the uses of inline assertions to detect HTs. The HT effect in a design is modification of functionality and leakage of critical information. The FIFO can be reset by a soft_reset in that signal Trojan was added; that is, an internal signal is an active high signal of that block coming from the synchronizer module during a time out. Synchronizer block has three out signal, that is, vld_out_x, and this signal is generate depending at the empty status of FIFO like conditions ($vld_out_0 = \sim empty_0$, $vld_out_1 = \sim empty_1$, and $vld_out_2 = \sim empty_2$). The signals soft_reset_0, soft_reset_1, and soft_reset_2 are three internal signals for each FIFOs, and these signals go high if the read_enb_x like (read_enb_0, read_enb_1, read_enb_2) is not assert within the 30 clock cycles of the vld_out_x. As explained functionality, now, after adding Trojan, the effect on those three internal reset signals of this block, reset signal goes high after one clock cycle without read_enb_x signal not assert within the 30 clock cycles of vld_out_x.

Fig. 5 FIFO block with hardware Trojan



4.2 *Inline Assertions*

Inline assertions are primarily used to validate the behavior of the design and capture the knowledge about how a design should operate. Assertions are the properties, which must be true. Assertion increases the controllability and observability of a design. Controllability is the measurement of the ability to activate, stimulate, or sensitize a specific point within the design. Observability is the measurement of the ability to observe the effects of a specific, internal, stimulate point within the design. Assertions monitor the expected behavior, forbidden behavior, and signal protocols and also depend on the quality of the stimulus.

The objective of the proposed methodology of IC design flow is shown in Fig. 6 to detect the HTs at the behavioral level of RTL block using inline assertions and verified by our IP design with UVM methodology. Inline assertions are embedded check-in RTL code executable specifications, during the simulation phase assertions monitor specific conditions that occur or a specific sequence of events occurs, expected behavior, forbidden behavior, and signals protocols.

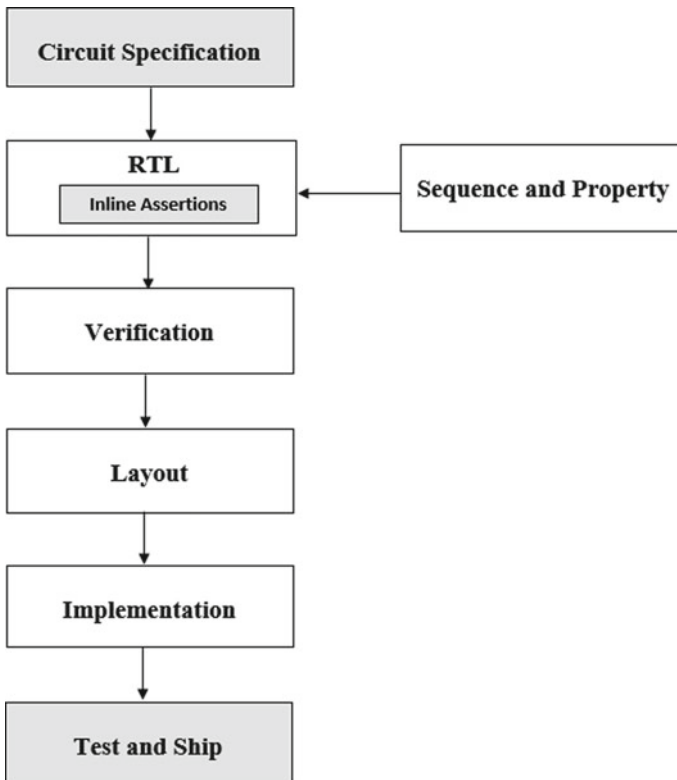


Fig. 6 Proposed IC design flow

Mapping the property and sequence into the proposed design module of inline assertions for Trojan detection. It also produces warnings and errors when a specified condition fails and the sequence does not complete properly. Inline assertions dependent on the clock cycles and test expression are evaluated at clock edges dependent on the variables involved for the sampled values. A variable sampling, evaluations are done at the preponed and observed region of the scheduler. Our inline assertions are placed in a module, interface, procedural block. It can be used with both dynamic and static tools.

5 UVM Verification

UVM methodology is a standard framework to build the verification environment; it has its base class library like `uvm_component`, `uvm_sequence_item`, `uvm_object`. In UVM language, TLM is used as a standard communication mechanism to achieve interoperability configuration of the test bench from the top level. It generates scenarios independent of the test bench environment. UVM achieves reusability in plug and play manner. Typical UVM test bench hierarchy is shown in Fig. 7.

Agent: UVM agent is also called universal verification component (UVC). An agent can be encapsulated, ready to use, reusable, and configurable components. It contains a driver, monitor, and sequencer. Test bench infrastructure can have more than one agent. It can configure as an active and passive agents. Driver: It gets data repeatedly from a sequencer; it drives the DUT based on the protocol using the virtual interface. Derive a driver from the `uvm_driver` base class. Monitor: The monitor extracts information from the bus signal and translates it into transactions. It is connected to other components, via standard TLM ports. Drive a monitor from

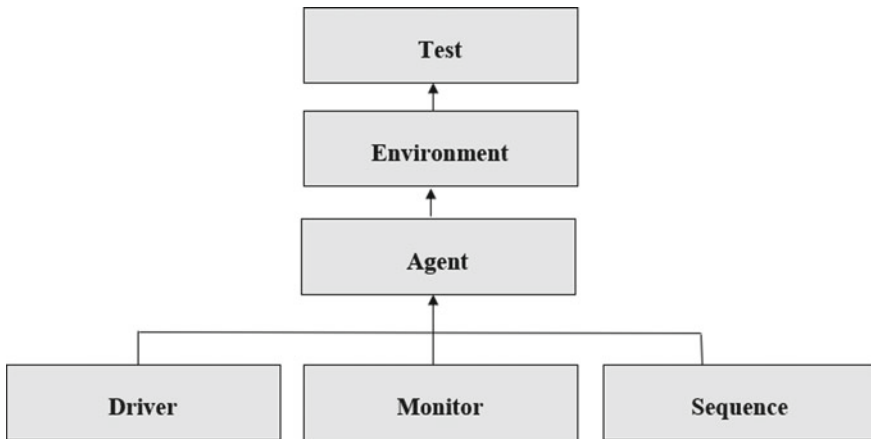


Fig. 7 UVM test bench hierarchy

the uvm_monitor base class. Sequencer: It creates stimuli depending on restrictions and can do so on the fly or at zero time. A factory can be used to override sequences. It is derived from uvm_sequencer. Environment: It is at top of the UVM test bench architecture and contains one or more agents depending on the design.

6 Result and Discussion

The simulation output result of the router top IP design with Trojan implementation is as shown in Fig. 8, and the result of the three data outputs is zero because of the Trojan present in the FIFO block. The data packet could not able to find data coming from source because read enable signal is not becoming high within the 30 clock cycles of valid signal.

The simulation output result of the router top IP design without Trojan implementation as shown in Fig. 9.

The output results of router IP design as shown in Figs. 8 and 9 are synthesized and implemented in Xilinx Vivado tool.

From, Table. 2 shows the output results of inline assertions, and it gives assertions coverage results for each signal at particular sequence and property. As Trojan was added in soft_reset signal, the inline assertions are failing at the soft_reset signal. So, it shows that the Trojan has detected. Inline assertions are implemented in Aldec Rivera Pro tool.

The simulation output result of the design under verification by using UVM is as shown in Fig. 10.

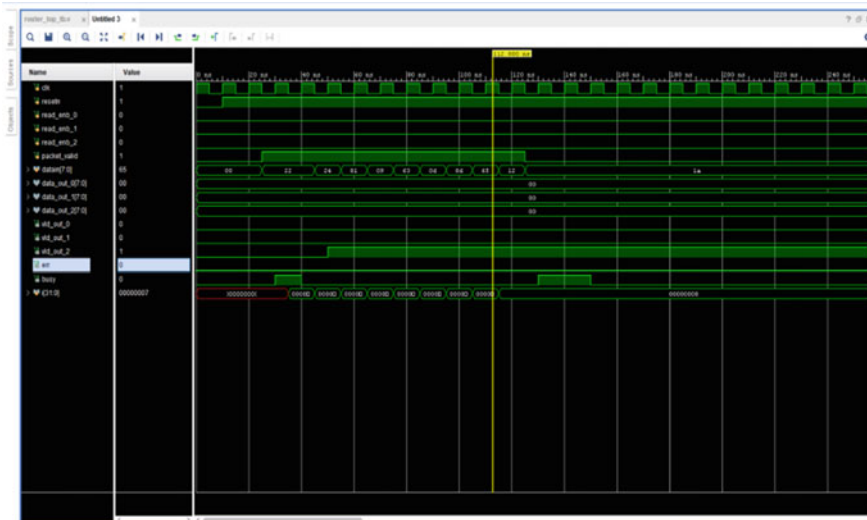


Fig. 8 Router top simulation result with Trojan

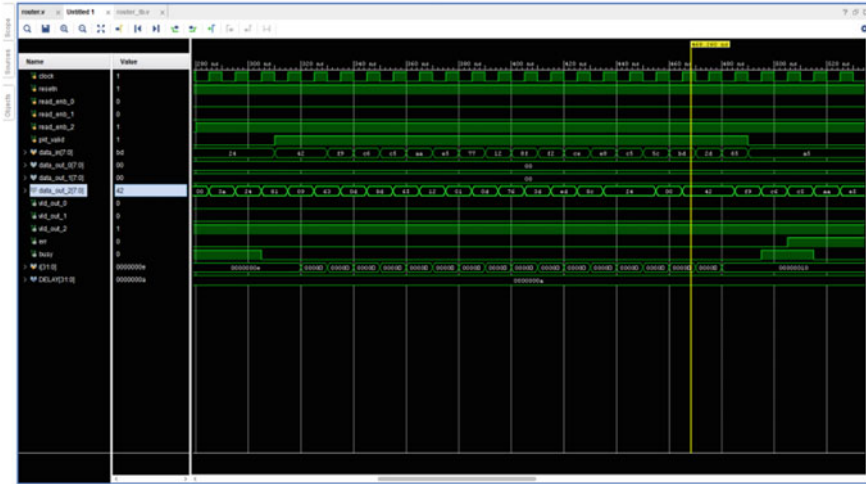


Fig. 9 Router top simulation result without Trojan

Table 2 Inline assertions result

Signal	Assertions		Assertions coverage (5)	Result
	Sequence	Property		
Reset	reset_seq	reset_prty	100	Passed
Busy	busy_seq	busy_prty	100	Passed
read_enb	read_seq	read_prty	100	Passed
ld_state	pvld_seq	pvld_prty	100	Passed
pkt_vld	pvld_seq	pvld_prty	100	Passed
vld_out	vldo_seq	vldo_prty	100	Passed
lfd_state	deassert_seq	deassert_prty	100	Passed
Empty	vldemp_seq	vldemp_prty	100	Passed
Full	fifo_seq	fifo_prty	100	Passed
soft_reset	vld_soft_seq	vld_soft_prty	0	Failed
parity_done	psns_seq	psns_prty	100	Passed
low_pkt_vld	parity_seq	parity_prty	100	Passed

Table 3 shows the result of power and area of router IP design with inline assertions. Our design is synthesized in 90 nm technology with Synopsys DC. From Synopsys DC tool, power and area results are obtained.

The code coverage results are as shown in Fig. 11; total cumulative is statement coverage (SC) with 87%, branch coverage (BC) with 80%, expression coverage with 45%, condition coverage with 69%, and Toggle coverage with 48%.



Fig. 10 DUV output result

Table 3 Power and area result

Parameter	Circuit
Total power (uW)	Router IP design
Total area (μm^2)	8.290
	2221.595629

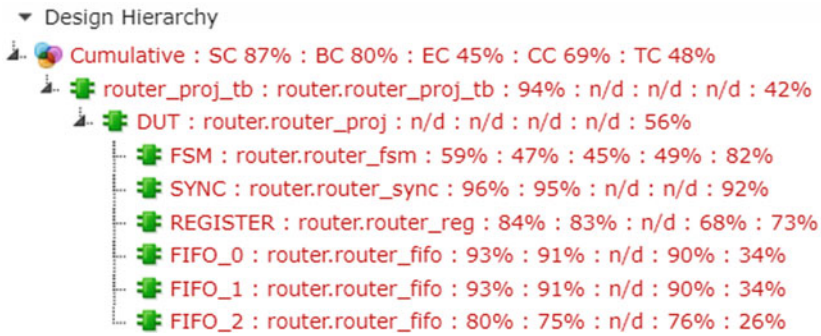


Fig. 11 Code coverage report

Table 4 Functional coverage report

Coverage type	Hits% (%)	Goal/at least (%)	Status
Coverpoint coverage	100	100	Covered
Covergroup coverage	80.555	100	Uncovered

From, Table 4 shows the result of the functional coverage, all test cases or test scenarios have been passed, and all bins are covered.

7 Conclusion

This research work has successfully designed and implemented the router IP protocol using Verilog HDL language. Also, this research work has proposed the application of inline assertions in order to detect HTs at the behavioral level of the design or system. Router IP design with inline assertions occupied very little power and area. And also, the proposed design is verified with different test cases using UVM methodology. Assertions must be defined carefully; incurrent assertions can give misleading results. Debugging assertions will be difficult.

References

1. M. Tehranipoor, C. Wang, *Introduction to Hardware Security and Trust* (Springer, New York, NY, USA, 2011)
2. Y. Jin, N. Kupp, Y. Makris, Experiences in hardware Trojan design and implementation, in *Proceedings of the 2009 IEEE International Workshop on Hardware-Oriented Security and Trust, HOST '09* (IEEE Computer Society, Washington, DC, USA, 2009), pp. 50–57
3. T.F. Wu, K. Ganesan, Y.A. Hu, H.-P. Wong, S. Wong, S. Mitra, TPAD: hardware Trojan prevention and detection for trusted integrated circuits. *IEEE Trans. Computer-Aided Des. Integr. Circ. Syst.* **35**(4), 521–534 (2016). <https://doi.org/10.1109/TCAD.2015.2474373>
4. M. Tehranipoor, F. Koushanfar, A survey of hardware Trojan taxonomy and detection. *IEEE Des. Test Comput.* 10–25 (2010)
5. M. Hemachand, E. Prabhu, Secured netlist generation using obfuscation technique. *J. Critical Rev.* **7**(4), 878–881 (2020)
6. V. Veena, E. Prabhu, N. Mohan, Improved test coverage by observation point insertion for fault coverage analysis. *Proc. Int. Conf. Trends Electron. Inform. ICOEI 2019* **8862789**, 174–178 (2019)
7. C. Mattihalli, S. Ron, N. Kolla, VLSI based robust router architecture. *Third Int. Conf. Intell. Syst. Modell. Simul.* **2012**, 43–48 (2012). <https://doi.org/10.1109/ISMS.2012.32>
8. S.R. Ramesh, R. Jayaparvathy, Artificial neural network model for arrival time computation in gate level circuits. *Automatika* **60**(3), 360–367 (2019)
9. S.R. Ramesh, R. Jayaparvathy, Probabilistic activity estimator and timing analysis for LUT based circuits. *Int. J. Appl. Eng. Res.* **10**(13), 33238–33242 (2015). ISSN 0973-4562

10. M. Boule, Z. Zilic, Efficient automata-based assertion-checker synthesis of PSL properties. *IEEE Int. High Level Des. Valid. Test Workshop* **2006**, 69–76 (2006). <https://doi.org/10.1109/HLDVT.2006.319966>
11. M. Bilzor, T. Huffmire, C. Irvine, T. Levin, Evaluating security requirements in a general-purpose processor by combining assertion checkers with code coverage. *IEEE Int. Symp. Hardware-Oriented Secur. Trust* **2012**, 49–54 (2012). <https://doi.org/10.1109/HST.2012.6224318>
12. X.T. Ngo, J.-L. Danger, S. Guilley, Z. Najm, O. Emery, Hardware property checker for run-time hardware Trojan detection. *Proc. IEEE ECCTD*, 1–4 (2015)
13. U. Alsaiari, F. Gebali, Hardware Trojan detection using reconfigurable assertion checkers. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **27**(7), 1575–1586 (2019). <https://doi.org/10.1109/TVLSI.2019.2908964>
14. IEEE Standard for Universal Verification Methodology Language Reference Manual, in *IEEE Std 1800.2-2020 (Revision of IEEE Std 1800.2-2017)*, pp. 1–458, 14 Sept. 2020. <https://doi.org/10.1109/IEEESTD.2020.9195920>
15. R. Madan, N. Kumar, S. Deb, Pragmatic approaches to implement self-checking mechanism in UVM based TestBench. *Int. Conf. Adv. Comput. Eng. Appl.* **2015**, 632–636 (2015). <https://doi.org/10.1109/ICACEA.2015.7164768>

A Tool to Extract Onion Links from Tor Hidden Services and Identify Illegal Activities



Varun Nair and Jinesh M. Kannimoola

Abstract The dark web is a covered segment of the Internet that provides privacy-protected network access. Tor is a volunteer run prominent dark web network that becomes heaven for criminals to conduct illegal activities. The use of multilayer encryption to achieve anonymity poses a significant hurdle for the law enforcement agency to monitor illicit activities inside the hidden Network. Our study investigates an alternative method to extract the hidden service descriptor from the network. These descriptors also called onion links open a door to hidden services inside dark web. We use a flaw in the v2 protocol to collect the address of hidden service from the memory of a Tor Hidden Service Directory. Automated data extraction and analyzes module provide more insight into contents propagating in Tor network. Using our experiment setup, 4000 onion links are collected and examined. Our analysis shows that socially unjust materials form significant portions of the Tor network.

Keywords Tor · Dark web · Hidden service · Onion protocol · Memory extraction · Hidden service directory · TorBot

1 Introduction

The 21st century, where technology has improved so much to make one's life better, has also brought a list of problems that can cause unalterable consequences in a person's life [13, 20, 25]. The Internet has been a powerful mechanism for communication, facilitating people to connect globally to access and exchange material with no geographical barrier. Dark Web provides uncensored internet to users and prevents threat actors and governments from monitoring users' activity. A great undertaking

V. Nair · J. M. Kannimoola (✉)
Department of Computer Science and Applications, Amrita Vishwa Vidyapeetham,
Amritapuri, India
e-mail: jinesh@am.amrita.edu

V. Nair
e-mail: varunnair@am.students.amrita.edu

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
S. Smys et al. (eds.), *Inventive Computation and Information Technologies*, Lecture Notes
in Networks and Systems 336, https://doi.org/10.1007/978-981-16-6723-7_3

but quickly plagued by many anti-social elements which thrived on secrecy. Adolescents and young adults are having access to materials that are proven to cause mental damage [4]. It provides access to Silk markets, which have daily visits of over 8 million and allows anyone to buy drugs using cryptocurrencies [11]. Our motivation lies in finding illegal undertakings on the internet and identifying them to make the web a safer place. Many frameworks are available on the internet which support anonymous and secure access of web content. We investigate the deanonymization of the hidden services present in Tor, a widely used open source framework in dark web [24].

Tor is a hidden service network run by a group of volunteer-operated servers that focus on improving privacy and security on the Internet. It uses a system of circuits (network of Tor servers) called overlay networks to support uncensored and privacy-preserved content delivery over the Internet [26]. The Tor architecture only allows the client and no other node or even the service to be aware of the complete circuit [6]. The users connect to the service via a series of circuits instead of making a direct connection which allows both parties to transfer information over the public network securely. Only clients with access to specific addresses obtained through out-of-band methods can communicate with the content provider. These addresses, also known as onion addresses, are queried with a Distributed Hash Table (DHT) data structure to retrieve descriptors for specific services hosted by content providers that help to start the communication.

The Onion Routing Protocol [23] uses entry, middle and exit nodes to create a three layer encryption of data that is being transferred between the client and the service. Hidden services are the hosted services in the Tor Network. The enhanced anonymity feature with hidden services in Tor is designed with the help of Introduction Point (IP) servers to mask the content provider. The server creates a descriptor that contains information on how to find the introduction points and how the request access to the service after the assignments of introduction points [16]. It generates the onion addresses from the hidden service public key and pushed the to the distributed hash table (DHT). The DHT is a group of HSDir nodes and receives descriptor requests from clients and receive descriptor publications from the service. Each service gets an ID which is created from the encoded onion link with the function of time and the bit 0 at the end [16]. The IP will give the client the information about Rendezvous Point (RP) where the host will be waiting for the client to establish the connection to the service. This is another part of improving the privacy even if the Introduction Point is known to an adversary it doesn't mean the service is compromised. It is only when the client connects to the RP that the transfer of data takes place.

In our research, we exploit the flaw in the v2 onion link generator and identify the hidden service [14]. We dump the memory of the tor process from `/proc/"torProcess ID"/map` directory and converted to strings which allow us to select the descriptor public key which is used to create generate the onion link. TorBot [17], an open source intelligence tool, is used to crawl the dark web content based on the onion link extracted in the above step. We analyze and categorize the crawled data. Our tool identified over 4000 unique onion links from the Tor HSDir that were hosted by us. We crawled these sites and sent the data to the indexer to identify the content. The

remainder of this paper is organized as follows. Section 2 describes the related work in this field; Sect. 3 gives a detailed description of our methodology; Sect. 4 discusses our experiment setup and result, and Sect. 5 concludes the paper with summary of future works.

2 Related Works

Deanonymization of Tor is one of the most extensively studied research problems. The survey [21], list out the various researches in Tor networks and pointed out that traffic interception is the most frequent factor in deanonymizing Tor. References [3, 5] developed machine learning approaches to differentiate Tor traffic from normal network traffic. Jawaheri et al. [1] demonstrate a method to deanonymizing Tor users in hidden services who rely on Bitcoin as a payment method by utilizing leaked information from social networks. The paper [15] examines a method to uncover the network structures created between websites via hyperlinks. It provides deep insight into the virtual communities forms inside the dark web. The fundamental difference of our approach resides in the crawling points collection.

Researchers attempted various attacks on the Tor to discover the hidden feature of the network [21]. Kadianakis et al. [10] collected around 200 GB of unzipped data about Tor relays from the CollecTor platform. Heninger and Halderman's tool were used to find potential weak keys, i.e., potentially factorable keys. They also found relays that have shared modulus, giving them the ability to calculate each other's private keys. Protocol-level attacks against Tor [12] focuses on how the attacker can duplicate, modify and delete the cells of a TCP stream of a sender and carry out DoS attacks. The effectiveness and countermeasures of the attack are also discussed in the literature. Chakravarty [7] illustrated a novel attack that identifies all Tor relays participating in a given circuit. Sulaiman and Zhioua [22] developed an attack to take the advantage of unpopular ports in the Tor network, which can compromise circuits in the Tor network. In our approach, we exploited the weakness of Tor protocol instead of attacking the network,

Many tools are available to crawl the dark web content [8, 21]. We have used TorBot [17], an open-source intelligence tool to crawl the Tor network. It recursively goes through the pages in a breadth-first search strategy and identifies links, emails, metadata and text. The multi-threading feature improves the performance and allows the user to stay anonymous by randomizing the header information and IP address "n" requests. The intelligence module enables to crawl of these services by using the existing database present in the intelligence extractor. The uniqueness of our approach is that how we extract intelligence from Tor HSDir. We use the data from HSDir to generate the onion links, which allow us to get into the dark web without using a onion browser. The web front in our tool provides a user-friendly interface to analyze the content inside dark web.

3 Design and Methodology

The primary focus of our work is to find illegal services running in Tor through various techniques such as network sniffing, memory extraction, and crawling. Our tool runs on a Tor relay with the hidden service directory flag. The hsfetcher module checks this flag and gathers all memory addresses from /proc/“TorProcessID”/maps. The dumped memory converts to strings and extracts keys of v2 onion links, which serve as the crawling points. These Onion links can be further crawled to obtain fine-grain information about the content providers. Figure 1 outline our methodology and remaining portion explain it in more detail.

Network Sniffing: Network sniffing is a method to obtain the descriptors when they get published by the hidden service or requested by the client. When the client requests a descriptor, it creates an ID derived from the onion address using the hashing mechanism. Although it is not possible to regenerate the address from the ID because of the hashing mechanism. It should be possible to see the response of the proxy which would contain the requested descriptor. This led to the investigation of the possibility of capturing these interactions between hidden services or clients and the HSDir. There are multiple servers which are known as the hidden service directories in the Tor network. They are similar to DNS servers in that they store information on hidden services. When a server running Tor is configured to run as the middle node and HSDir flag is given if it is continuously run for 5 d. The DHT stores all the descriptor information. The process only requires the relay to have the HSDir flag, this will enable the tool to sniff the traffic and move on to the Memory extraction.

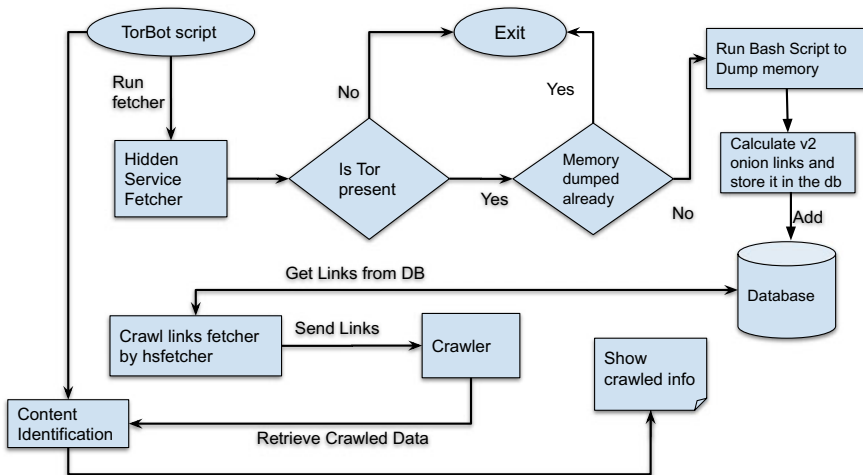


Fig. 1 Overall system

Memory Extraction: We have used memory extraction as a reliable method to capture the descriptor from HSDir server. After attaining the HSdir flag, we first get the process id of the Tor service and access the process's proc/map file. The contents need to be filtered for mappings with read/write permissions since the data being extracted needs to be written and read on the fly. With those mappings, we can extract the start and end memory addresses and pass them to GDB (Debugger) for dumping everything in between. Using the strings utility we convert the dump into a human-readable form. The v2 host's public key starts with "rendezvous-service-descriptor" and ends with "—END SIGNATURE." Feeding this information into a regex, we can extract all the v2 public keys. We followed the subsequent steps to generate the onion address from the extracted public keys.

1. Decode the public key using base64 scheme
2. Pass the value from previous step through the SHA1 hashing function and keep the first 80 bits (half of the output)
3. Encode these 80 bits using base32 encoding and changed into lower-case. This will give the v2 onion link.

Crawling the links: Crawling services refer to the extraction of useful data from the entire web. The crawling process begins with a list of web addresses from past crawls and onion links received by the memory extraction. As the crawlers visit these websites, they use links on those sites to discover other pages. Our crawling service is powered by TorBot [17], which extracts almost all information from a website. It can grab all the HTML data present in the hidden directory. It can also grab other onion links, email addresses, and much more. The tool uses BeautifulSoup [19] python framework to extract HTML data from the web services. This is an effective way to extract intelligence from websites.

Identifying the services: The tool striped the body of crawled data and execute identifier script on them. The script uses a "YAML" file which contains a list of keywords and categories. It search for the keyword over stripped content and categories into distinct classes. Basically, each onion site categories based on keywords present on the site.

Front End We integrated all this module under a GUI created in Angular JS and Flask framework. The results from each query is sent back as a jsonified response.

We extracted the public key of content provider from Tor HSdir and generate the onion address from these keys. The collected onion links act as a entry point to the Tor network, which feed to the crawler for data collection. The data analyzer works on the crawled data and categorize into different classes.

4 Result and Discussions

Our experimental setup included two servers running on Azure cloud with 2 CPU's, 8 GB of RAM and unlimited data bandwidth. We have configured the Ubuntu 18.04 LTS on these systems with Tor 0.4.5.7. All the ports from 9000-9052 were set as open so that Tor server is visible to the outside world and torrc file was changed in order to set it as Tor Relay. The Relay data bandwidth limits to 6 MB/s for incoming traffic and 5 MB/s for outgoing traffic. These servers were hosted in Azure, running the servers for one month will cost the user around 100USD. Due to this running multiple servers will take a decent amount of funding. The high-network traffic in the network can also lead to increased cost. We monitor the data traffic to the server using nix tool [18]. Figure 2 shows the output from nix tool in our server.

Our experimental setup run for 2 weeks and captured 4000 onion links. Figure 3 shows the snapshot of captured links. TorBot crawls the content of onion links and identify the service running on them. Some of the sites contains multiple services. For example, dark markets consist of platforms to sell drugs, weapons, stolen credit cards and even malware. Our identifier script categorized the crawled data into four classes such as dark markets, socially unjust contents, bitcoin exchange and stolen data hosting. Table 1 shows number of site in each category. Some services are more prevent than others. This can be taken in the order of percentage as previous researches points to the same [9].

Figure 4 describes the percentage of contents present in the Tor network. Table 1 and Fig. 4 clearly show that socially unjust content such as child pornography, extremist forms and chat are prominent in the dark web. The tool identifies the prominent use of dark web and found that most of the services that are hosted in the network are illegal and they pose a significant threat to the normal population. Unfortunately, we didn't find any site with useful content. Our research highlight the importance of dark web tracking and improvement of dark web intelligence.

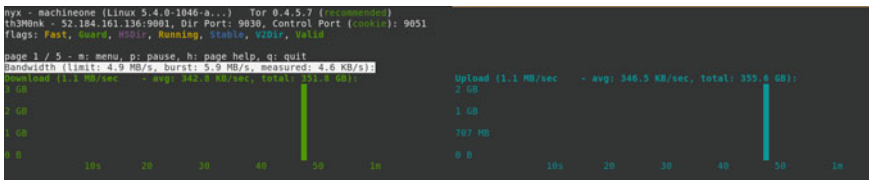


Fig. 2 Output of nix tool

Fig. 3 Onion links

2577	chawqxz7y7e5t7xd.onion	Unknown	None
2578	rmxdeslt5nmu2gce.onion	Unknown	None
2579	tiirylyh4lmykrp6.onion	Unknown	None
2580	zg2txwp55epf6yp7.onion	Unknown	None
2581	3sg3b36rrvotlvmr.onion	Unknown	None
2582	cvfbdqf5wxt77qv5.onion	Unknown	None
2583	hi2yaj2v3yl4nggt.onion	Unknown	None
2584	fsvronp7lnpgonrd.onion	Unknown	None
2585	4dhznyhxfjwpuqko.onion	Unknown	None
2586	dyszthz2r6ps6mlo6.onion	Unknown	None
2587	rws7itrxz4gsjhge.onion	Unknown	None
2588	hackxkci23ntfpqf.onion	Unknown	None
2589	666rmzy3a5jjpbzz.onion	Unknown	None
2590	age3njrobrgnywmn.onion	Unknown	None
2591	sabgsqstdlfr2tp.onion	Unknown	None
2592	ekst rav2rxjhglff.onion	Unknown	None
2593	fvb6ffhf5nvcglwk.onion	Unknown	None
2594	6fi6skicyxa3nfyi.onion	Unknown	None
2595	dwaobj3ym7lzwi4aj.onion	Unknown	None
2596	puyy2lur2eestjqc.onion	Unknown	None
2597	6hgcdpzq6qcf32py.onion	Unknown	None
2598	b7b3adyjnky7jk5k.onion	Unknown	None
2599	wu2frbofdijpa2i5.onion	Unknown	None
2600	d37mc3j5zgzbcf7p.onion	Unknown	None
2601	2jarnl2eko2kewxf.onion	Unknown	None
2602	fbsqxg5ivyfdqq6u.onion	Unknown	None
2603	qaxncfjdd5yzbohq.onion	Unknown	None
2604	amn4hp3p226afpuf.onion	Unknown	None
2605	46re322b5d25dxcg.onion	Unknown	None
2606	cyoqomz5dja4ds2j.onion	Unknown	None
2607	t2pevbp16mq6kujj.onion	Unknown	None
2608	tna7xtsgjshikqyo.onion	Unknown	None
2609	zmgxsojyhsh5g7o3.onion	Unknown	None
2610	tg2ce5yewo5pabhb.onion	Unknown	None
2611	4yrab4g44uzjwioi.onion	Unknown	None
2612	7oq4tfoirdmsx3fx.onion	Unknown	None
2613	h6rzvym7l6fgwoLm.onion	Unknown	None
2614	wg7ke4xy2qffwed7.onion	Unknown	None
2615	c3a3frypgixzvv3.onion	Unknown	None
2616	hxov3hqbuonphemm.onion	Unknown	None

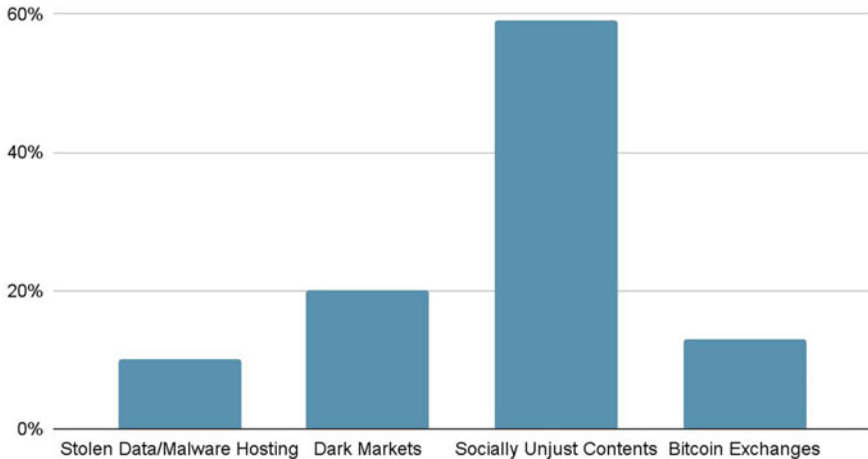
5 Conclusion

The Tor project aims to provide privacy and security to its users and in accomplishing that a larger number of threat actors hosting illegal and unjust contents. This raises the question should the dark web be monitored? Our research focused on the type contents distributed among thousands of hidden services. A front end was developed so that any user can check the contents of these services without the need for prior knowledge on the subject. We found that that the v2 protocol is still used in the Tor network and contents of the services were mostly illegal and socially unjust. These findings lead to the need for proper monitoring of the Tor network with ensuring the privacy of normal users. We would like to further enhance this tool to provide a state-of-the-art solution that can help stop criminals from using anonymity service to threaten our country’s security. Al Nabki et al. [2] classified the illegal activities on TOR network into 26 classes. Integration of such approaches in content identification will improve the accuracy of classification.

Table 1 Number of tor sites

Content type	Number of sites
Dark markets	800
Malware and stolen data hosting	400
Socially unjust contents	2360
Bitcoin exchanges	520

Website Contents

**Fig. 4** Percentage of tor site contents

References

1. H. Al Jawaheri, M. Al Sabah, Y. Boshmaf, A. Erbad, Deanonymizing tor hidden service users through bitcoin transactions analysis. *Comput. Secur.* **89**, 101684 (2020)
2. M.W. Al Nabki, E. Fidalgo, E. Alegre, I. de Paz, Classifying illegal activities on tor network based on web textual contents, in *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, vol. 1, Long Papers. pp. 35–43 (2017)
3. M. AlSabah, K. Bauer, I. Goldberg, Enhancing tor's performance using real-time traffic classification, in *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, pp. 73–84 (2012)
4. G. Ballarotto, B. Volpi, E. Marzilli, R. Tambelli, Adolescent internet abuse: a study on the role of attachment to parents and peers in a large community sample. *BioMed Res. Int.* **2018** (2018)
5. J. Barker, P. Hannay, P. Szewczyk, Using traffic analysis to identify the second generation onion router, in *2011 IFIP 9th International Conference on Embedded and Ubiquitous Computing (IEEE, 2011)*, pp. 72–78
6. A. Biryukov, I. Pustogarov, R.P. Weinmann, Trawling for tor hidden services: detection, measurement, deanonymization, in *2013 IEEE Symposium on Security and Privacy (IEEE, 2013)*, pp. 80–94
7. S. Chakravarty, A. Stavrou, A.D. Keromytis, Identifying proxy nodes in a tor anonymization circuit, in *2008 IEEE International Conference on Signal Image Technology and Internet Based Systems (IEEE, 2008)*, pp. 633–639

8. C. Iliou, G. Kalpakis, T. Tsirikla, S. Vrochidis, I. Kompatsiaris, Hybrid focused crawling on the surface and the dark web. *EURASIP J. Inf. Secur.* **2017**(1), 1–13 (2017)
9. S. Jeziorowski, M. Ismail, A. Siraj, Towards image-based dark vendor profiling: an analysis of image metadata and image hashing in dark web marketplaces, in *Proceedings of the Sixth International Workshop on Security and Privacy Analytics*, pp. 15–22 (2020)
10. G. Kadianakis, C.V. Roberts, L.M. Roberts, P. Winter, Major key alert anomalous keys in tor relays, in *International Conference on Financial Cryptography and Data Security* (Springer, 2018), pp. 3–19
11. E. Kermitis, D. Kavallieros, D. Myttas, E. Lissaris, G. Giataganas, Dark web markets, in *Dark Web Investigation* (Springer, 2021), pp. 85–118
12. Z. Ling, J. Luo, W. Yu, X. Fu, W. Jia, W. Zhao, Protocol-level attacks against tor. *Comput. Netw.* **57**(4), 869–886 (2013)
13. N. Mannilthodi, J.M. Kannimoola, Secure IoT: an improbable reality, in *IoT BDS*, pp. 338–343 (2017)
14. J. Marques, L. Velasco, R. van Duijn, *Tor: Hidden Service Intelligence Extraction* (2018)
15. B. Monk, J. Mitchell, R. Frank, G. Davies, Uncovering tor: a examination of the network structure. *Secur. Commun. Netw.* **2018** (2018)
16. S.J. Murdoch, Hot or not: revealing hidden services by their clock skew, in *Proceedings of the 13th ACM Conference on Computer and Communications Security*, pp. 27–36 (2006)
17. P. Narayanan, R. Ani, A.T. King, Torbot: open source intelligence tool for dark web, in *Inventive Communication and Computational Technologies* (Springer, 2020), pp. 187–195
18. Nyx: Tor Nyx. <https://nyx.torproject.org> [online]. Accessed 17 May 2011
19. L. Richardson, BeautifulSoup. <https://www.crummy.com/software/BeautifulSoup/bs4/doc/> [online]. Accessed 17 May 2011
20. M. Rm, D. Radha, A comprehensive approach for network security, in *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)* (IEEE, 2018), pp. 420–426
21. S. Saleh, J. Qadir, M.U. Ilyas, Shedding light on the dark corners of the internet: a survey of tor research. *J. Netw. Comput. Appl.* **114**, 1–28 (2018)
22. M.A. Sulaiman, S. Zhioua, Attacking tor through unpopular ports, in *2013 IEEE 33rd International Conference on Distributed Computing Systems Workshops* (IEEE, 2013), pp. 33–38
23. P. Syverson, G. Tsudik, M. Reed, C. Landwehr, Towards an analysis of onion routing security, in *Designing Privacy Enhancing Technologies* (Springer, 2001), pp. 96–114
24. Tor: Tor Project. <https://torproject.org> [online]. Accessed 17 May 2011
25. R. Vinayakumar, K. Soman, P. Poornachandran, S. Akarsh, Application of deep learning architectures for cyber security, in *Cybersecurity and Secure Information Systems* (Springer, 2019), pp. 125–160
26. P. Winter, A. Edmundson, L.M. Roberts, A. Dutkowska-Żuk, M. Chetty, N. Feamster, How do tor users interact with onion services? in *27th {USENIX} Security Symposium ({USENIX} Security 18)*, pp. 411–428 (2018)

Adaptive IoT System for Precision Agriculture



V. Geetha Lekshmy , P. A. Vishnu, and P. S. Harikrishnan

Abstract Precision agriculture refers to the application of modern tools and techniques to increase crop productivity in an environment-friendly manner. In the proposed work, a model of self-adaptive system for precision agriculture is developed. This Internet of Things (IoT)-based agriculture system mainly incorporates two functions, automated irrigation and pest detection and is augmented with machine learning models to make it self-adaptive. It handles the sensor failure events automatically by predicting the possible sensor values and keeps the system running without interruption. The system notifies the user about the failure so that it can be replaced later, thus avoiding abrupt termination or malfunctioning of the system. Another adaptive aspect of the proposed system is that it can adjust the system parameters based on prediction of stochastic environmental parameters like rain and temperature. Occurrence of rain is predicted by a machine learning model, and based on this, the system parameters like frequency of getting moisture sensor values are adjusted. This adaptation is fruitful during occurrence of continuous rain when the soil is wet and the moisture content information can be collected less frequently, thus saving the power consumption involved in data collection. The learning models long short-term memory (LSTM) and random forest are used in implementing adaptive functions. The automated irrigation becomes active on fixed times, and the amount of water dispensed is based on the values obtained from soil moisture sensors deployed. The pest detection module captures the images of field and detects mainly the bird pests attacking the crop. The object detection technique, Yolo4, is used to spot the pest.

3rd International Conference on Inventive Computation and Information Technologies ICICIT 2021.

V. Geetha Lekshmy · P. A. Vishnu (✉) · P. S. Harikrishnan
Department of Computer Science and Applications, Amrita Vishwa Vidyapeetham,
Amritapuri, India
e-mail: vishnupa@am.students.amrita.edu

V. Geetha Lekshmy
e-mail: geethalekshmy@am.amrita.edu

P. S. Harikrishnan
e-mail: harikrishnanps@am.students.amrita.edu

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
S. Smys et al. (eds.), *Inventive Computation and Information Technologies*, Lecture Notes
in Networks and Systems 336, https://doi.org/10.1007/978-981-16-6723-7_4

Keywords Adaptive agriculture system · Adaptive IoT systems · Automated irrigation · Pest detection · YOLO · LSTM · Random forest

1 Introduction

Agriculture is one of the most important area where the researchers are focusing on to bring out cost-effective technological solutions. The farmers of India are still in need of user-friendly and effective systems that increase agricultural productivity and reduce human labor. In current era of smart devices, intelligent systems should be devised for performing routine work of farmers like irrigation, pest/disease detection, and weed control. In addition to the routine work, if the smart system is able to adjust itself to changing parameters of the environment with out any human intervention, then that would help to avoid abrupt termination/malfunctioning of the system. There are many research works on automated irrigation systems for agriculture [14, 17–19] and design and development of systems for precision farming [1, 4, 11, 13, 23, 24, 27, 30]. Research work on adaptive IoT systems in area of agriculture includes adaptive power utilization [2, 8], adaptive network mechanism to improve upon network performance in smart farms [25], optimal service selection of agriculture IoT [25] for handling multiple dynamic service requests. Most of these research works in the area of adaptive agricultural systems do not deal with proactive adaptation of systems based on stochastic environmental parameters. In this work, we are trying to deal with both reactive and proactive adaptation of IoT systems based on sensor failure information and environmental parameters. Novelty in this work is the idea of proactive adaptation incorporated in adaptive agricultural systems. This work addresses the following aspects in area of smart agriculture: 1. Automated irrigation and pest detection. 2. Handling of sensor failures using machine learning. 3. Proactive adaptation of agriculture IoT system based on environmental parameters.

Attack of pests is one of the significant problems that the farmers are facing. In order to prevent these pest attacks, farmers use many techniques like placing scare crow, spraying pesticides, using insect traps [12]. In this work, we propose a system that automatically detects presence of bird/insect pests in the field and scares/drives them away using buzzers/pesticides. Here, we are using a PIR sensor, a ESP32 camera module, and YOLO framework for detecting birds and insects in real time. Whenever the pest approaches PIR sensor, it activates the camera and it captures the photograph and sends it to the edge system for further processing. If presence of a bird is detected, then buzzer is activated to scare away the bird. Similarly, if some pests are detected, pesticide pump is activated.

Self-adaptive systems are those systems which adapt itself to satisfy new requirements and environmental changes that arise after the system is up and running. In the proposed system, sensor failures are handled automatically using deep learning techniques that predict the value of sensor whenever the sensor fails to read the value. This system also predicts the possibility of rain and proactively adjusts system parameters, to keep system working with optimum power usage.

2 Related Work

This section includes previous works in area, IoT in agriculture, machine learning (ML) in agriculture IoT systems, adaptive IoT systems, ML-based adaptive IoT systems. A consolidated report on literature is shown in Table 1.

3 Methodology

A working model of the proposed system is implemented, and it deals with the two automated functions in farming, irrigation, and pest detection and control. It is capable of self-adaptation in case of sensor failures, and it also implements proactive adaptation based on rain prediction for adjusting system parameters.

3.1 Background Study

3.1.1 Sensing Subsystem

The heart of the system is NodeMCU, a development board which is powered by an ESP8266 [15], which is a 32-bit micro-controller which is Wi-Fi enabled. It is a low-cost micro-controller, and that makes it an ideal one for our application. The ESP32-CAM module can be widely used in various IoT applications. It has a low-power 32-bit CPU and can also serve as an application processor up to 160 MHz clock speed. It also supports image Wi-Fi upload.

3.1.2 Deep Learning Model

The deep learning model used for the reactive adaption is LSTM. Long short-term memory network (LSTM) is a kind of recurrent neural network (RNN) and is designed to prevent the long-term dependency problem that happens in RNN. It can remember information for long periods of time as their default behavior. LSTM is a powerful algorithm that can classify, cluster, and make predictions about data, particularly, time series and text. Since we are dealing with time series dataset, LSTM is more suitable.

3.1.3 Machine Learning Model

The machine learning model used for the proactive adaptation is random forest. Random forest is a popular algorithm which is a part of the supervised learning

Table 1 Comparative analysis of existing surveys and current study

S. No.	Literature title	Topics focused by authors within the research article	Significant difference of the current research article
1	Design and development of an IoT-based smart irrigation and fertilization system for chili farming [25]	Proposed an IoT-based framework for monitoring and scheduling irrigation using evapotranspiration method.	Designed an adaptive IoT system that monitors environmental parameters and irrigate accordingly
2	IoT-based smart irrigation monitoring and controlling system [28]	Uses a wireless sensor network to monitor the environmental conditions and irrigates the plant accordingly Sensed data are stored on the cloud server for predicting environmental parameters for decision-making and controlling actions	Sensed data are stored on the cloud server for predicting environmental parameters for decision-making and controlling actions
3	IoT-based automated crop protection system [13]	In [13], a crop protection system is proposed where PIR sensor detects the presence of animals in the farm and that activates night vision camera, and snaps are sent to the processing system for identifying animal using The framework uses ultrasound frequencies to keep the animals from entering the field	Designed a low-cost IoT system that uses PIR sensor to detect the presence of birds and small pest in real time, that activates camera, and snaps are sent to the edge system for identifying pests using YOLOv4
4	ScareDuino: smart farming with IoT [20]	Proposed an IoT device that uses motion sensors to detect and repel crows	Buzzers are activated if birds are detected, and pesticides are sprayed if small pests are detected in order to protect the crops
5	Quantitative verification-aided machine learning: a tandem approach for architecting self-adaptive IoT systems [11]	A combined architecture of self-adaptive IoT system with ML and QV is shown. Satisfies the acceptable QoS levels with respect to energy consumption and network traffic.	Proposed a combined architecture of self-adaptive IoT systems with ML and DL techniques
6	A machine learning-driven approach for proactive decision-making in adaptive architectures [23]	An examination of various decision-making techniques in self-adaptive systems was acted in reinforcement learning (RL)-based methodologies for planning and adaptation to make smart decisions in system is presented in [23]	Deals with proactive and reactive adaptations
7	AgriTalk: IoT for precision soil farming of turmeric cultivation [12]	AgriTalk [12] conduct experiments on turmeric cultivation, which indicates that the turmeric quality is significantly enhanced through this system AgriTalk respond to quick and dynamic change of the field environment conditions in soil cultivation	Proposed a self-adaptive IoT systems that respond to quick and dynamic change by predicting the field environment parameters in order to help farmers

(continued)

Table 1 (continued)

S. No.	Literature title	Topics focused by authors within the research article	Significant difference of the current research article
8	A smart agriculture IoT system based on deep reinforcement learning [9]	In this paper [9], a smart agriculture system with the help of a recently emerged advanced AI technique, deep reinforcement learning (DRL), was introduced for smart decision-making, promising model utilized in building smart horticulture systems.	Uses DL techniques to handle the sensor failure case which is a promising model that can be utilized in building smart horticulture systems

algorithm based on decision trees. Random forest builds multiple decision trees by picking “K” number of data points point from the dataset and merges them together to get a more accurate and stable prediction. Here, new sensor values are feed into random forest algorithm to achieve proactive adaptation.

3.1.4 YOLO

The you only look once framework (YOLO) [26] is a real-time object detection algorithm, which is, perhaps, the best object detection algorithm [7]. YOLO utilizes an entirely distinctive methodology [32]. It uses convolutional neural network (ConvNet/CNN) for doing object detection in real time. The object detection is done with a single-layer neural network to the full image and afterward divides the image into locales and predicts bounding boxes for every locales. These bounding boxes are weighted by the anticipated probabilities. YOLO is famous on the grounds that it accomplishes high exactness while also being able to run in real time. The calculation “only looks once” at the image as in it requires just one forward propagation to go through the neural network to make predictions.

3.1.5 Transfer Learning

Transfer learning (TL) is prominent in deep learning method where a pre-trained model is reused to train another. It is popular in the field of deep learning because it helps to train deep neural networks with comparatively small data, and essentially, it attempts to exploit what has been learned in one undertaking to improve ratiocination in another. A pre-trained model for some minimal classes is selected, and that model is enhanced with custom object detection dataset. The new set of weights will help the custom model to become more precise. When TL is used, the time required in training a model is short, when compared to training a model from scratch, and the model will converge faster. Figure 1 shows the illustration of how TL works with a pre-trained model.

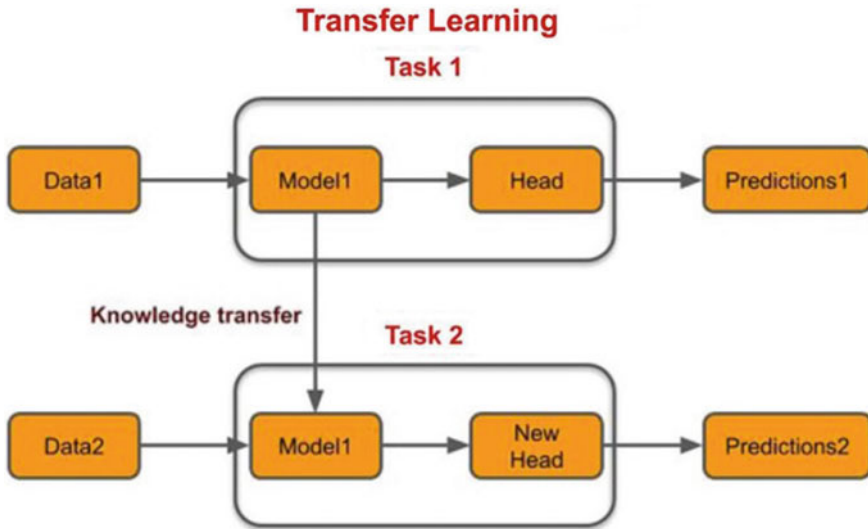


Fig. 1 Transfer learning from pre-trained model [6]

3.2 Agriculture IoT System

Figure 2 shows the overall architecture of the agriculture IoT system. This system consists of different sensors like soil moisture sensors, humidity sensors, temperature sensors connected to a control board. Different actuators like water pump, buzzers, pesticide pump are also connected to it. This control board consists of a NodeMCU module that receives data from all the sensors and sends it to edge system and then to cloud for storage and further processing [28]. The values received from the sensors, like DHT11 and soil moisture sensor, are processed in the control board, and an instruction is given to switch on and switch off the pump. The pump is switched on at a particular time in the morning and evening. The range of moisture sensor values and temperature sensor values determine the time duration of irrigation. All the sensor values are read at fixed time intervals, and it is stored in cloud for further processing. Figure 3 shows the implementation of the IoT system.

The pest detection part of the IoT system consists of an ESP32-CAM Wi-Fi module and a PIR sensor which is deployed in the field. The PIR sensor detects the presence of pests in the field and activates the camera module. This camera module captures the image of the pests and sends it to the locally connected edge system for processing. In that system, an object detection model based on YOLOv4 is used to identify the presence of birds and insects in the image. If the birds are detected, a command is issued to the control board to switch on a buzzer that can scare away [20] the birds, and if the insects are detected, a command is issued to the control board for spraying the pesticide. At the same time, a message is sent to the farmer's mobile

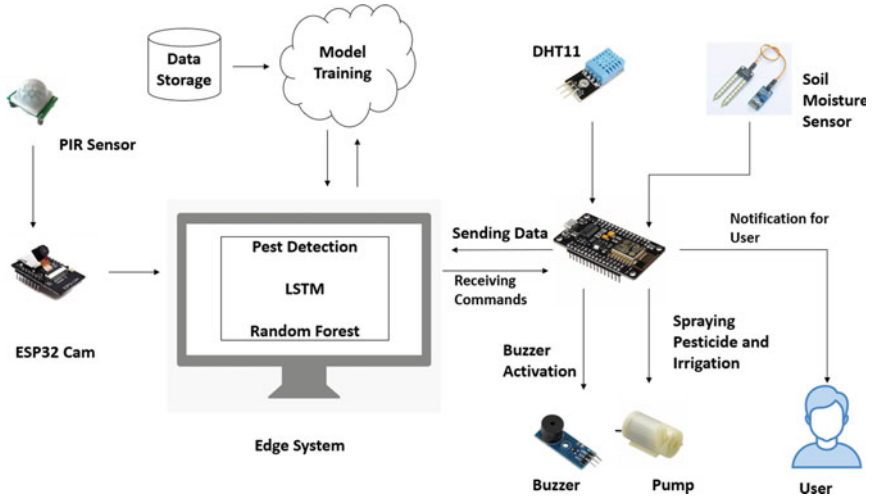


Fig. 2 System architecture



Fig. 3 Proposed system

about the presence of pests. The pest detection is done real time, minimizing the delay in switching buzzer on.

We are using a custom dataset consisting of around 3000 images, which we have collected from Internet and captured using a 12-megapixel (f/1.8) mobile camera from our surroundings. All the images in the training dataset with high resolution are cropped and scaled to a size of 448×448 to reduce computing complexity [34]. These images were labeled using an image annotation tool LabelImg [33]. This tool

is used to create bounding boxes around particular objects in the image. We use rotation, scaling, flip, and other transformations like contrast adjustment in order to avoid over-fitting that may occur in network training [7]. This expands the training dataset to 5000 images. Then, the entire dataset is split into 80% training and 20% validation datasets.

In order to accelerate the training of our custom model, a pre-trained model [3] of YOLO framework is utilized in this work. The model consists of 53 convolutional layers, one pooling layer and one fully connected layer. It is trained for about 6000 iterations. Subsequently, most of the parameters in the network are adjusted in accordance with an adequate reach. At that point, a new convolution layer is added depending on the pre-trained model to convert the classification model into detection model [34], and the network parameters are tweaked with the dataset. Thus, transfer learning [31] used here improved the accuracy of the model and reduced training time.

3.3 Adaptive Agriculture IoT System

Self-adaptive systems are those system that will automatically adjust itself to the changing state of the system. The proposed system mentioned in this paper mainly deals with two types of adaptations. Reactive adaptation and proactive adaptation [22].

Reactive adaptation: Reactive adaptation is the adaptation that takes place only if the system finds out there is a need for adaptation [10]. In this case, the sensor failure is considered as for reactive adaptation.

Proactive adaptation: Proactive adaptation is like the system will frequently check for adaptation to happen before the impacts of climate change are observed [21]. In this case, our system predicts the change in the environment like occurrence of rain and adjusts the system parameters.

In this work, two adaptations are dealt with 1. using predicted sensor values for uninterrupted working of the system and 2. adjusting parameters of IoT system for improving system efficiency say, by reducing power consumption.

In the IoT system, the duration of irrigation depends on temperature and moisture sensor values. So, the sensor failures can affect the seamless working of the system. To cope with that, the proposed system uses deep learning algorithm to predict the value of sensors and prevent the malfunctioning of the system. This work mainly deals with the temperature sensor failure case. When the IoT system fails to record valid temperature values, the system triggers an adaptation process. In adaptation process, the LSTM model predicts the possible temperature value. This value is used for determining the duration of irrigation. Thus, a sensor failure does not affect the proper functioning of the system. This kind of adaptation is reactive adaptation, where adaptation happens whenever a particular event occurs. In addition to this, a message is also sent to alert the farmer about temperature sensor failure.

The LSTM model is created in AWS cloud platform [29]. We have created an instance in AWS and purchased an elastic IP. Using this IP, we bind our LSTM program. Later, we created an API to get values from the sensors. A time series weather dataset from Kaggle is used here which consists of data from 2006 to 2016 [9]. After some preprocessing, this dataset is then made dynamic by adding temperature values from the IoT system working model into it. The dataset is split into 80% training and 20% test datasets.

In this work, proactive adaptation is also done which is based on prediction of rain. The system predicts whether there will be rain for next “t” time units, and this prediction is used for adjusting the system parameters, like the frequency of collecting values from a particular sensor, say moisture sensor. If the weather predicted is not rainy, then the sensor values are to be collected in short intervals of time, that is, more frequently. On the other hand, if rain is predicted, then sensor values need to be collected less frequently. So, this type of adaptation can adjust system parameters without human intervention.

For the rain prediction, we use a dataset [9] consisting of various parameters like temperature, humidity. Random forest model is also created in the same cloud platform. Using the provided elastic IP, we bind our ML model and created an API to get inputs from the sensors. After preprocessing the dataset, then, it is fed into the random forest along with the new sensor values. The dataset is split into 80% training and 20% test datasets. After getting the values in our IoT system, we perform the adaptation by adjusting the frequency of data collected by the sensors.

4 Experimental Results

The performance and results of our proposed system will be explained in this section. The mean average precision (mAP) [5], a popular metric, is used here for measuring the accuracy of object detectors [16]. The mAP compares the ground-truth bounding box to the detected box and returns a score. After 6000 iterations, accuracy obtained from the mAP calculation is as follows:

Last accuracy mAP = 96.66 % and the best = 93.24 %.

For the adaption module, we have used a deep learning model and a machine learning model. The mean squared error is used for obtaining the accuracy of the models. Table 2 shows the accuracy of models we used.

Table 2 Accuracy of our models

Models used	Accuracy
Random forest	93.65
LSTM	88.38

5 Conclusion and Future Work

Though many agriculture IoT systems are available for precision agriculture, many farmers are still depending on the primitive practices which involve huge man power. The farmers are still reluctant to adopt these kind of systems since they are lacking expertise to manage these systems. So, agriculture IoT systems should be user-friendly and self-adjusting/managing so that human intervention can be minimized. This work proposes a basic working model of an adaptive agriculture IoT system, which is cost-effective.

The pest detection module of the system uses ESP32-CAM. However, ESP32-CAM system has its limitations to capture small insects in higher resolution which affects the accuracy of detecting small pest in our system.

Since this system is just a working model, as the future work, it can be enhanced to make into a complete working system that can be deployed in the fields.

References

1. M. Abbasi, M.H. Yaghmaee, F. Rahnama, Internet of things in agriculture: a survey, in *2019 3rd International Conference on Internet of Things and Applications (IoT)* (IEEE, 2019), pp. 1–12
2. S.S. Abou Emira, K.Y. Youssef, M. Abouelatta, Adaptive power system for iot-based smart agriculture applications, in *2019 15th International Computer Engineering Conference (ICENCO)* (IEEE, 2019), pp. 126–131
3. AlexeyAB: darknet (2016). <https://github.com/AlexeyAB/darknet>
4. M. Ayaz, M. Ammad-Uddin, Z. Sharif, A. Mansour, E.H.M. Aggoune, Internet-of-things (iot)-based smart agriculture: toward making the fields talk. *IEEE Access* **7**, 129551–129583 (2019)
5. S.M. Beitzel, E.C. Jensen, O. Frieder, MAP (Springer US, Boston, MA, 2009), pp. 1691–1692. https://doi.org/10.1007/978-0-387-39940-9_492, https://doi.org/10.1007/978-0-387-39940-9_492
6. P. Bhavsar, An ultimate guide to transfer learning in nlp (2019). <https://www.topbots.com/transfer-learning-in-nlp>
7. A. Bochkovskiy, C.Y. Wang, H.Y.M. Liao, Yolov4: optimal speed and accuracy of object detection. [arXiv:2004.10934](https://arxiv.org/abs/2004.10934) (2020)
8. F. Bu, X. Wang, A smart agriculture iot system based on deep reinforcement learning. *Future Gener. Comput. Syst.* **99**, 500–507 (2019)
9. N. Budincsevity, Weatherdataset (2017). <https://www.kaggle.com/budincsevity/szeged-weather>
10. J. Camara, H. Muccini, K. Vaidhyanathan, Quantitative verification-aided machine learning: a tandem approach for architecting self-adaptive iot systems, in *2020 IEEE International Conference on Software Architecture (ICSA)* (IEEE, 2020), pp. 11–22
11. W.L. Chen, Y.B. Lin, Y.W. Lin, R. Chen, J.K. Liao, F.L. Ng, Y.Y. Chan, Y.C. Liu, C.C. Wang, C.H. Chiu et al., Agritalk: Iot for precision soil farming of turmeric cultivation. *IEEE Internet Things J.* **6**(3), 5209–5223 (2019)
12. N.G. Dev, K. Sreenesh, P. Binu, Iot based automated crop protection system, in *2019 2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*, vol. 1 (IEEE, 2019), pp. 1333–1337
13. M. Dholu, K. Ghodinde, Internet of things (iot) for precision agriculture application, in *2018 2nd International Conference on Trends in Electronics and Informatics (ICOEI)* (IEEE, 2018), pp. 339–342

14. L. García, L. Parra, J.M. Jimenez, J. Lloret, P. Lorenz, Iot-based smart irrigation systems: an overview on the recent trends on sensors and iot systems for irrigation in precision agriculture. *Sensors* **20**(4), 1042 (2020)
15. I. Grokhotkov, Esp8266 arduino core documentation. ESP8266 (2017)
16. J. Hui, Map (mean average precision) for object detection (2018). <https://jonathan-hui.medium.com/map-mean-average-precision-for-object-detection-45c121a31173>
17. R. Jisha, G. Vignesh, D. Deekshit, Iot based water level monitoring and implementation on both agriculture and domestic areas, in *2019 2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*, vol. 1 (IEEE, 2019), pp. 1119–1123
18. C. Kamienski, J.P. Soinenen, M. Taumberger, R. Dantas, A. Toscano, T. Salmon Cinotti, R. Filev Maia, A. Torre Neto, Smart water management platform: Iot-based precision irrigation for agriculture. *Sensors* **19**(2), 276 (2019)
19. C. Kamienski, J.P. Soinenen, M. Taumberger, S. Fernandes, A. Toscano, T.S. Cinotti, R.F. Maia, A.T. Neto, Swamp: an iot-based smart water management platform for precision irrigation in agriculture, in *2018 Global Internet of Things Summit (GloTS)* (IEEE, 2018), pp. 1–6
20. L. Lim, H. Sambas, N. MarcusGoh, T. Kawada, P. JosephNg, Scareduino: smart-farming with iot. *Int. J. Sci. Eng. Technol.* **6**(6), 207–210 (2017)
21. A. Metzger, A. Neubauer, P. Bohn, K. Pohl, Proactive process adaptation using deep learning ensembles, in *International Conference on Advanced Information Systems Engineering* (Springer, 2019), pp. 547–562
22. H. Muccini, K. Vaidhyanathan, A machine learning-driven approach for proactive decision making in adaptive architectures, in *2019 IEEE International Conference on Software Architecture Companion (ICSA-C)* (IEEE, 2019), pp. 242–245
23. V. Patil, K. Al-Gaadi, D. Biradar, M. Rangaswamy, Internet of things (iot) and cloud computing for agriculture: an overview, in *Proceedings of Agro-Informatics and Precision Agriculture (AIPA 2012)* (India, 2012), pp. 292–296
24. R. Prabha, E. Sinitambirivoutin, F. Passelaigne, M.V. Ramesh, Design and development of an iot based smart irrigation and fertilization system for chilli farming, in *2018 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)* (IEEE, 2018), pp. 1–7
25. M.R. Ramli, P.T. Daely, D.S. Kim, J.M. Lee, Iot-based adaptive network mechanism for reliable smart farm system. *Comput. Electron. Agric.* **170**, 105287 (2020)
26. J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779–788 (2016). <https://doi.org/10.1109/CVPR.2016.91>
27. P. Rekha, V.P. Rangan, M.V. Ramesh, K. Nibi, High yield groundnut agronomy: an iot based precision farming framework, in *2017 IEEE Global Humanitarian Technology Conference (GHTC)* (IEEE, 2017), pp. 1–5
28. S.B. Saraf, D.H. Gawali, Iot based smart irrigation monitoring and controlling system, in *2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)* (IEEE, 2017), pp. 815–819
29. A.W. Services, Aws (2006). <https://aws.amazon.com/>
30. M. Sharaf, M. Abusair, R. Eleiwi, Y. Shana'a, I. Saleh, H. Muccini, Architecture description language for climate smart agriculture systems, in *Proceedings of the 13th European Conference on Software Architecture*, vol. 2, pp. 152–155 (2019)
31. J. Talukdar, S. Gupta, P.S. Rajpura, R.S. Hegde, Transfer learning for object detection using state-of-the-art deep neural networks, in *2018 5th International Conference on Signal Processing and Integrated Networks (SPIN)*, pp. 78–83 (2018). <https://doi.org/10.1109/SPIN.2018.8474198>
32. M. Tan, R. Pang, Q.V. Le, Efficientdet: scalable and efficient object detection, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10781–10790 (2020)
33. Tzotalin: Labelimg (2015). <https://github.com/tzotalin/labelImg>
34. Y. Zhong, J. Gao, Q. Lei, Y. Zhou, A vision-based counting and recognition system for flying insects in intelligent agriculture. *Sensors* **18**(5), 1489 (2018)

Web Design Focusing on Users Viewing Experience with Respect to Static and Dynamic Nature of Web Sites



R. M. Balajee, M. K. Jayanthi Kannan, and V. Murali Mohan

Abstract In recent days, Web design is mainly focused on screen size of the device and for that they are using some predefined cascading style sheet (CSS) like bootstrap. The responsive HTML is making some impact on viewers based on their viewing experience. A few Web sites additionally allow consumers to drag and drop controls to customize the Web page and freely publish the generated Web page. This has limited the user to a set of predefined Web page design styles. Also, the Web design is focusing and realigning the existing and new content of the page according to the new product recommendation by analyzing the previous purchase and similar purchased item. This recommendation is also based on the gender as well. Individual shopper's decision-making styles are also making an impact on shopping items. The Web details are getting varied on the basis of geographic location and the culture. Depending upon the Web contents, the Web design is changing itself in order to present their contents in a proper way. The researchers are also focusing about the links on the Web page and also about the number of clicks required to reach the particular content and its impact over the user by using this process. The alignment of pictures, videos, and some important text are having ability to contribute for attractive Web design. The papers are based on Web design, and the influence on the user has been taken into account for a survey over here. There are several factors that can influence Web design and the users who use it; we are on our way to developing better Web page designs that improve the viewing experience of users.

Keywords Screen size · Responsive HTML · Viewing experience · Alignment · Links on Web pages · Web design

R. M. Balajee (✉) · M. K. Jayanthi Kannan · V. Murali Mohan
Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation,
Vaddeswaram, Guntur, Andhra Pradesh, India

M. K. Jayanthi Kannan
Department of Computer Science and Engineering, JAIN (Deemed to be University), Bangalore,
India

1 Introduction

The unique Web links are counting more than one billion in today’s world [1]. In these, huge numbers of Web sites are developed by individuals, small- and medium-sized enterprises [2]. The professionals and also non-professionals are now building Web pages due to the easier way of drag and drop approach, which is introduced by some Web sites in order to build our own Web pages to publish [3]. This aspect brings a question mark on the viewing experience of the developed Web pages, even though it is filled up with some useful contents. The design of Web pages is considering the following parameters,

- (i) Text element and formatting
- (ii) Link element and formatting
- (iii) Graphic element and formatting
- (iv) Page formatting
- (v) Page performance
- (vi) Site architecture

Figure 1 illustrates the building blocks of Web page design,

Text, link, and graphical elements and their formatting are the basic blocks of Web design, over that page formatting [4], page performance, and finally site architecture laid on the top.

The building blocks are associated with few aspects and those aspects are listed here,

Text elements aspects: The level of text included on the page, wherein the split up has been made with tables and their visibility.

Link elements aspects: The number of links associated with the page and Web site and link type, which is based on the element, which enables the link.

Graphic elements aspects: A number of graphical items including images, videos, and animations are included on the Web page. It also includes image and video quality and the length of video play.

Text formatting aspects: This deals with font style, size, color, font family, quality of text, underline, bold and highlighting text, etc.

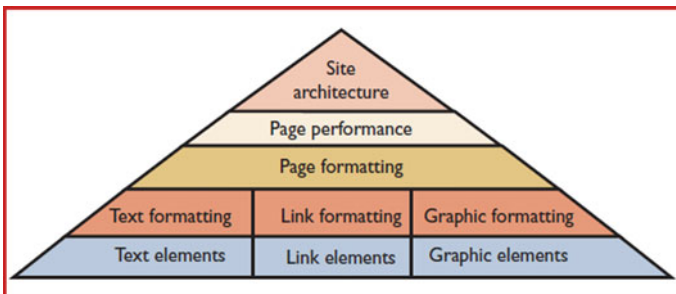


Fig. 1 Building blocks of Web design

Link formatting aspects: Link view ability based on its size, contents surround it and uniform pattern over the Web site.

Graphic formatting aspects: The height and width of graphical and positioning of those elements in Web pages.

Page formatting aspects: This deals with cascading style sheet and alignment of elements in Web page. [5] The combination of color impacts the viewing experience, menu bar placement, navigation links, positioning of interactive elements, etc.

Page performance aspects: Page content loading time is the only aspect we are considering for page performance.

Site architecture aspects: This includes page formatting and page performance in it together. [6] The number pages associated with the current page and entire Web site, number of pages in the Web site their inter link and flow control.

2 Literature Survey

The Web design of non-professional should also be improved and for that one tool has been developed, and this will inspect the building blocks of Web pages with multiple measures on each aspect [7]. A total of 154 measure is being examined to create page and site rating which in turn helps the user in modifying the Web pages. The tool can compute up to 157 pages in a Web site. It has reached 84% of accuracy on feedback. The rating provided by the tool and the participants is examined against each other for finding the accuracy of the developed tool.

The design components of Web pages can also be contents, containers, sidebar and header [8]. Each component can be designed with any width and height but there should be preferable width and height for each. The preferable width and height were found by user voting after viewing various templates participated in the contest [9]. The templates participated and the user voting for that are given as average over here in Table 1, which says about width of the content; similarly scores given by users over the containers width are given in Table 2; the header width and sidebar width are presented in Tables 3 and 4, respectively.

All the Web sites are associated with a set of Web pages and links for that too. The links may be from home page or from any subsequent page [10]. In survey paper [10], they set a threshold value for reaching destination page and they are checking

Table 1 Scoring for width of content

Characteristic	Width in pixels range	Score
Content	968–1200	5
	736–968	4
	504–736	3
	272–504	2
	40–272	1

Table 2 Scoring for the containers width

Characteristic	Width in pixels range	Score
Container	1088–1200	5
	976–1088	4
	864–976	3
	752–864	2
	640–752	1

Table 3 Scoring for header width

Characteristic	Width in pixels range	Score
Header	260–300	5
	220–260	4
	180–220	3
	140–180	2
	100–140	1

Table 4 Scoring for sidebar height

Characteristic	Height in pixels range	Score
Sidebar	260–300	5
	220–260	4
	180–220	3
	140–180	2
	100–140	1

average links required to do that against the threshold value set. If the average is more, then the pages need to be rearranged in such a way to reach the threshold value.

The Web site design is based on the Web site developer idea in early times but it should also consider the end user who is using it, and according to that, the Web site design needs to be re-modified or build [11]. The Razi University students of Iran conducted the survey on Web design and users need to develop satisfied design to build new Web site. For this, they had considered 12 different university Web sites and conducted analysis.

3 Results and Discussion

The weight in Table 5 is obtained from the average marks provided by the users over the 16 parameters specified, namely comfortable, efficiency, simplicity, uniqueness, professional look, attractiveness, perfection, creativeness, neat design, fast response,

Table 5 Web site analysis result

Image identity	Webpage weightage	Page BG Color				Header and footer color				Image section						Logo		Main menu		Multi-language feature	
		Blue color	White color	Golden color	Green color	Blue color	Red color	Gray color	Orange color	Side Large	Side Medium	Side Small	Location Right	Location Center	Location Left	Location Right	Location Left	Vertical text	Drop down	No	Yes
1	2.98	0	1	0	0	1	1	0	0	0	0	1	1	1	1	0	0	1	0	1	
2	4.41	0	0	1	0	0	1	0	0	1	0	0	1	1	0	1	0	0	1	0	
3	2.78	0	1	0	1	0	0	0	0	1	1	0	0	0	1	0	0	1	1	0	
4	1.33	0	1	0	0	1	0	0	0	1	0	1	1	0	1	0	0	1	0	1	
5	2.63	0	1	0	0	1	0	0	0	1	0	1	1	1	0	1	0	1	0	1	
6	2.72	0	1	0	0	1	0	1	0	0	1	1	1	1	1	0	0	1	0	1	
7	3.56	1	0	0	0	1	0	0	0	1	0	1	1	1	1	0	0	1	0	1	
8	1.78	0	1	0	1	0	0	0	0	1	0	1	1	1	1	0	0	1	0	1	
9	4.72	0	1	0	0	1	0	0	0	1	0	1	1	0	1	0	0	1	0	1	
10	4.23	1	0	0	0	1	0	0	0	1	0	1	1	0	1	0	1	0	0	1	
11	1.69	0	1	0	0	1	0	0	0	1	0	1	1	0	1	0	0	1	0	1	
12	1.93	0	1	0	0	0	0	0	1	0	1	1	1	0	1	0	0	1	0	1	
Sum	34.76	7.79	22.56	4.41	4.56	23.86	7.39	2.72	1.93	7.97	23.32	4.47	18.55	31.98	13.67	29.35	5.41	4.23	30.53	2.78	31.98

Table 6 Web sites with shapes

Features	With	Without
Geometric pictures	20	40
Animation	40	20
Duotone presentation	18	42
Grid format	44	16
3D things	13	47
Large text	49	11

plain, lovely, hope, beautiful, update and modern design. The value of 1 and 0 indicates the availability of particular content on that Web site.

In a result of this made survey, it is clear that, white background color with blue header and footer is attractive than other combinations. Medium-size image with center alignment is looking good to score with end user. Right logo placement, dropdown box, and multilanguage feature are also good to have in the developing Web site.

The Web site design from 2013 to 2017 is considered, as well as the differences between them. Banks, schools, and libraries are among the 60 Web sites selected for their investigation. They found to have the following data present in Table 6,

They found that, animation, large typographical things and grid are used in most of the Web sites. Geometric shape, duotone and 3D are less popular among the metric.

Even in the year of 2018, the e-shopping in Pakistan was not so popular like other countries and so a survey has been conducted over that to identify the reason. The reason may be on the Web site design, belief of new online shopping method and some other problems [5]. 156 respondents are selected for this review. They found that, 89% of e-shopping has been done by females. Only 10% of the people are older than 40 years who does the online shopping. 49% of people doing only one transaction on online and continue it further [12]. In the end, they found Pakistanis are afraid to share their personal and economic status online.

Interactivity, color, quality, navigation, content and typography play an important role in Web page design attributes [13–15] to determine the trust, satisfaction and loyalty in terms of avoiding uncertainty in design among various available cultures. The focus on social network will help in collaborative filtering on product recommendation process [16]. Due to various products recommended for different user, the alignment may get changed on the Web page. These e-shopping Web sites should also consider gender of viewers, because of different mindset of male and female toward Web design [17]. Males are independent who most of the time required relevant information by themselves and which in turn reduces the effort in searching process. Women are much interested in social media and as communal nurturers, using the Internet as a tool in the way of maintaining social bonds of them [18–21].

There is a mental model [2] which says about the viewing point over the Web pages regarding particular type of content., based on this the Web content can be extracted from the particular position [22, 23].

When we are speaking about the Web scrapping, the position of the content need to be analyzed before trying to extract it from a particular position. For example, the extracted text may be title or advertisement or the actual content. To make it more precise, the expected area in the screen for each type of Web particulars extraction has been analyzed here, and the result is shown in Fig. 2.

Most of the Web site users are now viewing the Web site on mobile phone [24–26], due to the smart phone availability and Internet connection. The Web site design is moreover based on the screen size of the device which is used to view it, and the differences are shown in Fig. 3 (the same Web page is been opened in mobile

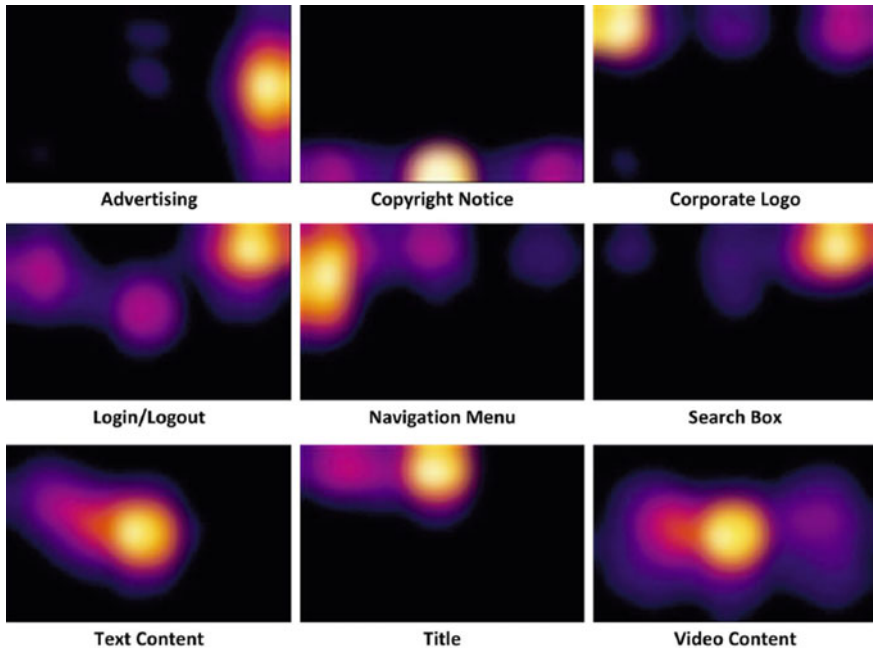


Fig. 2 Viewing areas of human with the expectations from previous experience



Fig. 3 Alignment of Web site in mobile, desktop, and smart TV

phone, desktop, and smart TV). The Web page alignment in different devices has been having the notable difference. Figure 2 is actually the analysis made on the desktop version.

It is not like a desktop application, which can be created with some fixed measurements and had an ability to adjust a little bit according to the screen. The problem on Web design is that, it can be viewed on the mobile phone and as well as in smart TV with some 50 inch screen [27]. The change in web design such like (due to different devices and browser paltfomes) causes major issues in the content alignment and navigation over that [28, 29] and this further leads to make the be changes according to that screen size to provide good viewing experience to user.

The personalized Web pages are increasing in today's world, and retrieval of information from those Web pages is becoming difficult for the search engine. To overcome this problem, in survey paper [30, 31], they proposed a solution, they want Web sites to specify about their origin and user location has been extracted and according to that the Web sites are linked to search information. This will make a way to do efficient search over the available Web sites based on location.

4 Conclusion

The Web design is based on the user's device and hardware support for those devices, as well as the user's gender and mindset. Finally, the Web page displays the content. The Web page design should not be static in nature; else, anyone visiting those Web sites will be turned off by them. The e-shopping Web sites and other commercial Web sites are focusing on attracting the user with their products; this will not be perfectly accomplished without the proper design of Web pages. The proper Web design is relied on the dynamic nature of Web pages and static size and positioning of Web elements, so that the Web pages can also alter its design dynamically according to the demand on viewer side. The concepts discussed here are proving that, the Web pages are dynamically designed with the likes of alignment according to the screen, personalized product recommendation-based views and static design like color chosen, size, and position of each viewing component in Web page, and finally the tools to perform Web page analysis. The aforementioned approaches are making the Web page more attractive and utilizable for the end users.

References

1. A. Lafrance, How many websites are there. *The Atlantic*, p. 30 (2000)
2. D.S. Soper, S. Mitra, The nature, antecedents, and impacts of visuo-spatial mental models of web interface design. *IEEE Access* **4**, 7930–7939 (2016)
3. J. Lu, Y. Zhou, J. Zhang, The concept of WEB design patterns based on the website design process, in *2011 International Conference on 2011-09-24 on Information Technology, Computer Engineering and Management Sciences (ICM)*, vol. 4 (IEEE, 2011), pp. 49–52

4. S.B. Boddu, V.K. Anne, R.R. Kurra, D.K. Mishra, Knowledge discovery and retrieval on World Wide Web using web structure mining, in *2010 Fourth Asia International Conference on Mathematical/Analytical Modelling and Computer Simulation* (IEEE, 2010 May 26), pp. 532–537
5. T.V. Sai, S. Haaris, S. Sridevi, Website evaluation using opinion mining. *Int. J. Eng. Technol. (UAE)* **7**(2), 51–53 (2018)
6. J.K. Sastry, V.C. Prakash, V.S. Virajitha, P. Srija, M. Varun, Structure-based assessment of the quality of WEB sites. *Int. J. Eng. Technol.* **7**(2.7), 980–983 (2018)
7. M.Y. Ivory, M.A. Hearst, Improving web site design. *IEEE Internet Comput.* **2**, 56–63 (2002)
8. D.B. Kamesh, J.S. Bhanu, J.K. Sastry, An architectural approach for assessing quality of web sites. *J. Eng. Appl. Sci. (Asian Res. Publ. Netw.)* **13**(15), 4503–4513 (2018)
9. D. Sorn, S. Rimcharoen, Web page template design using interactive genetic algorithm, in *2013 International 2013-09-04 on Computer Science and Engineering Conference (ICSEC)* (IEEE, 2013), pp. 201–206
10. B. Singh, H.K. Singh, An efficient approach for improving website design, in *2015 Fifth International Conference on 2015-04-04 Communication Systems and Network Technologies (CSNT)* (IEEE, 2015), pp. 945–949
11. F. Noori, S.S. Zadeh, M. Kazemifard, Designing a University web site by considering users' emotion and using Kansei engineering, in *2015 Sixth International Conference of 2015-04-27 Cognitive Science (ICCS)* (IEEE, 2015), pp. 66–71
12. S. Qayyum, M. Rehman, M. Saleemi, I. Ilyas, S. Rafiq, Analyzing the impact of security and website design on E shopping behavior of consumers: a case study of Pakistan, in *2018 International Conference on 2018-03-03 Computing, Mathematics and Engineering Technologies (iCoMET)* (IEEE, 2013), pp. 1–12
13. C.N. Faisal, M. Gonzalez-Rodriguez, D. Fernandez-Lanvin, J. de Andres-Suarez, Web design attributes in building user trust, satisfaction, and loyalty for a high uncertainty avoidance culture. *IEEE Trans. Human-Machine Syst.* **6**, 847–859 (2017)
14. Y. Venkata Raghavarao, K. Sasidhar, J.K.R. Sastry, V. Chandra Prakash, Quantifying quality of WEB sites based on content. *Int. J. Eng. Technol. (UAE)* **7**(2), 138–141 (2018)
15. J.K. Sastry, N. Sreenidhi, K. Sasidhar, Quantifying quality of WEB site based on usability. *Int. J. Eng. Technol.* **7**(2.7), 320–322 (2018)
16. E.Q. Silva, C.G. Camilo-Junior, L.M. Pascoal, T.C. Rosa, An evolutionary approach for combining results of recommender systems techniques based on collaborative filtering. *Expert Syst. Appl.* **53**, 204–218 (2016)
17. M.E. Hupfer, B. Detlor, Sex, gender and self-concept: predicting web shopping site design preferences. *Int. J. Electron. Bus.* **7**(3), 217–236 (2009)
18. K. Macklem, Women to lead new digital gold rush. *Financial Post C*, 56 (2000)
19. M. Krantz, The great online makeover. *Time* **155**(4), 64–65 (2000)
20. J.D. Mosley-Matchett, Marketers: there's a feminine side to the Web. *News Field Marketing* **32**(4), 6 (1998)
21. S.M. Smith, D.B. Whitlark, Men and women online: what makes them click? *Marketing Res.* **13**(2), 20–27 (2001)
22. B.P. Kolla, A.R. Raman, data engineered content extraction studies for Indian web pages, in *Computational Intelligence in Data Mining 2019* (Springer, Singapore, 2019), pp. 505–512
23. K.B. Prakash, Information extraction in current Indian web documents. *Int. J. Eng. Technol. (UAE)* **7**(2), 68–71 (2018)
24. Q. Feng, H.L. Chen, Design a mobile website for university library, in *2011 International Symposium on 2011-12-09 IT in Medicine and Education (ITME)*, vol. 1 (IEEE, 2011), pp. 99–102
25. B.v. Priya, D.J. Sastry, Assessment of website quality based on appearance. *Int. J. Emerging Trends Eng. Res.* **7**(10), 360–375 (2019)
26. J. Bhanu, D.B. Kamesh, J.K. Sastry, Assessing completeness of a WEB site from quality perspective. *Int. J. Electr. Comput. Eng.* **9**(6), 5596 (2019)

27. E. Perakakis, G. Ghinea, Smart enough for the web? A responsive web design approach to enhancing the user web browsing experience on smart TVs. *IEEE Trans. Human-Mach. Syst.* **47**(6), 860–872 (2017)
28. X. Li, J.A. Chishti, The impact of emerging website design features, in *2017 4th International Conference on 2017-11-11 Systems and Informatics (ICSAI)* (IEEE, 2017), pp. 1657–1662
29. J.K. Sastry, V.C. Prakash, G. Sahana, S.T. Manasa, Evaluating quality of navigation designed for a WEB site. *Int. J. Eng. Technol.* **7**(2.7), 1004–1007 (2018)
30. Y. Tang, H. Wang, K. Guo, Y. Xiao, T. Chi, Relevant feedback based accurate and intelligent retrieval on capturing user intention for personalized websites. *IEEE Access* **6**, 24239–24248 (2018)
31. B. Vishnu Priya, J.K.R. Sastry, Computing quality of structure of a web-site. *Int. J. Adv. Trends Comput. Sci. Eng.* **8**(5), 2142–2148 (2019)

Image-Based Authentication Security Improvement by Randomized Selection Approach



R. M. Balajee, M. K. Jayanthi Kannan, and V. Murali Mohan

Abstract In recent days, the value of data stored in disk space (may be localized or cloud) is on the higher side while comparing to the past. In these circumstances, higher the information value provides the higher possibility for data hacking. This situation will develop further and will not be a setback, which implies that the security of those data should be improved as well. This element of security enhancement will be dependent on increasing physical and electronic security. Sensor-based and sensorless approaches can be used to offer electronic data authentication. The sensor-based method depends on specific extra feature on the login device and the state of sensible item with environmental barriers to provide authentication. This raises the question of (i) specified extra feature availability with cost associated with it, (ii) accuracy of sensor devices with respect to sensible item with environmental impact, and (iii) device stability, reliability, and also additional power consumption for sensing device. When coming to sensorless approach for authentication, the simple traditional password scheme is not enough now, and there are some authentication schemes, which will make us to enter different password or pass input ever time, which are already in the pool. This raises the question of the pool size and user remembrance, which is proportional to pool size. If we need a better security, there is a demand to increase the pool size and result in increasing the burden to remember past input for authentication. This research work focuses on reducing the burden of remembering pass input with larger pool. This paper proposes a novel method and implements a bag of password scheme to overcome the aforementioned drawback. As a consequence, with the proposed technique, we are determining the smallest amount of random selection required to select the ideal pool size, resulting in greater authentication security and less complexity from the end user's perspective.

R. M. Balajee (✉) · M. K. Jayanthi Kannan · V. Murali Mohan
Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation,
Vaddeswaram, Guntur, Andhra Pradesh, India

M. K. Jayanthi Kannan
Department of Computer Science and Engineering, JAIN (Deemed to be University), Bangalore,
India

Keywords Authentication security · Electronic security · Sensor-based approach · Sensorless approach · Password scheme · User remembrance · Pool size · Bag of password · Random selection

1 Introduction

The security for the data is becoming the most important concern because of the increasing data theft and hacking of the valuable data. The data which are not even found as valuable but available in huge volume is also attracting the hackers toward that because the analysis over that huge dataset may provide valuable results which may result in finding past trend or future prediction based on past. This sometimes leads to the negative outcomes [1, 2].

The security for the data available in the cloud is the field area to focus and improve. In this aspect, the security can be improved by introducing the encryption algorithms, and the data can be transferred to destination from source [3–6]. This cryptography is having its security concern on transferred data and the most valuable information available in the cloud. When considering the most valuable information available in the cloud, the hackers can break the cryptography only after entering into the page to retrieve the data. This will only happen when the authentication is been successfully done. As a result, when the authentication mechanism is weaker, it exposed the circumstances for the hackers to play with cryptography algorithms. So, the overall security can be improved by improving the authentication security [7].

When trying to improve the authentication security, there are many methods to do authentication. Each of them has been providing its own advantages and drawbacks according to the place where it has been used of [8, 9]. Few of the authentication mechanisms are also combined to produce multistage authentication which in turn increases the security level [10–12] at the same time, and it increases the load on the end user in terms of network resource utilization, time consumption, and the additional steps required from the end user side to complete the process. Even the behavior of the users can be identified for analyzing their characteristics for predicting, the current attempt of login authentication is from the original user or not [13, 14]. In this case, there is always possibility to overpredict or underpredict the user behavior and due to the improper proportion, it leads to a discomfort for the user on the authentication process, and also it leads to other users to login to the system by underpredicting the behavior of the user.

2 Literature Survey

Today, the end users are supposed to be associated with variety of Web portals for their requirements. They have to login different Web portals for example backing based,

entertainment based, education based, virtual meeting based, mail communication logins, etc. So, this introduces the difficulty of using different password, and so, the said reason in turn increases the possibility of using same password for most of the login. This introduces the leak in authentication security. This can be avoided by bringing the different sort of security mechanism which will not allow the user for entering the same password.

One of the attractive authentication security mechanisms is by using the colors and providing the users the option of choosing different color combination to login. This will attract the user and as well as provide some alternative for avoiding repeated password over multiple Web portals.

The graphical way of authentication system [14, 15] introduces such new possibility to avoid the password leak, and the graphical way of authentication may be based on some flash work, some animated video paly, image based, and finally by audio files. When it goes into more graphical approach, then the requirement leads to unusual time consumption in login process with the utilization of much CPU resources. This will cause negative impact over the end user on the authentication process. So, it is very important to handle this at correct level.

The attractive graphical input for authentication process can be fetched by the showcase of color palettes. The user needs to select corresponding color combination on the color palettes to submit the pass input for validation purpose [16]. If it got matched, then the system will authenticate to login to the dashboard or home page.

In the target toward increasing the authentication security, the researchers are also coming up with adding some specialized or specific devices for authentication. The device may be iris scanner and finger print scanner [17–19], and even the innovations lead to palm scanner kind of. The security can be improved by sensor-based devices on authentication process [20, 21]. All these can be more personalized and are capable of providing good security, but think the today's trend about the end user is having multiple computing, processing, and storing devices on different locations which leads to so many additional authentication sensor devices. This increases the cost of authentication as well.

The authentication mechanism had also been strengthening by the external world communication in terms of mobile messages or by e-mail transfer. The additional security strings or the identified security leak warnings are passed to e-mail of end user [22]. So, they can be notified for further process.

3 Proposed Technique: Bag of Password Approach

When the dynamic nature of pass string increases then that will increase the security for authentication. In this way, the attempt to increase the pass string depends on the stored equivalent in servers database. To match these two (pass string and stored value), user wants to remember all the pass strings to give exact one as an input.

To produce one of the pass strings on the pool, the one need to be filtered based on the conditions and hint selected by the program. The same condition will be applied

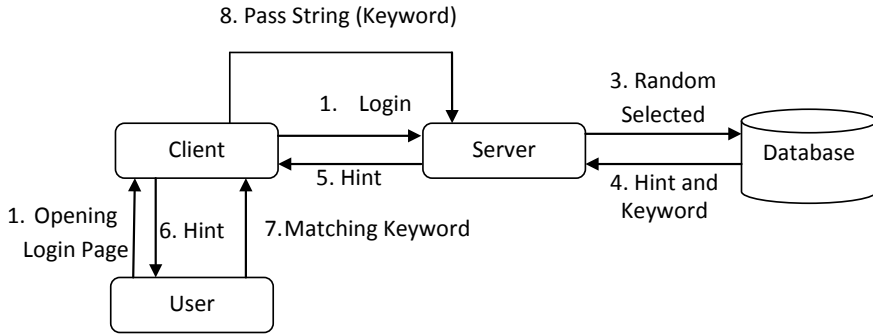


Fig. 1 Login process with random selection of hint

on the server side to retrieve the required pass string from the pool of pass strings in connected database or dataset. The hint will be given to the user by a program, which is selected based on the conditions designed in Fig. 1. Now, the end user wants to give the pass string, which is matching the hint.

Bag of Password Algorithm

- Step 1 The end user will open the login page
- Step 2 Login request made from the client
- Step 3 Server generating random id value which is associated with the image store in database
- Step 4 Server retrieved the random image (hint) and the associated keyword form the server
- Step 5 The retrieved image (hint) will be provided to the client
- Step 6 The end user will provide the matching string and which will be sent to the server for verification
- Step 7 If the verification got succeed then, the user permitted to his dashboard or else navigated back to login page.

Example: If there are 100 pass strings are stored in servers database for a single user and the same 100 strings user want to remember with its hint match. This is very much in the structure of {key, value} pairs. This makes burden on user to remember 100 {key, value} pairs like that.

In this bag of password approach, we need to minimize the burden on user to remember and recall the matching pass input/string. To do it so, hint is the key term to make the job easier to user. Better hints can do the job better here. The hint may be of following type as in Table 1,

From Table 1, it is clear using image as hint will be better option. Image can be used in two ways.

- (i) Clicking or dragging a particular part of image
- (ii) Typing keyword of the image.

Table 1 Hint type and issues behind

Hint type	User recall complexity	Server-side storage space	Other issues
Text	High	Low	Nil
Image	Low	Medium	Nil
Video	Medium	Very high	Time-consuming authentication
Color	Low	Medium	Restricted to number of items (clear separation of colors)

The first option, doing something on image may be attractive but having following problems,

- (i) Easy viewable to others near by
- (ii) Lesser number of permutations only there to break that.

The second option will be much better and having greater stability.

Bag of Password Approach with Software to Software Authentication: Not recommended in all Scenarios

Assume the scenario where there is large amount of data in cloud where the possibility of hacking is on the higher side, and the local security is much better on the places we go and on the device we access. This scenario will occur when the user is trying to have common memory space in cloud which can be accessible from his various devices and places.

In the taken diagram as in Fig. 2, the image can be downloaded to match in local database and retrieve the keyword which can be given to cloud environment for authentication. In this case, for the improvement of security, we can go for thousands of images in the pool/bag.

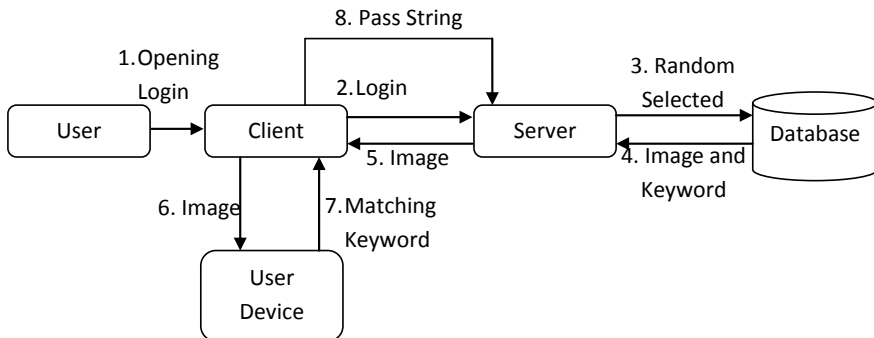


Fig. 2 Bag of password approach with software-to-software authentication

User Device: This is the personal user device where the user can store the thousands of images and the corresponding keyword. This device will be fed by the image, and according to that, it will produce the keyword back which will be fed by the user into the login page so that the authentication will be made. This is a blended authentication mechanism with personal hardware device not connected in Internet and authentication in Internet connected environment.

For other scenarios, this method will not be suitable because of following reasons,

- (i) Temporary dependency of few devices
- (ii) If the software got pirated then entire security is gone
- (iii) Setting up will required additional software.

Bag of Password Approach with Server Storage Reduction

In an organization among employees or among a medium-sized structured group, this type of approach is possible. Each and every employee or person will have his/her own cluster which holds half or less than half of his/her photographs, and the remaining should be linked with his neighbor photographs, for example, a group photograph can be linked with 20 members. There will be separate table to store the photograph id links for each person, which holds entire photograph link ids associated with him/her as shown in Fig. 3. The dataset fields associated are shown in Fig. 4.

By this way, we can able to save minimum amount of memory space to store photographs based on the similarity of the photographs present in each cluster. The similarity among the different cluster photographs will get varied for every situation,

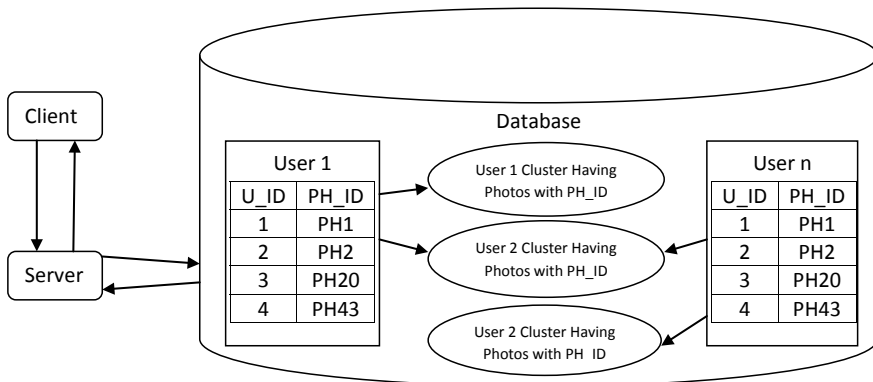


Fig. 3 Bag of password approach with server storage reduction

Where U_ID → User ID and PH_ID → Photo ID

PH_ID	PHOTO	PASS STRING
-------	-------	-------------

Fig. 4 Structure of user cluster

and thus it will be difficult to achieve a particular value but the system can be recommended, where the similarity index between the cluster crosses 10% of its memory size. This will make considerable drop in memory consumption.

Where U_ID → User ID and PH_ID → Photo ID

4 Mathematical Analysis on Bag of Password Approach

The pictures will be used to provide authentication security in this case. The primary prerequisite for employing this strategy is for the end user to remember the password during the authentication process. If some of the photographs are not utilized or are only used infrequently, it will make recalling a specific image keyword more difficult for the user. As a result, it is critical to offer balanced distribution throughout the available pictures in a timely manner. It is also critical to focus on the least number of iterations required to display all of the graphics in the login page.

Pool Limit

This bag of password technique should be implemented by considering the complexity in recalling the keywords based on the images shown. When the images are higher, the recalling complexity is also higher.

By considering the above user restrictions over the bag of password approach, we are not recommending the pool size of more than 100 images. For the test case, we are going to test with 10 images for the reason to write complete result and analysis here.

Probability Approach

We know that, we had selected 10 images for the test. Now we need to find the probability of getting all the images at least one time with minimum number of random selection over the images.

Consider the first iteration; whatever image gets selected randomly that image is the new one on the list. So the probability is as follows,

$$\text{Probability of selecting first new image} = (10/10) * 100 = 100\%$$

Similarly for the second iteration, the probability to select new image is from the slot of 9 images out of 10. So the probability is as follows,

Probability of selecting second new image = $(9/10) * 100 = 90\%$

Probability of selecting third new image = $(9/10) * 100 = 90\%$

Probability of selecting fourth new image = $(8/10) * 100 = 80\%$

.

.

Probability of selecting tenth new image = $(1/10) * 100 = 10\%$

The number of times the selection needs to be done is higher when the probability is lesser, for example to select the tenth new image, the probability is 10% that means 10 times the selection need to be done to achieve this (i.e., the $10/1 = 10 \rightarrow$ reverse of probability without multiplying with 100)

The overall probability of selecting all the images at least once is as follows,

Overall probability of selecting all the images at least once with minimum random selection (P)

$$P = \frac{10}{10} + \frac{10}{9} + \frac{10}{8} + \frac{10}{7} + \frac{10}{6} + \frac{10}{5} + \frac{10}{4} + \frac{10}{3} + \frac{10}{2} + 10$$

$$P = 29.28 \text{ selection}$$

5 Results and Discussion

The bag of password approach is been proven by the mathematical analysis and as a result we can say 30 iterations are required to have all the images (in the taken 10 sample images) at least once to the end user as a hint. It is very much important to achieve the standard deviation over image distribution and probability of getting all the images at least once with respect to authentication process. The better value in the said things will ensure the user remembrance, which is very important by considering the dynamic password entry and the number of passwords are increasing.

Comparison with other Authentication Techniques

When we go for authentication technique like biometric-based [11, 17, 18], iris recognition-based and face recognition-based are device associated which leads to additional cost in comparison with proposed bag of password approach. The above compared techniques also had issues on the flexibility of location because of the additional device need to be moved for login where ever we need.

The multifactor user authentication [10, 12, 19] leads to more complex approach than the proposed bag of password technique due to the additional steps involved in authentication and relied on the network signal strength as well.

When we come up with sensorless authentication mechanism, the direct text-based password technique [7] with some mix of alphabet, special characters, and

numbers can be broken by one short using eavesdropping technique. The password can be known and it can be reused.

When it comes to graphic-based approach which leads dynamic password entry, it provides more security than direct text-based password entry.

In graphical password approach, the color-based approach [16], pattern unlock [14], and spin wheel [15]-based approach have been taken for comparison.

The color-based approach [16] provides different options to choose different color combination as a password but it is very similar to direct text-based password approach except graphical representation. The password entry is more of static nature which in turn less secured against eavesdropping.

The pattern unlock [14] is having graphical approach but it is static in nature, and the same pattern is been used for multiple time logging into the device or Web portal. It is also difficult to draw with mouse pointers which reduces the speed and more exposed to others than proposed bag of password approach.

The spin wheel [15]-based approach is similar to our bag of password approach but the spin wheel will also have some text or image which is having limited space to display hint. It restricts end user to use smaller things which can fit in the spin wheel, and also it reduces the image uniform distribution which produces easy remembrance of hint and password combination to the end user.

The behavioral identification approach [13] can also lead to risk of restricting original user due to over prediction and allowing third part or hackers due to under prediction.

The proposed method clearly outbeaten other techniques by not having any issues based on Table 2.

Table 2 Issues related to different authentication approaches

Approaches/characteristics	Additional or specific device leads to additional cost	Forced slow presentation leads to security leak	Static nature leads to security leak	Dynamic nature leads to remembrance issue	Issue of over or under prediction
Biometric recognition	Yes	No	NA	NA	Yes
Face recognition	Yes	No	NA	NA	Yes
Iris recognition	Yes	No	NA	NA	Yes
Pattern recognition	Partially	Partially	Yes	NA	NA
Direct text-based password	No	No	Yes	NA	NA
Fuzzy-based behavioral prediction	No	No	No	No	Yes
Color-based password	No	Partially	Yes	NA	NA
Spin wheel approach	No	No	No	Yes	NA
Bag of password approach	No	No	No	No	NA

6 Conclusion

The image-based keyword (password) recall is completely based on the user capacity or ability to do that. Even though, it is user ability, when the user is not getting particular image for long time, then the possibility of forgetting that images keyword which is supposed to enter in the login page as input string in password block for authentication is on the higher side. Due to this reason, we are giving bigger importance to the number of images in the pool to select randomly. The above approach providing the value for probability of getting all the images (total images considered is 10) at least once is 29.28 selections (i.e., 30 selections), which is will take up higher number of selections when we go for more images. Similarly for 20 images, the over all probability of selecting all the images at least once with minimum random selection is 71.939 (i.e., 72 selection). Here, the point to note is the ratio of number of images with respect to selection required. For 10 images, the ratio is 1:3, for 20 images, the ratio is 1:3.59. When the number of images is being increasing, we are getting higher ratio which indicates more number of selection required for getting all images at least once. So, it has been recommended to have a pool size not more than 20 and not less than 10.

The proposed technique has been compared with other trending 8 start-of-the-art techniques to found that the bag of password approach is more efficient in providing the security by means of causing less security issues.

References

1. J.V. Chandra, N. Challa, S.K. Pasupuletti, Authentication and Authorization Mechanism for Cloud Security
2. R.M. Balajee, H. Mohapatra, K. Venkatesh, A comparative study on efficient cloud security, services, simulators, load balancing, resource scheduling and storage mechanisms. *IOP Conf. Ser. Mater. Sci. Eng.* **1070**(1), 012053 (2021)
3. B. Srikanth, M.S. Kumar, J.V. Ravindra, K.H. Kishore, The enhancement of security measures in advanced encryption standard using double precision floating point multiplication model. *Trans. Emerging Telecommun. Technol.* **31**(12), e3948 (2020)
4. K.A. Reddy, Network security using notable cryptographic algorithm for IoT data. *Int. J.* **8**(5) (2020)
5. G.S. Prasad, D.L. Praneetha, S. Srivalli, B.V. Sukesh, Information security in cloud by using enhanced triple-DES encryption algorithm. *Int. J. Innov. Technol. Exploring Eng.* **8**(5), 679–682 (2019)
6. L. Voleti, R.M. Balajee, S.K. Vallepu, K. Bayoju, D. Srinivas, A secure image steganography using improved LSB technique and Vigenere cipher algorithm, in *2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)* (IEEE, 2021 Mar 25), pp. 1005–1010
7. A.R. Phanindra, V.B. Narasimha, C.V. PhaniKrishna, A review on application security management using web application security standards, in *Software Engineering* (Springer, Singapore, 2019), pp. 477–486
8. B.S. Alladi, S. Prasad, Big data life cycle: security issues, challenges, threat and security model. *Int. J. Eng. Technol.* **7**(1.3), 100–103 (2018)

9. N. Srinivasu, O.S. Priyanka, M. Prudhvi, G. Meghana, Multilevel classification of security threats in cloud computing. *Int. J. Eng. Technol. (UAE)*. **7**(1.5), 253–257 (2018)
10. M.K. Rao, S.G. Santhi, M.A. Hussain, Multi factor user authentication mechanism using internet of things, in *Proceedings of the Third International Conference on Advanced Informatics for Computing Research*, 2019 Jun 15, pp. 1–5
11. K.R. Ramya, B.M. Josephine, K.D. Praveen, M.B. Maruthi, C.S. Kumar, An efficient and secured biometric authentication for IoT. *Int. J. Emerging Trends Eng. Res.* **7**(11), 604–609 (2019)
12. S. Nalajala, B. Moukthika, M. Kaivalya, K. Samyuktha, N.L. Pratap, Data Security in cloud computing using three-factor authentication, in *International Conference on Communication, Computing and Electronics Systems* (Springer, Singapore, 2020), pp. 343–354
13. A. Roy, S. Razia, N. Parveen, A.S. Rao, S.R. Nayak, R.C. Poonia, Fuzzy rule based intelligent system for user authentication based on user behaviour. *J. Discr. Math. Sci. Cryptogr.* **23**(2), 409–417 (2020)
14. G.K. Chaitanya, K. Raja Sekhar, Verification of pattern unlock and gait behavioural authentication through a machine learning approach. *Int. J. Intell. Unmanned Syst.* 2021
15. M.K. Rao, S.G. Santhi, M.A. Hussain, Spin wheel based graphical password authentication resistant to peeping attack. *Int. J. Eng. Technol.* **7**(2.7), 984–987 (2018)
16. P. Saranya, S. Sharavanan, R. Vijai, R.M. Balajee, Authentication scheme for session passwords using color and image. *Int. J. Smart Sensing Intell. Syst.* **15**, 10 (2017)
17. A. Tarannum, Z.U. Rahman, L.K. Rao, T. Srinivasulu, A. Lay-Ekuakille, An efficient multimodal biometric sensing and authentication framework for distributed applications. *IEEE Sens. J.* **20**(24), 15014–15025 (2020)
18. P. Yellamma, P.S. Rajesh, V.V. Pradeep, Y.B. Manishankar, Privacy preserving biometric authentication and identification in cloud computing. *Int. J. Adv. Sci. Technol.* **29**, 3087–3096 (2020)
19. S. Komatineni, G. Lingala, Secured E-voting system using two-factor biometric authentication, in *2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC)* (IEEE, 2020 Mar 11), pp. 245–248
20. K. Neeraja, P.R. Rao, D.S. Maloji, D.M. Hussain, Implementation of security system for bank using open CV and RFID. *Int. J. Eng. Technol.* **7**(2–7), 187 (2018)
21. R.C. Radhika, M.R. Narasinga Rao, S. Venkateswarlu, Review on the security issues in human sensor networks for healthcare applications. *Int. J. Eng. Technol.* **7**, 269–274 (2018)
22. P.L. Kumari, G.S. Sekhar, Key exchange and E-mail authentication using lagrange interpolation, in *Data Engineering and Communication Technology* (Springer, Singapore, 2020), pp. 253–264

Automatic Content Creation Mechanism and Rearranging Technique to Improve Cloud Storage Space



R. M. Balajee, M. K. Jayanthi Kannan, and V. Murali Mohan

Abstract The storage of electronic data as well as the demand for it has become a major problem in today's society. Data is becoming increasingly centralized in order to provide the flexibility to use it anywhere, at any time, and on any device. Due to the increasing mobility in modern devices, data productivity and accessibility in cloud storage are increasing. Data versatility is expanding on a daily basis, posing a management challenge. All methods of dumping data regularly over a period of time necessitated the deletion and rearrangement of a few data items in order to achieve greater efficiency in the data retrieval process. Currently, the researchers are focusing on the efficient searching algorithms and not on the combined technique of data prioritization, deletion, and rearrangement. The proposed automatic content creation mechanism (AACM) system will create new document after deleting unwanted contents and by merging few existing documents based on the top key words. Each and every document is associated with particular keywords. The proposed system leads to two outputs by considering the text, first to form core points with voting count and then to create new documentary on it. The proposed system can also focus on video, audio, and image in addition to text but however the major focus is given to text, which is the complex one of the four. The mechanism will move from lower priority/older one to higher priority/newer one on the basis of success rate with a particular cluster. This mechanism will save the valid information even from lower priority and older documents. It will also free up the space by deleting the unwanted sentences from older files, and all these depend on the threshold (confidence) value, which is auto-adjusted by the proposed mechanism on the basis of success rate. This will lead to a better memory management and prevention of core historical data.

Keywords Data prioritization · Data deletion · Data rearrangement · Automatic content creation mechanism · Saving valid information · Confidence value adjustment

R. M. Balajee (✉) · M. K. Jayanthi Kannan · V. Murali Mohan
Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation,
Vaddeswaram, Guntur, Andhra Pradesh, India

M. K. Jayanthi Kannan
Department of Computer Science and Engineering, JAIN (Deemed to be University), Bangalore,
India

1 Introduction

It has become more difficult to manage data storage on the cloud as it has grown in size. With the use of the Internet, the solution to the problem should be created as an efficient (highly accurate) one [1, 2]. Since the storage is not local, all data transactions performed by any user, administrator, or anyone else may incur high costs. In initial period, the raw data will be in cloud and the usage of those data for processing has been done locally by the corporate system or individual laptop but today the software, operating system, and laboratory setup are happening through Internet. This results in massive traffic in network where the users of cloud transfer their data. The cloud computing is not an independent domain because it is associated with network traffic, topologies used by the network, route selection by the node, growth of big data, service required by the user from cloud, the level of security with encryption [3, 4] and decryption technique used, authentication mechanism, searching algorithms, etc.

The usage of data is generally depends on software and the operating system which handles it. The software and the operating systems are also getting changed itself to a newer version for every time period past away. Certain technological advancements cause change in supportability of file on software and operating systems. This raises the question of data reliability and its accessibility over a long time period. Still these challenges are not addressed properly by researchers as well. The devices which use these data are also depend on software for its work and most importantly the revolution in technological device (smarter devices with touch screen and voice input for processing) raises the question of supportability in new devices with different input mechanism.

The cloud computing technology is the one which got introduced as a sequence of technological development from Web services, and in the way it crosses grid, network, and utility computing. It is in the view to the world as a fifth appearance after four successive technological advancements [5].

Due to common people's use of the Internet, it has grown popular even in villages. Since it has established wireless connectivity, it is critical to secure the network from hackers. In general, the attackers are having more advantage in wireless environment than wire connectivity network. Here, cryptography plays an important role to prevent the hacker getting into the network.

2 Literature Survey

One of the foremost characteristics of cloud computing is elasticity, and it is due to the expansion of storage space availability to the user on demand [6]. The local computer hard disks are difficult to maintain the backup regularly (periodic operation required lots of monitoring, storage spaces, man power, etc.) but on the cloud, the backup of data is taken periodically by the admin itself. Any failures or wrong data

transaction can roll back any time when the user wants it to do. This also leads to additional storage space.

The cloud storage is based on pay to use concept. The common people who had desktop system on their home will have storage in size of terabyte for their usage. This will not happen in cloud that easily because for every megabyte we are consuming, we need to pay for it but it offers flexibility in terms of location accessibility, device accessibility, and time based accessibility [7]. This is the reason why it cannot be wasted with garbage of data, and it should be maintained well to save space on cloud. Many compression algorithms and auto-data deletion algorithms are came into the art of cloud computing.

The data storage in the cloud can be optimized by doing the deduplication [8, 9] over the selected storage, and hence the data which are present twice or more than that can be removed to have exactly single instance of it, but this will only remove the data's which are exactly redundant and not the similar data which are exist.

The deduplication algorithms applied already in the research are also checking over for the duplication in the document level or at the block level of data [10]. The deduplication can be focused in the level of small statements to filter out any extra content and that is not made yet.

When we are speaking about the cloud, then there is a big question of resource management [11, 12]. The resource can also be storage as well. The focus of our research is on the storage sector of the cloud. This resource allocation can be efficiently allocated by considering the availability of resources and the demand for that. The primary goal is to use the resources as maximum as possible so that we can reduce the ideal or not utilizing the resources to increase the profitability of provider and increase the service to clients.

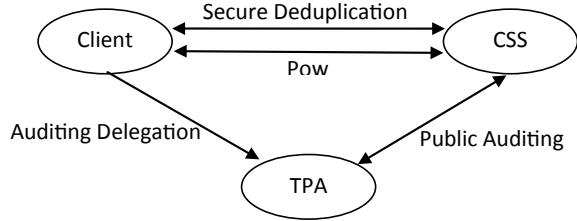
The storage resources can be allotted to different users and that cannot be switched between users like other computing resources [13, 14]. So these storage spaced resources need to be allocated with much more concern than the importance provided to other resources.

The cloud data is been exposed to more security issues, since it is having the nature of wireless connectivity. These security issues can be properly addressed by the encryption techniques [15, 16]. The encryption can come into part only after authentication, if the authentication mechanism [17, 18] fails, then the hacker is getting the possibility to hack the data and that is where the encryption techniques play a role. The authentication can be used in hiding the sensitive data based on the identity [19] provided on the login time. The authorization can be restricted to access only the general data, and the hidden information will be authorized to access for the specific users as per their privileges.

The cloud storage can be hacked by the phishing-based attacks [20], and those data hacked on this type of attack are more sensible one. This can be prevented by identifying the node in wireless environment which launches this phishing attack and removing or restricting the access of that particular node. The detection of the hacker node can be done by observing the behavior of the nodes in network.

There are some more ways like public auditing [21] and usage of virtual machines [22] which improved the cloud storage security. The public auditing can look for the

Fig. 1 TWA-ECS architecture diagram



deviation in result over the period of time, and based on the deviation observed, the corrective measures will be taken. The virtual machines on the cloud will provide security for the data in means of hiding the physical location of the server. When considering the virtual machines, there is flexibility to switch the actual servers and that will not be noticed by the end user or hacked who are trying to access the data.

Take-Young Youn, KU-Young Chang, Kyung Hyune Rhee, and Sang Uk Shin are the doctorate holders from Korean university and they published the research paper in IEEE about efficient cloud storage. They introduced the concept to reduce the wasted space in cloud [23].

The motive of the researchers here is to do two things,

- (i) To perform secure deduplication of encrypted data.
- (ii) To perform public integrity auditing of data.

From here on we will call it as two-way-approach of efficient cloud storage (TWA-ECS). The TWA-ECS performs challenge response protocols using the BLS signature-based homomorphic linear authenticator. We utilize a third-party auditor for performing public audit, in order to help low-powered clients. The TWA-ECS approach is shown in Fig. 1.

The TWA-ECS consists of the following entities.

- (i) Client/user
- (ii) Cloud storage service (CSS)
- (iii) Third-party auditor (TPA).

3 Problem Description

The problem description found after the literature survey is pointed out below,

- (i) The cloud storage is increasing because of our day-to-day personal and work activities. Due to increase in storage size, it is very difficult to retrieve the required data from it.
- (ii) The researcher today focusing about the efficient retrieving algorithm to overcome the issue. Any retrieving algorithm will work efficiently with lesser storage and in presence of more relevant data.
- (iii) There are also techniques to delete the unwanted files from cloud to free up the memory space but the historical data are lost for the future.

- (iv) The TWA-ECS technique is only focusing on removing duplicate or repeating files. It is not focusing about the removing technique of older files.
- (v) To overcome all these things, we need to free up the space filled by older/non-accessed data and also to prevent the value of the data as well.

4 Proposed System

The cloud is the great storage space to store user's data, so that it can be retrieved at any time, any place, and any device. The accessibility of cloud storage space is handled using browser through Web application. When the users are dumping the files in cloud, then there will be demand for cloud storage space. This demand can be handled by considering two things. First, the storage space can be increased but it results in spending more cost by the user as well as service provider. Second, the files in the cloud can be deleted, so that it reduces the demand for additional memory required in cloud. The problem with the second method is to select the files to delete. If I am deleting the older files, then it may also be required by the user further. If I am deleting the non-accessed files, then it may also be the newer one and it required time to get top of the search list.

The cloud space that I am considering is the public cloud, and any user can upload the data and any user can search for the data. Finally, the solution is to choose the file which is older and non-accessed file.

Even though the proposed mechanism chosen the file to be deleted and if the file is deleted, then the content of that file cannot be accessed anymore. The deleted file may contain many paragraph, audio, and video contents as well. Few may be important and may be required in future as well. These few contents need to be identified, and those contents need to be stored separately. While filtering those contents, the files need to join together with common key word. Further reduction can also be done with taking the key sentence and providing the votes for the key sentence. The key points focused by the proposed mechanism is listed below,

- (i) The older/non-accessed contents in the cloud need to be deleted and at the same time core data of any document need to be prevented.
- (ii) The core content of the document is of lesser important then it also need to be deleted from the cloud.
- (iii) It is important to say, there should not be any data which is permanently present in cloud and there should not be any data with good value (historical data) should be deleted.
- (iv) The focus is on to develop automatic content creation mechanism (ACCM) for handling the cloud storage efficiently.
- (v) Here all documents, paragraphs, videos, and images are assigned with separate ID to handle and identifying it. This is done while uploading the document to the cloud.

Table 1 Documents considered as a dataset

Document id	Key words	User priority	Data importance	Access frequency	Date of storage
1	Sports Cricket Sachin	7/10	7/10	5/10	11/02/2021
2	Sports Cricket Dhoni	6/10	7/10	8/10	16/02/2021
24	Sports Football XXX	7/10	5/10	6/10	12/02/2021
25	Sports Tennis YYY	8/10	7/10	6/10	14/02/2021

The ACCM mechanism works with the following parameters, (i) document id, (ii) content id, (iii) key words, (iv) user priority, (v) data importance, (vi) access frequency, and (vii) date of storage.

To understanding ACCM mechanism, we consider 25 documents handled by the proposed mechanism is listed in Table 1.

The ACCM will produce two different types of output and with artificial intelligence; it will grow up to touch the documents with higher usage. It will grow confidently with respect to success rate of it over a period of time. The outputs of the mechanism are.

- (i) Automatic document creation
- (ii) Historical data creation.

Initially, the proposed mechanism needs to choose the file to do its process. Let we saw how it will choose the file. The file choosing process will go through two-dimensional way. Let we put these into two algorithms namely.

- (i) 1D file choose algorithm
- (ii) 2D file choose algorithm.

1D File Choose Algorithm

- Step 1 Set $n = 2$ and Choose file with the Access Frequency Value $< n$.
- Step 2 Apply the second filter with time period older than 1 month.
- Step 3 Find Average = (User Priority Value + (Data Importance Value * 2))/3 and chose files with Average $< n$. If the document is the system created one then Average = Data Importance.
- Step 4 Applying ACCM to form two outputs and let the new document will not be applied with any mechanism until a month.
- Step 5 Every new document will be checked after a month of time.

Table 2 Keywords of the documents

Document 1	Document 2	Document 3	Document 4
Sports Cricket Sachin	Sports Cricket Dhoni	Sports Football Xxx	Sports Tennis Yyy

If ((Access Frequency Value $> n$) & (Data Importance Value $> n$), then increase n as $n = n + 1$ for all $n < 4$;

If ((Access Frequency Value $\leq n$) & (Data Importance Value $\leq n$), then decrease n as $n = n - 1$ for all $n > 2$;

Step 6 The above steps will be repeated for every hour.

2D File Choose Algorithm

Step 1 Choose the documents with least number of similar key words.

Step 2 Applying ACCM to form two outputs and let the new document will not be applied with any mechanism until a month.

Step 3 If recursive 1D algorithm is applied on the file then choose documents with greater number of similar key words with in the exponential growth of previous number of key words.

Understanding the 2D File choosing algorithm

Assume the following key words for the different document as shown in Table 2.

- (i) First we will consider the least keyword \rightarrow Cricket, Documents 1 and 2 is taken for the process.
- (ii) Second iteration, we will calculate the range value as range = (previous keyword * 2). Range = $2 * 2 = 4$.

We will consider the documents with maximum number of similar key words within the range calculated. Keyword \rightarrow Sports, Documents 1, 2, 3, and 4 is taken for the process.

Advantage

- (i) Prevent the historical data with voting count.
- (ii) Minimize the demand of additional storage in cloud environment.
- (iii) Increase the accessibility of each file.

5 Result and Analysis

The mechanism is tested with the sample of 25 documents, and in that the proposed algorithm found five similar documents which are related to sports to apply itself for the reduction of memory taken by the documents on the disk. The deduplication of documents can also be done by the proposed algorithms on the basis of removing the

internal similar statements. The TWA-ECS algorithm is also applied over the same 25 documents to reduce the memory space and the final result of both the proposed and TWA-ECS algorithm is been observed and plotted as a graph. The document set contains five documents which are duplicated out of total 25 documents. The duplicated documents are in the category of “history.” The “sports”-based documents are taken for the discussion here to examine the step-by-step result obtainment on those by applying the proposed algorithms. The said five documents can be given with the ids d1, d2, ..., d5 for the discussion purpose. The taken documents are text-based documents. The statements in each document are compared with statements with other documents to see the similarity over the documents. The similar documents are taken for the further process where some of the statements are deleted based on multiple criteria, and the new document with proper rearrangement of existing and surviving statements will be formed with lesser memory space. The older five documents will be deleted form the cloud.

One of the comparisons made between the statement s1 from document d1 and statement s2 from document d2 is shown in Table 3,

Voting for Statements

Among similar statements, only one should be selected for writing into the new file. The statement selected should have higher number of votes, and if two statements are having same number of votes then the bigger statement will be selected. This approach is due the fact; bigger statements can convey the information fully with some details. If two or more statements are having same number votes and length, then random statements among the selected group will be selected for the new document.

Figure 2 shows about the votes scored by different sets and statements of those sets. Every statement in a particular set will be given same number of votes as the set having. Few statements are common in two or more number of sets, and those statements will be merged in new sets and the existing sets will be deleted. If a set is having three elements and in that two statements are common with other two different sets, then those two statements are taken out to form two new sets with the matched elements. The third element which did not found any match will be deleted from the process.

The grouping of sets is happening in sequential manner. After the current grouping process over, it will again start from the number it left. Initially the mechanism found match between set 6 and 7, so these two sets are merged together with new set number 13. The set 6 and 7 will be removed from the group, and set 13 added in the group as well. Similarly, set 14 is formed. These newly formed sets are shown in Fig. 3.

Table 3 Comparison of statements in documents

	Total words	Similar words	Percentage of similar words	Result—matching similarity
S1	22	16	72	Max (72, 100) = 100
S2	16	16	100	

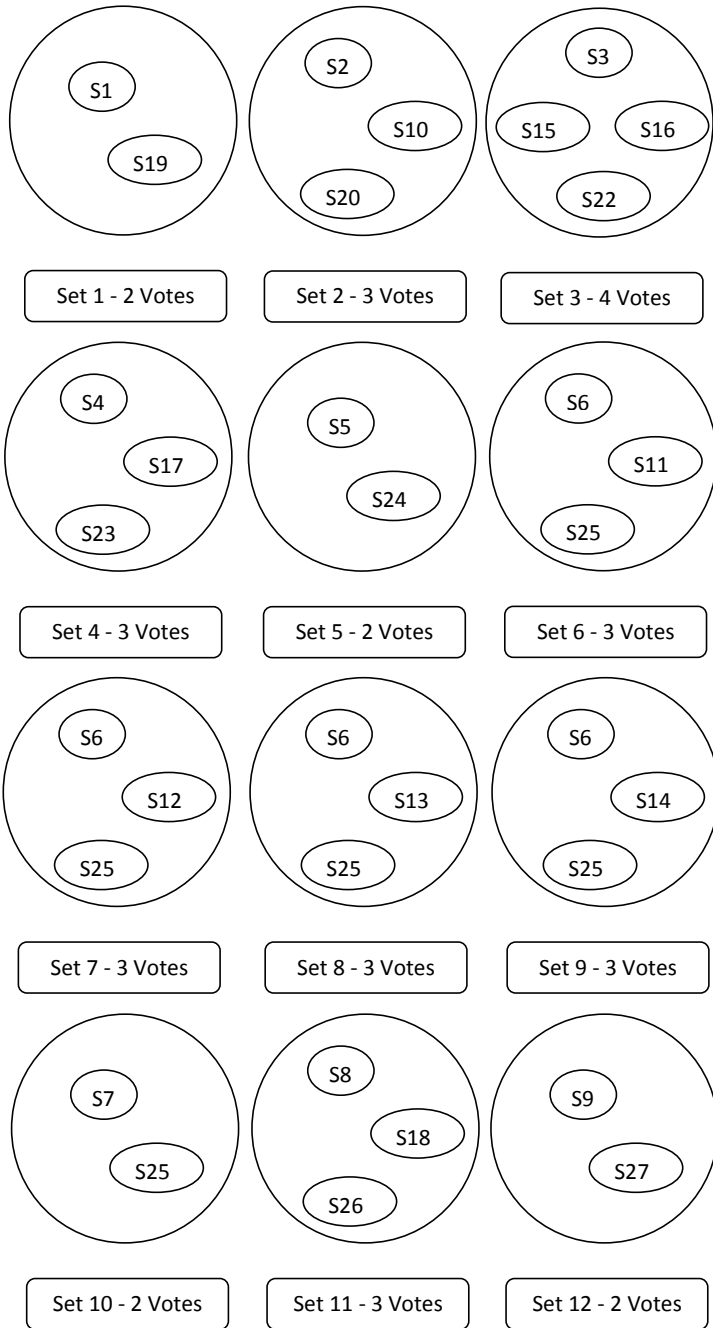
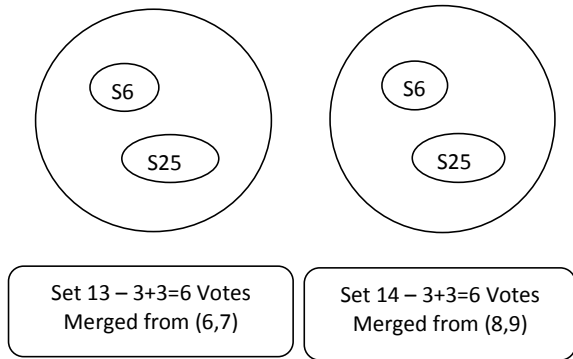


Fig. 2 Votes scored by various sets and statement among them

Fig. 3 New set 13 from set 6 and 7, new set 14 from set 8 and 9



The new set 14 is added to the group. Now the process starts from set 10, again the set 10 is having statement id of S7 and S25. The statement id S7 is not matching with any of the further sets in group, and so it has been removed from the group. Now the statement id S25 got merged with set 13 in the group (the set 13 is newly formed group on combination of another two sets). The set 13 is having statement id S6 and S25. The statement ID S6 got its pair with set 14. So the statement id of S6 from set 13 is merged with set 14. Now set 14 is having statement ids of S6 and S25. It is clear that S6 got already matched, and so the pair needed to be checked for S25. Since it a combination group, the pair may be within the group itself, and it need to be checked further more in sequential order. After checking the sequential order, the temporary groups are formed. Then these temporary groups are involved in checking process, and permanent group has been created here.

During the process of creating a temporary groups as in Fig. 4, the similarity table will also be created as shown in Table 4, because of that similarity only, the statements ids can be merged together.

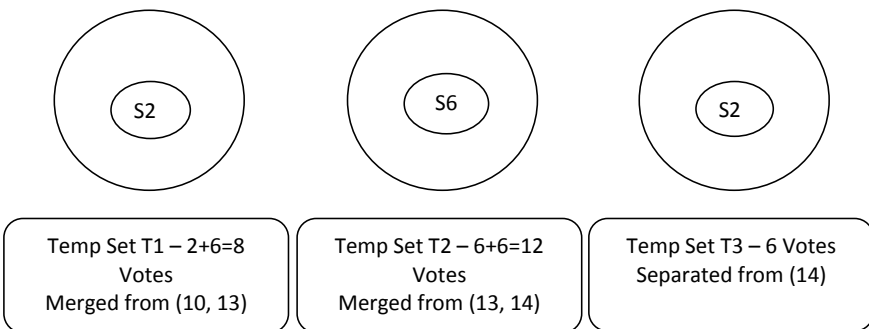


Fig. 4 Temporary sets are formed in between the process of forming permanent similarity identification

Table 4 Similarity mapping of statement ids

Set id	Statement id	Similarity statement ids
Set 10	S7	S25
Set 10	S25	S7
Set 13	S6	S25
Set 13	S25	S6
Set 14	S6	S25
Set 14	S25	S6

The above table is having redundancy on mapping the similarity, so these redundancies need to be avoided and the redundancy avoided result is shown in Table 5.

Now the permanent set has been formed after checking the similarities, the statement ids of S25 and S6 can be treated as a single group elements, due to the similarity mapping. The statement ID S6 also not having matches with other group elements and at the same time statement ID S25 is with 14 votes (dominating the similarity group and taking lead to next formation).

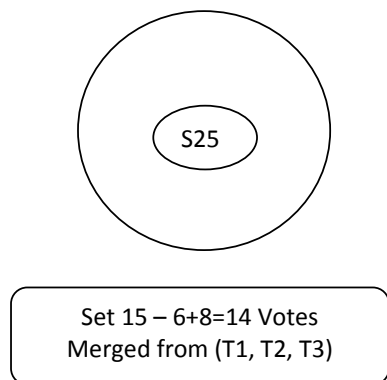
As a result, new set 15 has been formed as shown in Fig. 5 by merging three temporary sets T1, T2, and T3. The statement id S25 is ending with 14 votes. The final arrangement of sets and sequence change is shown in Fig. 6.

Each set is having one or more statements ids, The statement ids belong to same group are similar in nature, so each set should be lead with one statement ID and the remaining need to be removed.

Table 5 Removed redundancy from similarity mapping of statement ids

Set id	Statement id	Similarity statement ids
Set 10	S7	S25
Set 13	S6	S25

Fig. 5 New set 15 from set T1, T2, and T3



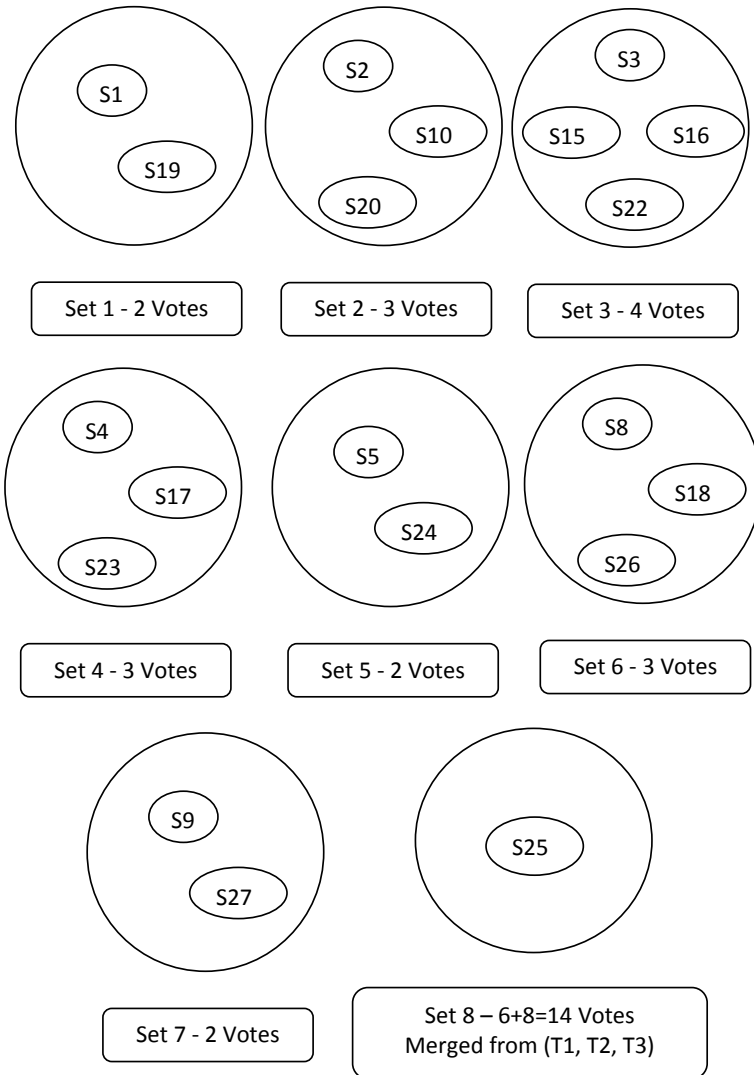


Fig. 6 Result group arrangements with votes of statement ids

The statement ids in the set are having same number of votes and so the lead statement ID should be selected with the help of the statement word count. Higher the word count will yield higher the priority to lead the set.

Formation of Resultant Document

After selection of statements, the statement will be arranged in such a way those older document statements are first and newer document statements are last. It will

Table 6 Inputs taken and considerations

Input/considerations	Value	Initial memory occupation (kb)
Total document	25	2124
Sports-based document	5	378
Historical documents	5	412
Other documents in a mix of domains	15	1334
Duplicate documents	4 (all historical docs)	330
Considered file format	.txt	–
Cloud environment	Google Cloud	–
Cloud-Based service	IaaS/PaaS	–

also show the number of votes which reflects the weightage of statements given by various documents. It prevents value of data as well as minimizes the memory.

The inputs taken and considerations for evaluation of algorithms in real-time implementation are been given in Table 6. The achieved results are also replicated in the calculations for easier understanding.

Total memory of 25 documents = 2124 kb.

Total memory occupied by these 5 + 5 documents = 378(sports) + 412(history) = 790 kb.

Total memory occupied by the new document = 39 kb (sports) + 42 kb (history) = 71 kb.

Memory saved = 2124 kb – 1405 kb = 719 kb.

Percentage of savings = 33.85%

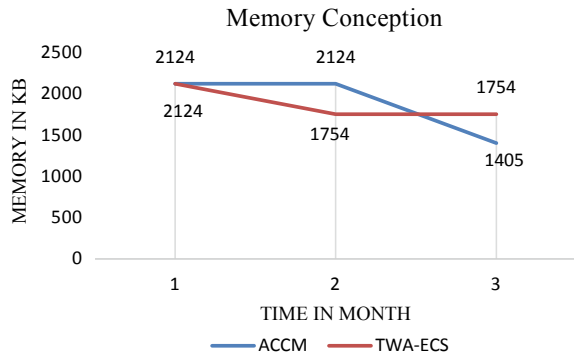
If more number of documents got merged, then the probability to get similar statements between them is higher. If similar statements are more, then it results in more weightage of statements (votes) and the same level of deletion of statements. When the statement deletion operation increases, there memory occupation will be decreased. Hence, it results in better percentage of savings.

The TWA-ECS technique has been applied on the same 25 documents to get the result of memory reduction and both proposed ACCM algorithms, and TWA-ECS algorithm has been plotted in a graph to compare the results. The comparison of results obtained by both the techniques is shown in Fig. 7.

6 Conclusion

The objective of reducing cloud storage space by deleting the lesser important content on the similar document groups to form the newer document with the proposed algorithm is properly implemented and shown that 33.85% of memory efficiency. This clearly states that the proposed algorithm is nearly 34% saving the memory space in cloud storage which is higher than the existing algorithm's memory savings

Fig. 7 Comparison graph of memory consumption



of nearly 18% in the cloud storage. The algorithm also designed in such a way, after showing this much of reduction in cloud storage, it will get feedback from the end user, and according to the feedback level, the confidence of the algorithm is been adjusted by the adjustment of threshold percentage value to obtain the similarity of statements among different documents. This induced the artificial intelligent to the algorithm according to the confidence level it gained form the end user. All these show the efficiency and ability of the proposed algorithm has been on top notch in all the circumstances.

References

1. S. Karimunnisa, V. Kompalli, Cloud computing: Review on recent research progress and issues. *Int. J. Adv. Trends Comput. Sci. Eng.* **8**(2), 216–223 (2019)
2. W. Haoxiang, S. Smys, MC-SVM based work flow preparation in cloud with named entity identification. *J. Soft Comput. Paradigm (JSCP)* **2**(02), 130–139 (2020)
3. N. Vurukonda, R.B. Thirumala, DC-MAABE: data centric multi-authority attribute based encryption on cloud storage. *J. Comput. Theor. Nanosci.* **16**(5–6), 1893–1901 (2019)
4. A. Bashar, Sensor cloud based architecture with efficient data computation and security implantation for Internet of Things application. *J. ISMAC* **2**(02), 96–105 (2020)
5. R.M. Balajee, H. Mohapatra, K. Venkatesh, Comparative study on efficient cloud security, services, simulators, load balancing, resource scheduling and storage mechanisms. *IOP Conf. Ser. Mater. Sci. Eng.* **1070**(1), 012053 (2021)
6. G. Sreeram, M.K. Kanumuri, M. Bodduluri, Improving cloud data storage performance based on calculating score using data transfer rate between the internetwork drives. *Int. J. Eng. Adv. Technol.* **8**(4), 1830–1835 (2019)
7. S. Myla, S.T. Marella, K. Karthikeya, B. Preetham, S.K. Hasane Ahammad, The rise of big data in the field of cloud analytics. *Int. J. Emerging Technol.* **10**(4), 125–130 (2019)
8. N. Srinivasu, B. Yashaswi, Dynamic user management for secure cloud deduplication using enhanced checksum approach. *Int. J. Innov. Technol. Explor. Eng.* **8**(7), 1585–1588 (2019)
9. R. Kaur, I. Chana, J. Bhattacharya, Data deduplication techniques for efficient cloud storage management: a systematic review. *J. Supercomput.* **74**(5), 2035–2085 (2018)
10. K. Ravindranath, Y.S.V. Balaji, B.M.M. Kumar, C.H. Chowdary, An efficient cloud storage management optimal with deduplication. *Int. J. Innov. Technol. Explor. Eng.* **8**(6), 54–58 (2019)

11. A.K. Bashir, R. Arul, S. Basheer, G. Raja, R. Jayaraman, N.M. Qureshi, An optimal multitier resource allocation of cloud RAN in 5G using machine learning. *Trans. Emerging Telecommun. Technol.* **30**(8), e3627 (2019)
12. K.A. Kumari, J.K. Sastry, K.R. Rao, Energy efficient load balanced optimal resource allocation scheme for cloud environment. *Int. J. Recent Technol. Eng. (IJRTE)* **8**(1S3) (2019)
13. A.S. Kumar, M. Venkatesan, Multi-objective task scheduling using hybrid genetic-ant colony optimization algorithm in cloud environment. *Wireless Pers. Commun.* **107**(4), 1835–1848 (2019)
14. S.P. Praveen, K.T. Rao, An effective multi-faceted cost model for auto-scaling of servers in cloud, in *Smart Intelligent Computing and Applications* (Springer, Singapore, 2019), pp. 591–601
15. S. Sindhura, S.P. Praveen, S. Syedbi, V.K. Pratap, T.B. Krishna, An effective secure storage of data in cloud using ISSE encryption technique. *Ann. Romanian Soc. Cell Biol.* **1**, 5321–5329 (2021)
16. L. Voleti, R.M. Balajee, S.K. Vallepu, K. Bayoju, D. Srinivas, A secure image steganography using improved LSB technique and Vigenere cipher algorithm, in *2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)* (IEEE, 2021 Mar 25), pp. 1005–1010
17. N. Sunanda, N. Sriyuktha, P.S. Sankar, Revocable identity based encryption for secure data storage in cloud. *Int. J. Innov. Technol. Explor. Eng.* **8**(7), 382–678 (2019)
18. P. Saranya, S. Sharavanan, R. Vijai, R.M. Balajee, Authentication scheme for session passwords using color and image. *Int. J. Smart Sensing Intell. Syst.* **15**, 10 (2017)
19. N. Bhuvaneswari, M. Trinathbasu, M. Srisathvik, R.K. Tenali, Identity based security auditing for data sharing with sensitive information hiding using cloud storage. *Int. J. Innov. Technol. Explor. Eng.* **8**(6), 1327–1333 (2019)
20. C. Thirumallai, M.S. Mekala, V. Perumal, P. Rizwan, A.H. Gandomi, Machine learning inspired phishing detection (pd) for efficient classification and secure storage distribution (ssd) for cloud-IoT application, in *2020 IEEE Symposium Series on Computational Intelligence (SSCI)* (IEEE, 2020 Dec 1), pp. 202–210
21. S. Chaudhari, S.K. Pathuri, A comprehensive survey on public auditing for secure cloud storage. *Int. J. Eng. Technol.* **7**(2.7), 565–569 (2018)
22. N. Chandrakala, B.T. Rao, Migration of virtual machine to improve the security of cloud computing. *Int. J. Electr. Comput. Eng.* **8**(1), 210 (2018)
23. T.Y. Youn, K.Y. Chang, K.H. Rhee, S.U. Shin, Efficient client-side deduplication of encrypted data with public auditing in cloud storage. *IEEE Access* **15**(6), 26578–26587 (2018)

Voter ID Card and Fingerprint-Based E-voting System



Rajesh Kannan Megalingam, Gaurav Rudravaram, Vijay Kumar Devisetty, Deepika Asandi, Sai Smaran Kotaprolu, and Vamsy Vivek Gedela

Abstract Voting is a fundamental right given to every citizen of a democratic country, with a minimum age requirement set by the respective countries. As such, one would expect the procedure for voting to be on the cutting edge of technology in terms of security and adhere to the highest standards. This paper proposes and discusses a method of E-voting based on dual-factor authentication in the form of unique identification (UID) number and the fingerprint of the voter for verification purposes. An algorithm for fingerprint recognition is also discussed in the paper along with the efficiency of the algorithm in CPUs of different computing powers. We created a website with the proper focus on securing the personal data of the constituents while also making it legible for the election officials to keep track of the progress of the election and avoid dual/multiple vote casting. The additional security provided by biometric authentication ensures that the system we propose meets the safety standard set by the Information Technology Act, 2000. Based on the experiments and results, we believe that the proposed anti-fraud E-voting system can bring confidence in voters that their vote is secured.

R. K. Megalingam (✉) · G. Rudravaram · V. K. Devisetty · D. Asandi · S. S. Kotaprolu
Department of Electronics and Communication Engineering, Amrita Vishwa Vidyapeetham,
Amritapuri, India
e-mail: rajeshm@am.amrita.edu

G. Rudravaram
e-mail: gauravrudravaram@am.students.amrita.edu

V. K. Devisetty
e-mail: devisettyvijaykumar@am.students.amrita.edu

D. Asandi
e-mail: asaideepika@am.students.amrita.edu

S. S. Kotaprolu
e-mail: ksaismaran@am.students.amrita.edu

V. V. Gedela
Department of EE, University of Cincinnati, Cincinnati, OH, USA
e-mail: gedelavv@mail.uc.edu

Keywords Electronic voting machine (EVM) · Fingerprint recognition · Unique identification (UID) · Hashing · Secure voting · Authentication · Biometric matching system (BMS)

1 Introduction

The system of E-voting has been introduced in many countries and has been in use to date by most of these countries. E-voting systems have been observed to have more advantages than the traditional paper-based system in aspects such as the counting of votes time period, and the mistakes in counting votes can be drastically reduced using the E-voting system. The problem the E-voting system lacks is transparency and false voting. The problem of being non-transparent can be rectified by using voter-verified paper ballots (VVPBs); this device prints the vote that has been cast and displays it to the voter, and through this, the vote cast can be verified. In order to prevent malpractices like false voting (voting using the other citizen's data) can be reduced by introducing a dual-factor authentication system to verify the voter before the vote has been cast.

AADHAR card consists of a 12 digit number that is unique to each Indian citizen or passport holder of India. It even consists of information, biometric and demographic data of the respective residents of India. The Unique Identification Authority of India (UIDAI) is responsible for collecting the data. The UIDAI was established by the government of India in January 2009. This research work proposes a voting system based on UID card and fingerprint authentication to enable a safe and transparent election process. While this process of using the AADHAR card is relevant only in India, the same method of storing the citizen's biometrics in a government database and linking to a unique identification card can be applied in other countries as well to imitate similar results.

Biometric security methods are proven to be more secure than the use of one-time password (OTP)-based or PIN-based security methods. The biometric security methods include fingerprint scanning, where a person is identified based on his/her fingerprints. As the fingerprints are unique to each individual, only the respective person can access the system whose fingerprint matches the one in the database. The proposed system practices dual-factor authentication, that is, the verification of the UID Card and the fingerprint of the voter, and does not allow a constituent to cast more than one vote because it automatically updates the status of the vote of a constituent on a dynamically rendered website created by us.

Our proposed system authenticates a voter based on the UID number and the voter's fingerprints. If the authentication at any of the levels fails (the UID number or the fingerprints won't match with any of the data present in the pre-enrolled database), then the voter will be denied access to voting. This process can prevent other people from casting false votes, as the voter needs to scan their fingers to pass the second level of authentication. Once a voter passes both the levels of security, then the voter

can vote. Then, the vote is updated in the database and will not allow a second vote, hence preventing a person from voting more than one time.

The website we designed can be personalized by the government or third-party contractors to be made to enlighten the average citizen on the process and security of the voting system. The website is also connected to a database and renders a complete list of the constituents and their status of voting, both of which are encrypted in a local database by hashing the data in it to secure them from the prying eyes. Hashing is used to map data of any size to a fixed length. This is called a hash value. Encryption is a two-way function whereas hashing is a one-way function. While it is technically possible to reverse-hash something, the computing power required makes it unfeasible. The election official can access this list by using the credentials issued to him/her, and it is up to their discretion whether or not to make this list public. In this way, the proposed system in this research paper ensures a safe and reliable election procedure with the utmost transparency so as not to raise any questions about rigged elections or false votes.

2 Problem Statement

The centerpiece of any successful democracy is the peaceful transition of power from one government to another. And this is only made possible by the process of voting. As the most powerful non-violent tool a citizen has in a democracy, the process of voting has to be without any flaws. There should not be any questions about the safety or reliability of the voting system in the mind of any citizen or constituent. While technology has greatly impacted every aspect of our life, the process of voting has, unfortunately, not undergone many changes since it was first proposed. There are 167 democratic countries in the world out of which around 34 countries have some sort of electronic voting system. Among these countries, only India, despite its population of 1.36 billion, has a 100% electronic voting system. When dealing with large democracies like India, the efficiency of EVMs during the voting process becomes extremely important. The 2019 parliamentary elections in India had a voter turnout of 67%, nearly 900 million registered voters across 542 parliamentary constituencies came out to vote. Some countries still rely on ballots to carry forward the voting procedure. The ballot system has many disadvantages such as long waiting lines, questions about authenticity of the ballots due to the votes not being transparent enough as there is no way for an individual to be sure the candidate they voted for received the vote with absolute certainty. To avoid questions about authenticity and safety of the voting process and to ensure the safety of the election officials, we present a new idea for implementing the process of voting.

3 Related Works

Paper [1] proposes a system which uses a biometric authentication system and additionally provides a facility to users to cast a vote using mobile phones. The first way is for the smartphone users, one-time password (OTP) is used. For other users who do not have smartphones, biometric methods such as fingerprint recognition is used as authentication. The main objective is to cast a vote from anywhere anytime. A large population of people is not aware of using smartphones even though they have one and it will be a huge task to educate voters. In the voter ID card and fingerprint-Based E-voting system, a person is allowed to vote if the fingerprint of a person is matched with his/her fingerprint stored in the database. No other electronic devices are needed for the voter to vote. A system which takes complete control over child vaccination status and monitors child vaccination schedules is proposed in paper [2]. It uses the fingerprint of the child from the database and performs fingerprint processing and classifies it. In e-vaccination, the fingerprints of the infants are taken, the accuracy is hard to verify. This drawback can be rectified by considering the fingerprints of their parents/guardians and applying the same system as discussed in this paper. Paper [3] takes measures to allow the voter to vote only if the voter logs into the system by using the right credentials which are generated by merging the two sets of credentials in the form of black and white dotted images generated by the computer and encrypted using a video cryptography scheme. Accessibility or voter education—Many people are not aware of how to use an email and how to log in and this gives rise to hackers. In the voter ID card and fingerprint-based E-voting system, there is no need for the voter to log into a system, only the voter's fingerprint is required. Paper [4] proposes a multifaceted online e-voting system. The requirements embedded in the design of the respective system permits well-secured authentication processes for the voter using combined simple biometrics. There are possible attacks such as: replay attacks, denial-of-service and session hi-jack which can be minimized by the use of the system proposed in this paper which uses a cloud-based database encrypted through salting and hashing.

A voting system based on an advanced RISC machines (ARM) processor and fingerprint sensor is proposed in the research paper [5]. The authors have used a simple liquid crystal display (LCD) display for user interface and a keypad for entering the information about the voters. This suffers the drawback of limited constituent enrolling due to limitations of the sensor memory. The research paper [6] give us an idea on how to proceed with the fingerprint recognition by providing an in depth explanation about two different image processing algorithms namely, scale invariant feature transform (SIFT) algorithm which is used for local feature matching and the fast library for approximate nearest neighbors (FLANN) which is applied to match the query image and reference image in dataset. In paper [7], the authors have implemented a secured platform for remote health monitoring services. They have used a Raspberry Pi to keep track of the health data. For security purposes, they came up with a password authentication key exchange mechanism based on hashing and zero-knowledge password proof. Paper [8] proposes a women security system

based on GPS modules and foregoes the need of a smartphone instead opting for a completely portable system based on microcontroller. The system communicates via Wi-Fi and transmits data in real time.

Paper [9] proposes a voting mechanism which identifies the voter based on his/her fingerprint image taken from a fingerprint sensor. In this model, encryption of the database is not done. In our model, the database encryption is done by hashing and salting. Paper [10] proposes a dual authentication one using iris recognition and the other is comparing fingerprints. The comparison technique used for iris recognition is hamming distance and for comparing fingerprints is Euclidean distance. This technique is not optimum for voting because the time taken for dual authentication is higher. In paper [11], the authors propose a model aimed at retrieving images from the database based on the context in the given image. The major steps involved are object recognition and image retrieval. Object recognition includes training phase and is done using SIFT, SURF (speed-up robust features), HOG (histogram of oriented gradient), and color Histogram. In image retrieval, similar images are selected based on rank of similarity. Paper [12] proposes a method in which voter has to place his/her voter ID which has a unique radio-frequency identification (RFID) tag. If the tag matches from the one in database, then the voter must verify his/her fingerprint. If the fingerprint verification is successful, the voter is eligible to cast his/her vote.

Paper [13] focuses more on the implementation of the EVMs, making the voting process more transparent. It also introduces the term 'Voting Status Flag' which plays a role in the authentication. In order to prevent malpractices during the election process, Kerberos (computer network protocol) is used. Our proposed system uses a security system that verifies the voter before casting the vote. We have even designed a website which improves the user interface. The EVM proposed in Paper [14] is built using LPC2148 which is the core of the ARM-7 processor. Here in the first stage of authentication, the card given is scanned using RFID and later the fingerprints are taken. Using RFID tags might become expensive and less secure while dealing with a large population, and considering this factor, our system uses a fingerprint sensor as the second stage, which is more secure than using RFID tags. In Paper [15], the authors explained about the fingerprint recognition algorithms. They explained the enrollment, matching, and extracting phases in the fingerprint algorithm. In our proposed model, the run-time of the fingerprint recognition algorithm can be reduced as we will be using 1:1 matching, using the unique identity number of each voter. Paper [16] gives more insight into fingerprint recognition. Biometric authentication systems has two modes, enrollment and recognition. The identity of the fingerprint is based on invariance and singularity. This paper discusses the process of recognition of fingerprints using image preprocessing, feature extraction stages etc.

Paper [17] states the advantages of using biometric methods such as fingerprint recognition as a mode of authentication. It even gives information about the method (Chaotic Arnold transform) used for recognizing the fingerprints. Our proposed system has a dual-security system where we take the unique identification number of a voter first, then the fingerprint of the individual will be compared with the one from the database with the ID entered initially. This will reduce the run-time of the algorithm. Paper [18] deals with the transparency of the E-voting system and

the advantages of E-voting. It provides various methods such as using voter-verified paper ballots (VVPBs) which helps the voter to verify the vote casted. In our proposed system, we introduced a biometric authentication level (fingerprint sensor) which can reduce malpractices such as false voting and multiple voting.

4 System Architecture

In the system we propose as shown in Fig. 1, the constituent first has to enter his voter ID number (the number assigned to him/her by the government) into a keypad connected to a processor. We have tested two processors for this project, namely Broadcom BCM2711 and Intel Core i7-9750H. The processor then verifies the number with the list of UID numbers stored in the external database and finds the related fingerprint based on this unique identification number in the form of voter ID. After successfully finding the number and verifying its status of voting, the fingerprint module is activated and asks the constituents to verify their fingerprint with the one stored in its database.

We have written a fingerprint algorithm which verifies whether or not they are matching and sends a message to the microcontroller to activate the ballot box if the fingerprints are matching. Once the microcontroller is activated after the verification process is completed, the constituent can cast their vote with the help of the ballot box. Once the constituent selects their candidate, the ballot box immediately deactivates and informs the microcontroller. The microcontroller in turn sends this information back to the processor which updates its database for preventing the constituent from voting again. The updated database is rendered in real time in a website under secure authentication which will be handled by the government either by establishing secure

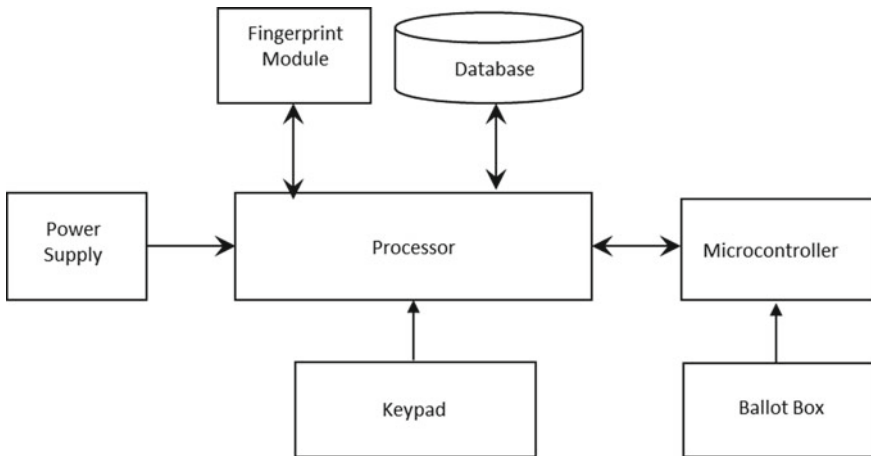


Fig. 1 System architecture

servers for handling the data influx or by hosting the website on a trusted third-party server system. This system of using a duality of processor and microcontroller can be made simpler by using a micro-processor such as Raspberry Pi, but it was decided against that due to drawbacks which we will discuss later in the paper.

5 Design and Implementation

5.1 System Flow Architecture

Figure 2 shows the control flow of the system from the point of input entry to the point of termination. There are two levels of authorization to be passed in order for the voter to cast his/her vote. The first authorization level includes the verification of the voter ID which can be inputted through a keypad. Once this level is cleared, the fingerprint module accepts the fingerprint of the voter. The verification of the

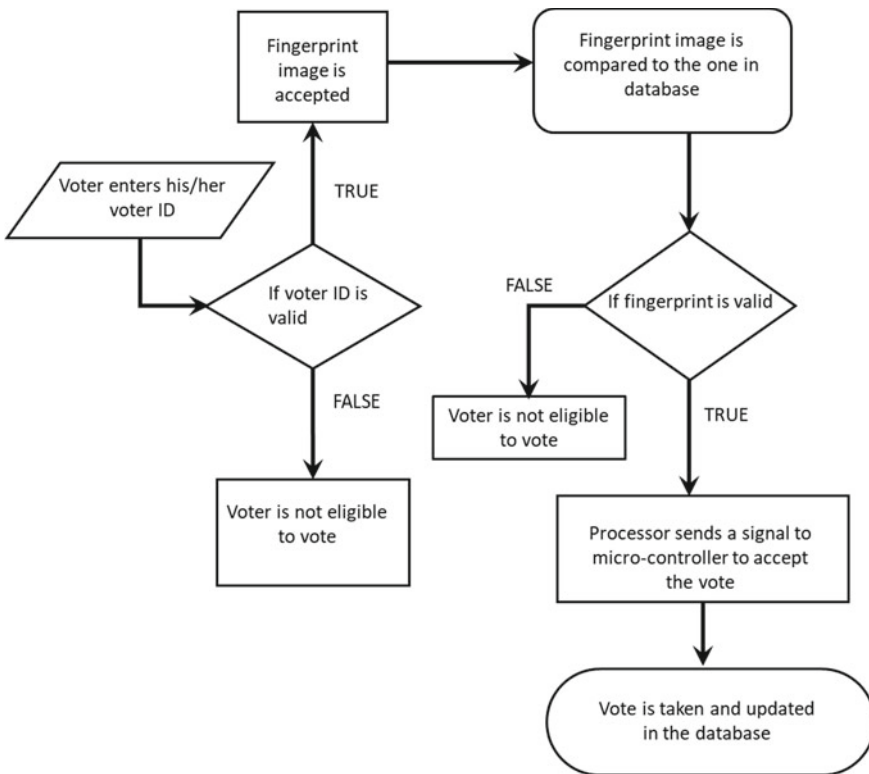


Fig. 2 System flow architecture

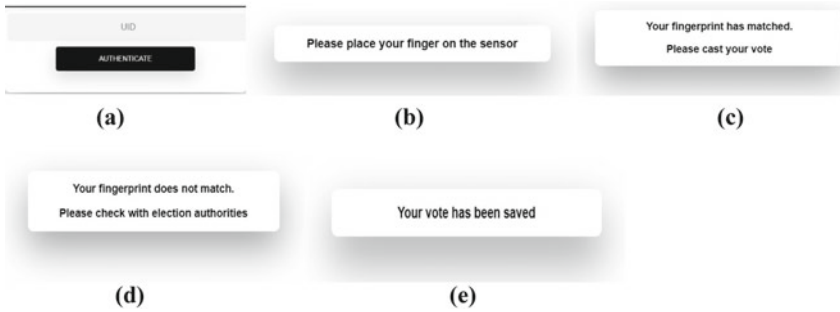


Fig. 3 User interface with respect to system flow outcomes

voter’s fingerprint comes under the second level of authorization. After the voter is successfully verified to be legitimate through both their voter ID and fingerprint, a signal is sent to the microcontroller which accepts the vote and sends the input back to the processor to be updated in the database.

Figure 3a is the main page where the voter has to enter their voter ID. If the voter ID is valid, then Fig. 3b is rendered in the webpage prompting the voter to give their fingerprint as the input through the fingerprint sensor. When the voter fingerprint image is taken, it is compared to the one in database. If the fingerprint is matched Fig. 3c is rendered in the webpage, if the fingerprint does not match Fig. 3d is rendered in the webpage. After the vote is cast, Fig. 3e is rendered and the vote is added to the database.

5.2 Hardware Used

Processor For the experimentation process, we have used two processors. We will discuss the advantage and disadvantage of both in the Experimentation and Results section.

- i. *Broadcom BCM2711*. This is the processor used in Raspberry Pi 4 and uses a 1.5 GHz 64-bit quad-core Arm Cortex-A72 CPU.
- ii. Intel Core i7-9750H. This is a 6 core, 12 threaded CPU with a base frequency of 2.6 GHz and Max Turbo frequency of 4.5 GHz.

Fingerprint-Sensor R307-TTL UART The R307 Fingerprint Module is a sensor that is used to scan fingerprints. It even includes transistor–transistor logic (TTL) and universal asynchronous receiver transmitter (UART) interfaces. The fingerprint data can be stored in the module and can be configured as 1:1 mode or 1: N mode for authentication. 1:1 matching is when a person uses either ‘Card + Fingerprint’ or ‘User ID + Password’ mode of authentication. Initially, the data (such as User ID) is entered, and once the data with the respective ID is found, it is matched

with the second input (such as the fingerprint). 1: N authentication is more user-friendly as no specification is needed. In this method of authentication, using the data entered by a person (like a fingerprint), one template from a list of up to a thousand pre-enrolled templates is picked. A 3.3 V or 5 V microcontroller can be directly interfaced using the fingerprint module. In order to create an interface with a PC serial port, a level converter like MAX232 is needed. R307 Fingerprint Module consists of a high-performance fingerprint alignment algorithm and a high-speed DSP processor. It even has other hardware which facilitates its performance, image processing, template storage, and other functions. Since the fingerprint sensor R3-07 is a module with a TTL UART interface, it is connected with a USB-TTL UART module to communicate with the computer. Interfacing with fingerprint sensor is achieved by using `pyfingerprint` library; this library allows the sensor to interface with Raspberry Pi and other Linux-operated machines and send data to them.

Microcontroller Arduino Uno is a development board based on the ATmega328p microcontroller. It has 20 general purpose input/output pins (GPIO), out of which 14 are digital input/output pins (6 pins are used as PWM output pins) and the remaining pins are 6 analog input pins. It includes a USB connection to program the microcontroller, a power jack, an in-circuit serial programming (ICSP) and a 16 MHz ceramic resonator. Arduino is connected to the computer via USB, and it is interfaced using `pyserial` library in python; this allows the processor to send or receive data from Arduino through a serial port.

Hardware requirements The hardware requires no GPU. It is recommended to have a minimum memory of 2 GB and CPU of base frequency 1.5 GHz.

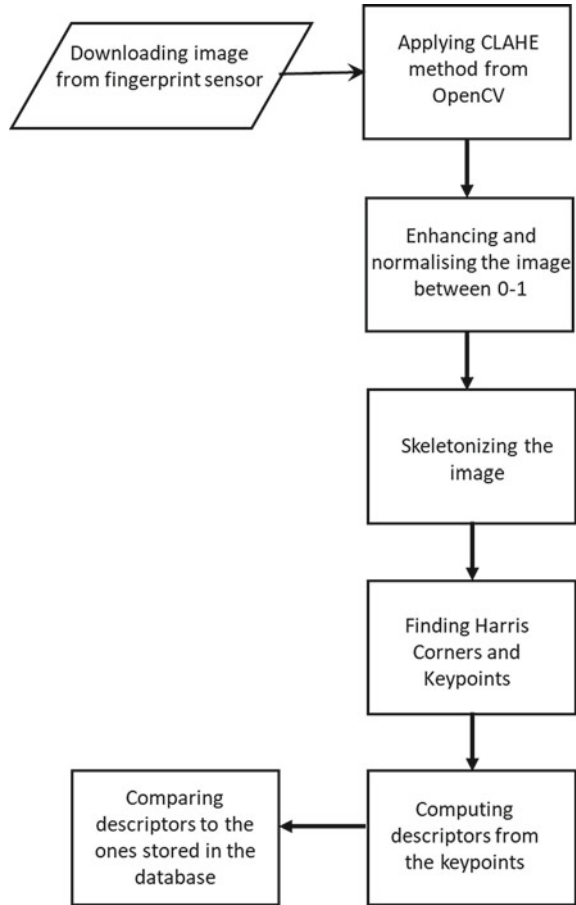
Interfacing the Various Components The fingerprint sensor R3-07 is a module with a TTL UART interface, and it is connected with a USB-TTL UART module to communicate with the computer. Interfacing with fingerprint sensor is achieved by using `pyfingerprint` library. This library allows the sensor to interface with Raspberry Pi and other Linux-operated machines and send data to them. Arduino is connected to the computer via USB, and it is interfaced using `pyserial` library in python; this allows the processor to send or receive data from Arduino through a serial port. The hardware requires no GPU. It is recommended to have a minimum of 2 GB RAM.

5.3 Fingerprint-Recognition Algorithm

As we can see from Fig. 4, after downloading the fingerprint image from using the inbuilt library inside the sensor R3-07 fingerprint sensor, we store it in a separate database.

The fingerprint algorithm is explained in the following steps. We apply the contrast-limited adaptive equalization (CLAHE) algorithm from the `opencv2` library in python. CLAHE is a variant of adaptive histogram equalization which takes care

Fig. 4 Algorithm for fingerprint recognition



of over-amplification of contrast. It will operate by selecting small regions of an image, the respective region is called tiles. The surrounding tiles are later combined using bilinear interpolation to remove artificial boundaries. Applying CLAHE to our original images equalizes it and improves its contrast. We then proceed to enhance the image. The image enhancement part of the algorithm draws heavy influence from [17] and improves the overall accuracy of the algorithm.

This enhanced image is further normalized and skeletonized. Skeletonization reduces the image into 1-pixel wide representation. In 1-pixel wide representation, the value of the pixel is either 0 or 1 indicating whether the pixel corresponds to the foreground of the image or background of the image. This is useful for feature extraction. The next part includes extracting corners and inner features of the image using the Harris Corners method. The final part of the algorithm includes extracting the key points of the image and computing their descriptors using oriented fast and rotated (ORB) which is a fast-robust feature detector in the cv2 library.

5.4 Software Used

Fingerprint-Recognition For the fingerprint recognition algorithm, we used Python because of its ease of coding as well as fast prototyping. Python is open source and can be integrated with Web frameworks easily which is very important for this project.

Backend We have used Node.js for handling all the data and event listeners since its processing speed is very fast as compared to other Web frameworks as it is an event-based model. Non-blocking input/output and asynchronous request handling which is a unique feature of Node.js makes it capable of processing requests without any delays.

Database As for the database we have picked MongoDB because it is very easy to integrate with Node.js. It is based on document-oriented storage and handles big data much more efficiently.

Simulation Environment The proposed model has been simulated in a Linux environment. We used the following python libraries: Numpy—1.17.3, OpenCV—4.5.1.48, Scikit image—0.18.1, Scipy—1.6.0, Pandas—1.22, pyfingerprint—1.5 while developing the fingerprint comparing algorithm.

6 Experiment and Results

6.1 Testing the Images for Fingerprint Recognition

All the experiments mentioned below were conducted in lighting conditions comparable to natural conditions and are subject to change based on the sensor used. The algorithm is performed on several types of images, original image, gray scaled image, and enhanced image. The results were accurate when enhanced image is used. So we applied the image enhancement algorithm proposed in [17] which can improve the clarity of ridge and furrow structures based on the local ridge orientation and ridge frequency estimated from the input images. The algorithm also recognizes the unrecognizable corrupted regions and removes them from further processing. Figure 5a, c represent the original fingerprint and Fig. 5b, d represent their enhanced images respectively. An important point to be mentioned here is that because the R3-07 is an optical-based fingerprint sensor, the pressure on the surface of the sensor is a crucial factor in ensuring that the image is stored properly and can be verified in detail. After the image enhancement part was successful, we moved on to comparing the fingerprint images without much difficulty. While it is true that each fingerprint is unique, there are similarities between any two pairs of fingerprints in the manner of the whorls or ridges that make up the fingerprint.

Figure 6 shows that the number of descriptors (descriptors describe the elementary characteristics of an image such as shape, color, and texture) matching when the same

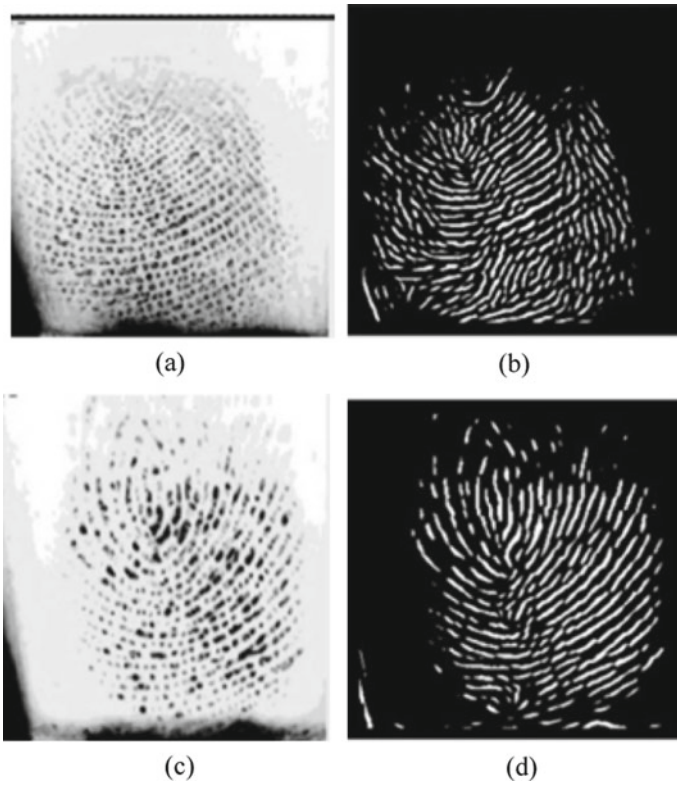


Fig. 5 Comparison of original image versus enhanced image

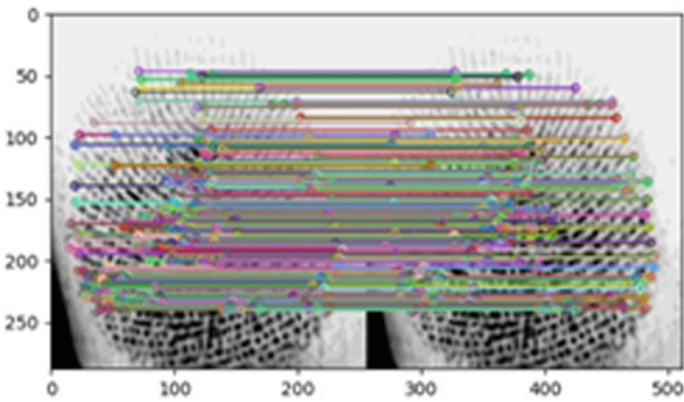


Fig. 6 Comparing the same fingerprint

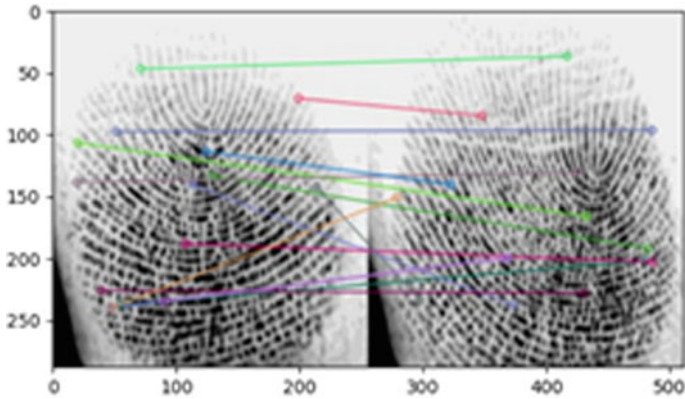


Fig. 7 Comparing different fingerprints

fingerprint is provided as input is very high. Figure 7 shows that although the number of descriptors matching for different fingerprints is very low, it is not necessarily zero. Therefore, in order to account for the small number of matching descriptors in unique fingerprints, we tried out the algorithm extensively on many images and set a threshold of ‘33’ in the algorithm which detects whether the number of matching descriptors is enough to validate the fingerprint or not.

6.2 Comparison of Different Processors

Since we are not necessarily using the inbuilt library for the sensor but instead running our own algorithm, we tested the run-time of the fingerprint recognition algorithm on two different processors, namely the Broadcom BCM2711 and the Intel Core i7-9750H.

Table 1 shows that the average run-time for comparing images in Broadcom BCM2711 processor is approximately 33.45 s, while the average time taken by Intel Core i7-9750H processor is approximately 10.26 s as shown in Table 2. Clearly, the

Table 1 Run-time in broadcom BCM2711 processor

Input	Time taken for comparison (Seconds)
Matching fingerprints set 1	34.6
Matching fingerprints set 2	32.72
Matching fingerprints set 3	35.2
Unique fingerprints set 1	31.29
Unique fingerprints set 2	34
Unique fingerprints set 3	32.9

Table 2 Run-time in Intel Core i7-9750H processor

Input	Time taken for comparison (Seconds)
Matching fingerprints set 1	10
Matching fingerprints set 2	10.8
Matching fingerprints set 3	10.8
Unique fingerprints set 1	9.8
Unique fingerprints set 2	9.8
Unique fingerprints set 3	10.1

advantage with the Intel processor is superior run-time, but it is not very cost efficient. The main advantage with the Broadcom BCM2711 processor is that it comes fitted into a Raspberry Pi model which can make the voting machine far more portable and accessible but at the major cost of efficiency and much slower run-times. While it is up to the personal discretion about which processor to use, in this work we proceed with the Intel Core i7-9750H processor for the reasons discussed earlier.

Figure 8 shows a basic model of the EVM system built for testing purposes. The fingerprint sensor R3-07 is located at the far left and connected to the Broadcom BCM2711 processor. The Arduino communicates with the processor once the fingerprint is verified and takes input from one of the push buttons. The time delay in communication with the Arduino and the server running in the background by Node.js

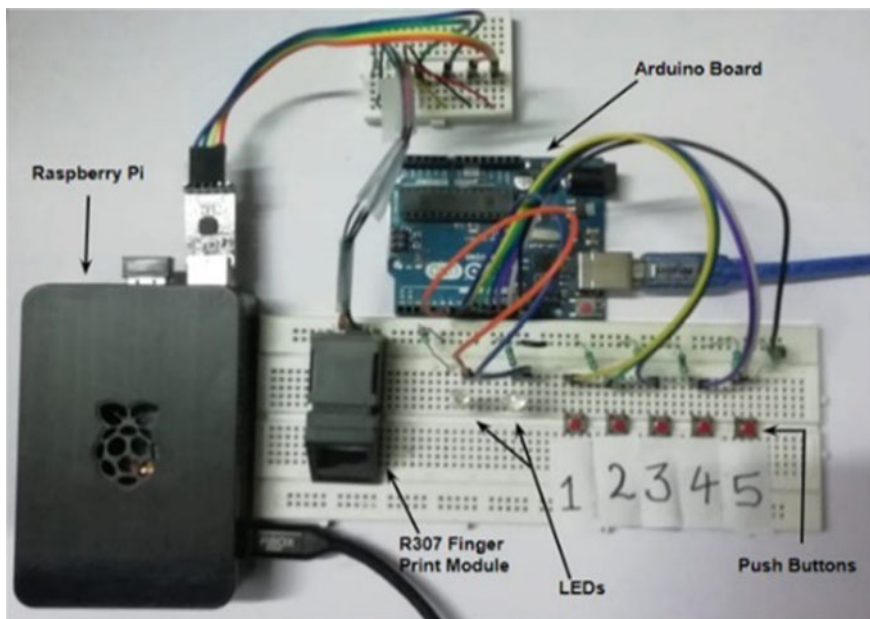


Fig. 8 Model-1 prototype of E-voting system using the broadcom BCM2711 processor

ID	Name	Gender	Status of Voting
1	voter 1	Male	Casted
2	voter 2	Male	Casted
3	voter 3	Male	Pending
4	voter 4	Male	Pending
5	voter 5	Male	Pending
6	voter 6	Male	Pending
7	voter 7	Female	Pending
8	voter 8	Female	Pending
9	voter 9	Female	Pending
10	voter 10	Female	Pending

Fig. 9 Status of votes rendered by server

is set to about 500 ms to ensure there is no data overlap. Once the Arduino detects that any of the buttons is pressed it sends a message to the server and stops communication immediately.

Salting is an apprehension that typically refers to password hashing. It is an exclusive value that can be added to the end of the password to create a different hash value. The hash value can be any random or personalized character that adds an additional layer of security to the hashing process, distinctively against brute force attack. Once the Arduino communicates with the server, the server immediately updates the database with the status of the voter and the vote cast. To prevent malpractice, the server is secured by many rounds of salting followed by hashing. The information in the database is updated in real time by the server and displayed as a list in a webpage as shown in Fig. 9 so that the election officials can keep track of who has cast their vote already in case any disputes arise.

7 Conclusion

In this work, we presented a voter ID and fingerprint-based E-voting system based on authentication in the form of voter identity card and fingerprint of the constituent. The need for such a multi-authentication system was described in the motivation section. The related works section discussed in detail about the existing research work related to secure E-voting systems and how our proposed system compares with the existing systems. We also presented the architecture of the proposed system, both hardware and the software. The process of identifying fingerprints and its results are explained in the experiments and results section along with the information in the database that is updated in real time by the server when the voter casts the vote. Based on the experiments and results, we believe that the proposed anti-fraud E-voting system can bring confidence to voters that their vote is secured.

We have some suggestions for the successful adoption of our proposed system by any agency. The website can be designed by the government through third-party developers in order to obtain a better user interface. The database can be expanded by the government in order to store data of a larger population. The fingerprint data can be obtained from the biometric data information of each resident which is stored in the government database. In the future, the system can be expanded by including the following features. The fingerprint recognition algorithm can be further optimized to reduce the run-time between acceptance and verification of the fingerprint. The prototype can be tested on better processors which can exponentially decrease the time-complexity of the entire procedure.

Acknowledgements We are gratified to express our gratitude towards the Electronics and Communication Engineering Department and HuT Labs of Amritapuri campus of Amrita Vishwa Vidyapeetham University for their ceaseless succour, without which this work would not have been progressed.

References

1. S. Patil, A. Bansal, U. Raina, V. Pujari, R. Kumar, E-mart voting system with secure data identification using cryptography, in *International Conference for Convergence of Technology (I2CT)* (2018). <https://doi.org/10.1109/I2CT.2018.8529497>
2. S. Vidhya, J. Rajiv Krishnan, B.A. Sabarish, P. Sachin, E-vaccination fingerprint based vaccination monitoring system. *Int. J. Pure Appl. Math.* **118**, 623–628 (2018)
3. S. Nisha, A. Neela Madheswari, Prevention of phishing attacks in voting system using visual cryptography, in *International Conference on Emerging Trends in Engineering, Technology and Science (ICETETS)* (2016). <https://doi.org/10.1109/ICETETS.2016.7603013>
4. M.A. Khasawneh, M. Malkawi, O. Al-Jarrah, A biometric-secure e-voting system for election processes, in *International Symposium on Mechatronics and Its Applications ISMA08* (2008). <https://doi.org/10.1109/ISMA.2008.4648818>
5. M. Venkata Rao, V.R. Ravula, P. Pala, Development of anti rigging voting system using biometrics based on aadhar card numbering. *Int. J. Sci. Eng. Adv. Technol. IJSEAT* **3**(2) (2015)
6. R.K. Megalingam, G. Sriteja, A. Kashyap, K.G.S. Apuroop, V.V. Gedala, S. Badhyopadhyay, Performance evaluation of SIFT and FLANN and HAAR cascade image processing algorithms for object identification in robotic applications. *Int. J. Pure Appl. Math.* **118**(18), 2605–2612 (2018)
7. R.K. Megalingam, K.S. Sarathkumar, V. Mahesh Kumar, A secured healthcare platform for remote health monitoring services, in *Conference: NGCT2105, IEEE International Conference on Next Generation Computing Technologies* (2015)
8. R.K. Megalingam, K. Jyothsna, T.S. Aparna, T. Anjali, M. Meera, S.D. Amruth, IoT-based women security system, in *2019 Inventive Communication and Computational Technologies, Proceedings of ICICCT* (2019)
9. M. Faheem Rana, A. Altaf, S.Z. Naseem, Enhanced real time system of E-voting using fingerprint, in *2013 International Conference on Electronics, Computer and Computation (ICECCO)* (2013), pp. 297–300. <https://doi.org/10.1109/ICECCO.2013.6718287>
10. T. Singh, Design of a dual biometric authentication system, in *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)* (2016). <https://doi.org/10.1109/ICEEOT.2016.7754806>

11. T. Bagyammal, L. Parameswaran, Context based image retrieval using image features. *Int. J. Adv. Inf. Eng. Technol. (IJAIET)* **9**(9), 27–37 (2015)
12. J. Deepika, S. Kalaiselvi, S. Mahalakshmi, S. Agnes Shifani, Smart electronic voting system based on biometric identification-survey, in *2017 Third International Conference on Science Technology Engineering and Management (ICONSTEM)* (2017), pp. 939–942. <https://doi.org/10.1109/ICONSTEM.2017.8261341>
13. R. Balaji, M.P. Muhammed Afnas, B. Praveen Kumar, V. Varun, C. Tamizhvanan, Embedded based E-voting system through fingerprint and aadhaar card verification (2019)
14. B. Madan Mohan Reddy, D. Srihari, RFID based biometric voting machine linked to aadhaar for safe and secure voting. *Int. J. Sci. Eng. Technol. Res. (IJSETR)* **4**(4) (2015)
15. M.M.H. Ali, V.H. Mahale, P. Yannawar, A.T. Gaikwad, Overview of fingerprint recognition system (2016), pp. 1344–1350. <https://doi.org/10.1109/ICEEOT.2016.7754902>
16. L. Hong, Y. Wan, A.K. Jain, Fingerprint image enhancement: algorithm and performance evaluation. *IEEE Trans. Pattern Anal. Mach. Intell.* **20**(8), 777–789 (1998). <https://doi.org/10.1109/34.709565>
17. J. Samuel Manoharan, A novel user layer cloud security model based on chaotic Arnold transformation using fingerprint biometric traits. *J. Innov. Image Process. (JIIP)* 36–51 (2021). <https://doi.org/10.36548/jiip.2021.1.004>
18. M. McGaley, J. McCarthy, Transparency and e-voting democratic versus commercial interests, in *Electronic Voting in Europe—Technology, Law, Politics and Society, Workshop of the ESF TED Programme Together with GI and OCG, July, 7th–9th* (2004), pp. 53–163

Intelligent CCTV Footage Analysis with Sound Source Separation, Object Detection and Super Resolution



Yash Khare, Abhijit Ramesh, Vishwaak Chandran, Sevagen Veerasamy, Pranjali Singh, S. Adarsh, and T. Anjali

Abstract CCTV cameras are found everywhere nowadays and are used to monitor, secure, and protect your property, or at the very least serves as intelligent CCTV footage analysis with sound source separation, object detection and super resolution. However, according to recent statistics, 80% of CCTV footage is discarded in the case of an investigation and is deemed uninformative. The reason being the grainy and low-quality video feed from CCTV cameras. Nonetheless, people thought about it and created video processing software or forensic tools that can improve the quality of the footage. Despite this, the latter are usually expensive or are only available to the authorities. Here developed is an open-source solution that is cross-platform and offers a seamless user interface for your average consumer. The application uses super-resolution to enhance image quality, object detection using YOLO v3, and sound extraction. Using actual CCTV footage as an example, the overall quality and output satisfying results for every functionality.

Keywords Super-resolution · Object detection · Sound source separation · Generative adversarial networks · UNet · Encoder-decoder architecture

Abbreviations

CCTV	Closed-circuit television
YOLO	You only look once is a state-of-the-art, real-time object detection system
PSNR	Peak signal-to-noise ratio
SSIM	Structural Similarity Index
SR-GAN	Super Resolution Generative Adversarial Network
GPU	Graphics Processing Unit
CPU	Central processing unit

Y. Khare · A. Ramesh (✉) · V. Chandran · S. Veerasamy · P. Singh · S. Adarsh · T. Anjali
Department of Computer Science and Engineering, Amrita Vishwa Vidyapeetham, Amritapuri,
India
e-mail: abhijitramesh@am.students.amrita.edu

CNN Conference on Neural Networks
GAN Generative Adversarial Network

1 Introduction

CCTV cameras are found nearly everywhere nowadays. In every street, building, or business, there will at least be one CCTV camera monitoring the surroundings and providing a sense of security. It acts as a deterrent to anyone thinking about carrying out criminal activities. Finally, it provides evidence in the case of a robbery or any unfortunate event that could happen. At the very least, that is what business owners and civilians expect from it. The reality of it, however, is very different.

Let's take the UK as an example. There are approximately 4 million cameras scattered all around the UK, and 1 million of those cameras are found in London only.

Taking the sheer number of cameras operating, it would be normal to assume that the crimes are quickly resolved. According to the metropolitan police, only one crime per year is solved for every thousand cameras in the UK. Considering that 20% of the total number of CCTV cameras in the world are found in the UK, that is not an encouraging amount.

The main reason is the low-resolution footage that is obtained from the CCTV cameras. The videos are grainy and blurry, which makes the identification of the suspect nearly impossible. To be identifiable, an individual would have to make up 100% of the video screen. Face recognition is therefore not even an option here. As such, most CCTV footage is disregarded in the case of an investigation. There are services such as MotionDSP that provide enhancement capabilities, but they all charge a fee for buying or using their software, and, in most cases, it does not qualify as cheap. MotionDSP provides a package as low as USD1609 to a maximum of USD3229 per license. The cheapest package offers simple object detection capabilities, but the most expensive one contains the video enhancement filters. In this case, the maximum fee to get our video enhanced. Such types of services are not even an option for individual users or even small business owners.

Here, the aim is to develop an open-source solution to this problem. Using techniques such as super-resolution, images can be enhanced and, as such, CCTV footage by enhancing every video frame. Object detection can also be performed using the state-of-the-art YOLO framework, which is freely available. Features such as audio extraction can easily be integrated. Encompassing all those techniques into one cross-platform application, the goal is to offer easy-to-use and accessible to all video processing applications.

2 Related Work

There have been several developments in this field where CCTV footage is enhanced. Still, there have not been many instances where object detection was done and then enhanced using a super-resolution model. Most articles focus on enhancing the entire footage rather than on specific frames which would be computationally heavy. An algorithm for the detection of events in CCTV surveillance systems based on their trajectories was proposed in tracking-based event detection for CCTV systems [1]. With the system that was proposed in the article, some predefined events, such as camouflage, scribbling's on walls, and presence of humans on staircases can be observed. The proposed method requires reduced human monitoring to fix the issue of the increasing alarms and look at the footage selected from the system. The limitation here was that only predefined events are detected, and the user does not have the freedom to enhance a corresponding frame. Also, the algorithm could not identify two different people as separate entities and considered them as a single blob frame before marking it as such.

The algorithm based on the fuzzy classification model proposed in [2] gives no outputs on a regular basis since it lacks enough data for making a successful prediction. The model mislabeled around 15% of the test data, and quite a few cases need the image to be examined thoroughly before any manipulation can be done on it. YOLO integration in CCTV cameras were used for purposes like pinpointing fire hotspots to overcome the need for sensors [3] to automatically recognize license plates [4].

An efficient super-resolution algorithm was proposed [5], which overcame the approaches that were too convoluted and impractical to be used in real-life scenarios by making use of the overlapping bicubic for real-time hardware deployment, they were able to upscale two-time ($s = 2$) from the original low-resolution image. Similarly, in another article [6], where a combination of super-resolution being an extension to multiple frames super-resolution method gave high-resolution images from a set of low-resolution images captured from CCTVs, Keren registration algorithm, and projection onto convex sets (POCS) were used to generate high-resolution images. This article gave results that were better than other resampling techniques.

3 Approach

The proposed algorithm uses three techniques, particularly super-resolution, object detection, and sound source separation. The use case diagram of the entire workflow of the proposed technique is shown in Fig. 1.

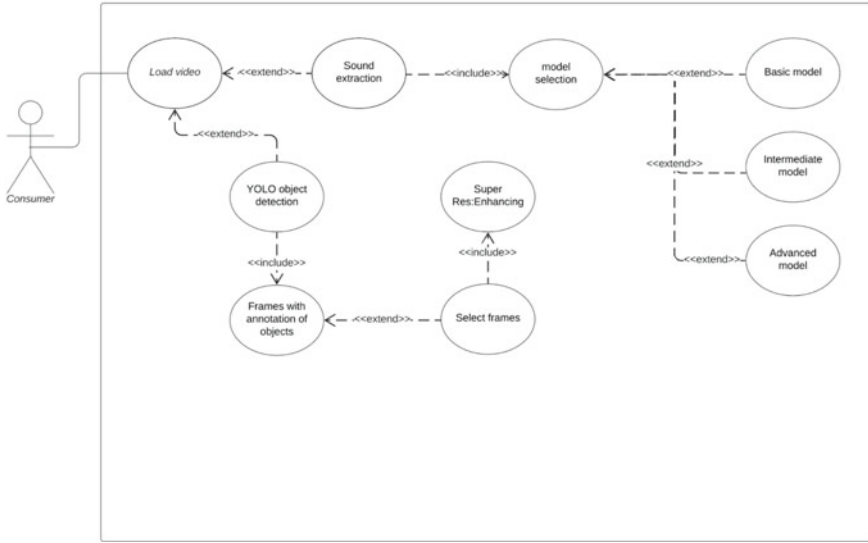


Fig. 1 Workflow of the CCTV footage analysis

3.1 Object Detection

The items are to be tracked in real-time and comprehend their actions, and YOLO was the best fit for the use case. This model was implemented with the help of convolution layers. In order to evaluate the performance of the mode, the proposed algorithm has made use of the PASCAL VOC detection dataset [7]. The convolution layers were for the feature extraction, and then the fully connected layers to predict the item’s class.

Implementation Details An architecture similar to GoogleNet for image classification [8] is followed. The network architecture made use consists of 24 convolutional layers along with two fully connected layers at the lead. In contrast to GoogleNet which makes use of inception modules, the proposed algorithm makes use of 3×3 convolutional layers preceded by 1×1 reduction layers, which is similar to Lin et al. [9].

The convolutional layers are pretrained on the ImageNet 1000-class competition dataset [10]. The accuracy of the model is comparable to that of the GoogleNet models present in the Caffe Models Zoo [11]. It took us approximately a week to train the network, after which a single crop top-5 accuracy of 88% on the ImageNet 2012 dataset was obtained. For training and inference, Darknet [12] is used. For the final layer, a linear activation function and while using LeakyReLU for all the remaining layers is used. Sum-squared error is used as the loss function, which helps in better optimization. Rate of 0.5 was fixed for the dropout layers after the first

Table 1 Object detection results

Delayed real time	Train	MAP	FPS
Faster DPM	2007	30.4	15
R-CNN minus R	2007	53.5	6
Fast R-CNN	2007 + 2012	70.0	0.5
Fast R-CNN VGG16	2007 + 2012	73.2	7
Faster R-CNN ZF	2007 + 2012	62.1	18
YOLO VGG 16	2007 + 2012	66.4	21

connection layers which helped in preventing the co-adaptation between different layers.

The network is trained on the PASCAL VOC 2007 and 2012 and the training and the validation subsets were created from these datasets. While testing the network in 2012, data from the VOC 2007 test set was also used for training. While training, a momentum of 0.9, decay of 0.0005, and passed data in a batch size of 64.

The learning rate had an incremental approach from 10−3 to 10−2. Higher learning caused the models to diverge because of unstable gradients. The network is trained on 10−2 for the first 75 epochs and then on 10−3 for 30 epochs, and finally 10−4 for 30 epochs. This seemed to render the best results.

The models are compared against other established models in delayed real-time prediction to understand the performance difference between them. All of them were trained on the PaSCAL VOC 2007 + 2012 dataset. Its performance surpassed state-of-the-art models like R-CNN minus [13], Fast R-CNN [14], Faster R-CNN VGG 16, Faster R-CNN ZF [1]. The results are displayed in Table 15.

3.2 Super Resolution

Enhancing the details and upscaling the low-resolution images to create a high-resolution image is called super-resolution. CCTV footage frames generally lack details or could be blurred or distorted. Super-Resolution can be applied here to upscale these frames to reveal crucial information from these images, for which otherwise the texture details would be absent. There is a wide range of applications for the same [16, 17]. The exciting work can be modified to incorporate SR and gain significant performance gain while having low power consumption devices. Deep residual network with skip-connection is used to create the super-resolution GAN. This network is optimized using mean squared error (MSE); apart from this, a novel perceptual loss using high-level feature maps of the CGG network and a discriminator to distinguish generated images from ground truth to provide the adversarial effect.

Implementation Details The goal is to train a GAN that can generate high-resolution counterparts for a given low-resolution image. This can be done by using a

generator network that is a feed-forward CNN. The network was trained to minimize the SR-specific loss function. This loss function consists of several loss components that the model can derive different characteristic features for the new SR image. The discriminator would be the same as that of Goodfellow et al. [18], which is optimized successively to the generator. The generator is trained to fool the discriminator in a challenging way that the discriminator learns more to differentiate between real and fake images while the generator learns to create fake images that look very much like the fake image. The model is trained on the architecture design proposed by Radford et al. [19] using LeakyReLU as the activation function with no max pooling present in the entire network. The experiments are done on the three extensively used benchmark datasets, particularly Set5 [20], Set14 [21], and BSD100. For testing, BSD 300 [22] is used. Then, between the low-resolution and its super-resolution counterpart, a scale factor of $4 \times$ between the low-resolution and its super-resolution counterpart is performed.

Training is done with the help of images from the ImageNet database [10]. Lower resolution images were obtained by downsampling from higher resolution images using bicubic kernel downsampling factor $r = 4$. MSE loss is calculated for the images. The optimizer used is Adam [23], and a learning rate of 10^{-5} alteration between the networks is done similar to n Goodfellow et al. [18].

The testing parameter used in mean opinion score testing since this is the best method among all the different approaches available for reconstruction of images that are perceptually cogent. Methods like PSNR or SSIM do not capture or reflect the image quality on how humans see images; this is where MOS testing shows its superiority.

This model is not particularly made with the intention of best computational efficiency. This is where the use of the YOLO model over the images comes in so that the SR GAN has only to be applied on a specific image that the user selects from the application. The SR-GAN is compared to SRRResNet and bicubic interpolation and a normal neural network since these are the best state-of-the-art methods. Table 2 shows the gain in performance in these models and shows how SR-GAN compared to the other models when it comes to super resolution.

Sample outputs of the proposed method on some frames from CCTV footage can be seen in Fig. 2 and 3. It can be observed that there is a very noticeable difference between the two images. The super resolutions output image has details that are extremely sharp compared to the raw input from a CCTV feed.

3.3 *Sound Source Separation*

The source separation problem has interested a large community of sound signal researchers for a couple of decades now. It starts from a simple observation: most video recordings from any source are usually a mix of several individual audio tracks (human voices, vehicles, construction machinery, etc.). The task of sound source

Table 2 Super resolution results

Set5	nearest	bicubic	SRCNN	SRGAN	HR
PSNR	26.26	28.43	30.07	29.40	∞
SSIM	0.7552	0.8211	0.8627	0.8472	1
MOS	1.28	1.97	2.57	3.58	4.32
Set14					
PSNR	24.64	25.99	27.18	26.02	∞
SSIM	0.7100	0.7486	0.7861	0.7397	1
MOS	1.20	1.80	2.26	3.72	4.32
BSED100					
PSNR	25.02	25.94	26.68	25.16	∞
SSIM	0.6606	0.6935	0.7291	0.6688	1
MOS	1.11	1/47	1.87	3.56	4.46



Fig. 2 Super resolution on a CCTV footage frame (outdoors)



Fig. 3 Super resolution on a CCTV footage frame (indoors)

separation is: given an audio track, the need is to recover these separate tracks, sometimes called stems. If successfully managed to do this, the aid of audio from video sources like CCTV footages or any footage with audio from a crime scene would increase a 100-fold. It would be easier to identify what area the footage is from the background noise; identification of the types of vehicles involved, if any,

even the voices of people involved might be accessible. The following parts cover how exactly this has been done.

Implementation Details The human brain is very good at isolating sources. Humans just need to focus on one of the sources in audio and will be able to hear it quite distinctly from the others. Yet, that is not really separation since all the other parts are still audible, making it tough to isolate the most critical parts. In many cases, it may not be possible to exactly recover the individual sources that have been mixed together. Thus, the challenge is to approximate them the best possible way, that is, extract sources as close as possible to the originals without creating too many distortions.

Our technique provides pre-trained models for:

- voice/accompanying vocals separation.
- 4 sources (stems) separation similar to the implementation of SiSec [24]
- 5 stems separation

Sound source separation in the proposed models follow an architecture like the UNets [25] following specifications akin to [26]. UNet incorporates convolutional neural network with an encoder/decoder architecture having skip connections. For the proposed algorithm, the implementation is UNets with 12 layers, with layers equally split between the encoder and the decoder, i.e., 6 for each. The main reason for using a UNet architecture is to avoid making use of multiple models. The most straightforward approach for sound source separation is to train independent models, each model is dedicated for estimating a single source. However, with the control mechanism of a UNet a single model can be trained to estimate several sources, and this single model achieves a performance that is at par with training dedicated models for each source.

The proposed method uses the UNet on each sound source to approximate a soft mask for each of them. A single UNet model each sound source by taking a mixed spectrogram as the input and providing an estimated sound spectrogram as the output. $L1$ -norm was used between the masked input mix spectrograms and source-target spectrograms to calculate the training loss. The network has been trained on the Bean dataset used in [26] using Adam as the optimizer [23]. It took us approximately one week to complete the training on a single GPU. The proposed algorithm makes use of soft masking and Wiener filtering in a multi-channel setting for the final separation from the source spectrograms is approximated.

Speed: The model is useful for near real-time inference on large datasets when using a GPU, as CNN-based architecture makes parallel computing highly efficient. The inference can be performed on a CPU as well. The models are tested on the audio extracted from up to 5 min of footage, on which a speedy result on a CPU is obtained and made a tool fully accessible to anyone with or without a GPU.

Separation performances: The performance of the proposed models is at par with models trained on the standard musdb18 dataset [27] without us performing any training or validation on musdb18. The obtained results are in terms of standard

Table 3 Sound source separation result

	Vocals			
	SDR	SIR	SAR	ISR
Mask	6.55	15.19	6.44	12.01
MWF	6.86	15.86	6.99	11.95
Open-Unmix	6.32	13.33	6.52	11.93
	<i>Other</i>			
Mask	4.24	7.86	4.63	9.83
MWF	4.55	8.16	4.88	9.87
Open-Unmix	4.02	6.59	4.74	9.31

source separation metrics [28], namely SAR: signal to artifacts ratio, ISR: image to spatial distortion ratio, SIR: signal to interference ratio and signal to distortion Ratio (SDR). The results are presented in Table 3 where a comparison of the model performance with Open-Unmix [24]. Also, presenting results for soft masking and for multichannel Wiener filtering (applied using Norbert [29]). As can be seen, the models are competitive with Open-Unmix and especially on SDR for all instruments for most metrics.

4 Future Work

Future goals include packaging the proposed method and deploying it as a desktop application that can be used by everyone. A desktop application is proposed for better security of data so that all the operations are performed on a user's local system with no data being uploaded to a cloud platform, leading to a risk of the data being leaked.

The proposed tool shall also be deployed in the form of a lightweight Web application in the future for users not requiring all the features. For this, the models trained need to be made light weight in order for the inference to happen on the videos in a timely fashion.

In order to improve the performance of the tool, the proposed sound source separation shall be further enhanced, especially with respect to extracting speech from a given video. An approach similar to the one mentioned by Manoharan et al. [30] shall be implemented and customized for this tool. For providing better results on enhancing the resolutions of the selected frames, methods proposed by Bhusan et al. In [31] shall be used. The object detection method currently proposed shall also be enhanced. 3D image modeling techniques [32] to identify objects with greater accuracy shall be implemented. 3D scene contractions techniques such as [33] can also be considered for high-performance systems.

5 Conclusions

The goal was to research and develop an open-source, intelligent, and easy-to-use application that would provide premium video-enhancement features and AI-driven object detection capabilities for this research endeavor. Taking the low-resolution footage that is associated with CCTV cameras into consideration, the hope is to offer a free option for any user/individual needing to enhance his/her video feed. To implement an actual prototype, the proposed algorithm uses the super-resolution model to improve video feed and integrated object detection through the use of an efficient real-time object detection algorithm and a stable sound extraction feature. The plan is to train the models on more video data by generating them synthetically [34]. Further, the plan is to implement a more robust technique for object detection and tracking by additionally using a classification technique based on decision trees [35].

Acknowledgements We offer our sincere gratitude to our mentor Anjali T, Assistant Professor, for helping us and leading us throughout this project without whom we wouldn't have been able to complete this. We would also like to express thankfulness to our colleague Akhil K G for his detailed observations and encouragement.

References

1. L.M. Fuentes, S.A. Velastin, Tracking-based event detection for CCTV systems. *Pattern Anal. Appl.* **7**(4), 356–364 (2004)
2. A. Matiolański, A. Maksimova, A. Dziech, CCTV object detection with fuzzy classification and image enhancement. *Multimedia Tools Appl.* **75**(17), 10513–10528 (2016)
3. D.P. Lestari, R. Kosasih, T. Handhika, I. Sari, A. Fahrurrozi, Fire hotspots detection system on CCTV videos using you only look once (YOLO) method and tiny YOLO model for high buildings evacuation, in *2019 2nd International Conference of Computer and Informatics Engineering (IC2IE)* (IEEE, 2019), pp. 87–92
4. R. Laroca, E. Severo, L.A. Zanlorensi, L.S. Oliveira, G.R. Gonçalves, W.R. Schwartz, D. Menotti, A robust real-time automatic license plate recognition based on the YOLO detector, in *2018 International Joint Conference on Neural Networks (IJCNN)* (IEEE, 2018), pp. 1–10
5. W. Ruangsang, S. Aramvith, Efficient super-resolution algorithm using overlapping bicubic interpolation, in *2017 IEEE 6th Global Conference on Consumer Electronics (GCCE)* (IEEE, 2017), pp. 1–2
6. N.N.A.N. Ghazali, N.A. Zamani, S.N.H. Sheikh Abdullah, J. Jameson, Super resolution combination methods for CCTV forensic interpretation, in *2012 12th International Conference on Intelligent Systems Design and Applications (ISDA)* (IEEE, 2012), pp. 853–858
7. M. Everingham, S.M. Ali Eslami, L.V. Gool, C.K.I. Williams, J. Winn, A. Zisserman, The pascal visual object classes challenge: a retrospective. *Int. J. Comput. Vision* **111**(1), 98–136 (2015)
8. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in *Proceedings of the IEEE conference on computer vision and pattern recognition* (2015), pp. 1–9
9. M. Lin, Q. Chen, S. Yan, Network in network, arXiv preprint [arXiv:1312.4400](https://arxiv.org/abs/1312.4400) (2013)
10. O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang et al., Imagenet large scale visual recognition challenge. *Int. J. Comput. Vision* **115**(3), 211–252 (2015)

11. B.V.L.C. Caffè, Models accuracy on imagenet 2012 val (2015)
12. J. Redmon, Darknet: open source neural networks in c 2018 (2013)
13. K. Lenc, A. Vedaldi, R-CNN minus r, arXiv preprint [arXiv:1506.06981](https://arxiv.org/abs/1506.06981) (2015)
14. R.B. Girshick, Fast R-CNN. CoRR, abs/1504.08083 (2015)
15. S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: towards real-time object detection with region proposal networks. *Adv. Neural. Inf. Process. Syst.* **28**, 91–99 (2015)
16. D. Arjun, P.K. Indukala, K.A. Unnikrishna Menon, Border surveillance and intruder detection using wireless sensor networks: a brief survey, in *2017 International Conference on Communication and Signal Processing (ICCSP)* (IEEE, 2017), pp. 1125–1130
17. S. Veni, R. Anand, B. Santosh, Road accident detection and severity determination from CCTV surveillance, in *Advances in Distributed Computing and Machine Learning* (Springer, Singapore, 2021), pp. 247–256
18. I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in *Advances in Neural Information Processing Systems*, vol. 27 (2014)
19. A. Radford, L. Metz, S. Chintala, Unsupervised representation learning with deep convolutional generative adversarial networks, arXiv preprint [arXiv:1511.06434](https://arxiv.org/abs/1511.06434) (2015)
20. M. Bevilacqua, A. Roumy, C. Guillemot, M.L. Alberi-Morel, Low-complexity single-image super-resolution based on nonnegative neighbor embedding (2012), pp. 1–10
21. R. Zeyde, M. Elad, M. Protter, On single image scale-up using sparse-representations, in *International Conference on Curves and Surfaces* (Springer, Berlin, Heidelberg, 2010), pp. 711–730
22. D. Martin, C. Fowlkes, D. Tal, J. Malik, A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, in *Proceedings Eighth IEEE International Conference on Computer Vision (ICCV)*, vol. 2 (IEEE, 2001), pp. 416–423
23. D.P. Kingma, J. Ba, Adam: a method for stochastic optimization, arXiv preprint [arXiv:1412.6980](https://arxiv.org/abs/1412.6980) (2014)
24. F.-R. Stöter, S. Uhlich, A. Liutkus, Y. Mitsufuji, Open-Unmix-a reference implementation for music source separation. *J. Open Source Software* **4**(41), 1667 (2019)
25. A. Jansson, E. Humphrey, N. Montecchio, R. Bittner, A. Kumar, T. Weyde, Singing voice separation with deep U-net convolutional networks (2017)
26. L. Prêtet, R. Hennequin, J. Royo-Letelier, A. Vaglio, Singing voice separation: a study on training data, in *ICASSP 2019–2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (IEEE, 2019), pp. 506–510
27. Z. Rafii, A. Liutkus, F.-R. Stöter, S.I. Mimilakis, R. Bittner, MUSDB18—a corpus for music separation (2017)
28. E. Vincent, R. Gribonval, C. Févotte, Performance measurement in blind audio source separation. *IEEE Trans. Audio Speech Lang. Process.* **14**(4), 1462–1469 (2006)
29. A. Liutkus, F.-R. Stöter, sigsep/norbert: first official norbert release (2019)
30. S. Manoharan, N. Ponraj, Analysis of complex non-linear environment exploration in speech recognition by hybrid learning technique. *J. Innovative Image Process. (JIIP)* **2**(04), 202–209 (2020)
31. S. Bhushan, D. Shean, O. Alexandrov, S. Henderson, Automated digital elevation model (DEM) generation from very-high-resolution Planet skysat triplet stereo and video imagery. *ISPRS J. Photogramm. Remote. Sens.* **173**, 151–165 (2021)
32. A. Sungheetha, R. Sharma, 3D image processing using machine learning based input processing for man-machine interaction. *J. Innovative Image Process. (JIIP)* **3**(01), 1–6 (2021)
33. Z. Murez, T. van As, J. Bartolozzi, A. Sinha, V. Badrinarayanan, A. Rabinovich, Atlas: end-to-end 3d scene reconstruction from posed images, in *Computer Vision—ECCV 2020: 16th European Conference*, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16, (Springer International Publishing, 2020), pp. 414–431
34. K. Namitha, A. Narayanan, M. Geetha, A synthetic video dataset generation toolbox for surveillance video synopsis applications, in *2020 International Conference on Communication and Signal Processing (ICCSP)* (IEEE, 2020), pp. 493–497

35. T. Anjali, N. Rakesh, K.M.P. Akshay, A novel based decision tree for content based image retrieval: an optimal classification approach, in *2018 International Conference on Communication and Signal Processing (ICCSP)* (IEEE, 2018), pp. 0698–0704
36. F.-R. Stöter, A. Liutkus, N. Ito, The 2018 signal separation evaluation campaign, in *International Conference on Latent Variable Analysis and Signal Separation* (Springer, Cham, 2018), pp. 293–305

A Real-Time Approach of Fall Detection and Rehabilitation in Elders Using Kinect Xbox 360 and Supervised Machine Learning Algorithm



V. Muralidharan and V. Vijayalakshmi

Abstract Nowadays, fall in elders is a major issue almost in all the countries. Sometimes, heavy fall in elders cause serious injuries which leads to major medical care. Fall may lead to disability and also cause mortality to the elderly people. Due to the development of science and technology, the life of the fallen elders is rescued, and the injuries are healed. The newly developed technologies bring happiness and makes the elders life comfortable. At present, fall detection and prevention draws the attention of researchers throughout the world. New technology like Kinect Xbox 360 brings a new way to develop new intelligent system, which could be used to monitor the elderly people in their daily activities. Kinect Xbox 360 is a low-cost device. It tracks the body movements. It is used by the elders doing rehabilitation exercises in the homely environment. Elders who are living alone face the risk of fall. Activity recognition system is a very important technology for elderly people to do their daily activities in their life. Physiotherapy is one of the branches of rehabilitation science which brings differences in the ability and makes the individual to lead a healthy life. In this paper, we are going to analyze various methods of human fall detection and techniques by noticing the daily activities of the elders. We are also going to see different types of machine learning algorithm used for fall detection.

Keywords Fall detection · Kinect · Supervised machine learning algorithm · Homely environment · Daily activities · Medical care

1 Introduction

Due to advancement of technology diagnosis and treatment of diseases, the span of life regarding elders is increasing day by day. It has been estimated by World Health

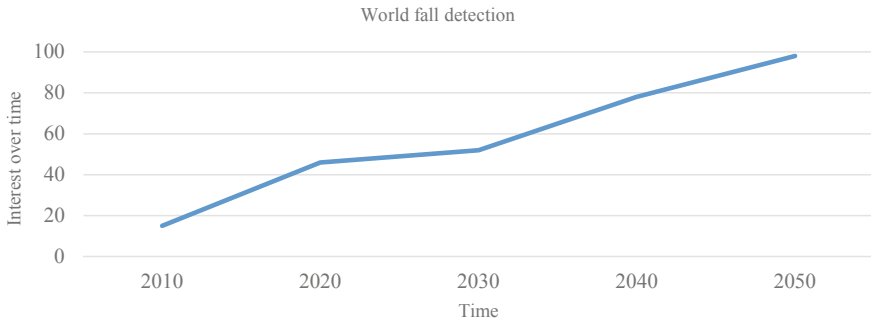
V. Muralidharan (✉)

Research Scholar, Department of Computer Science, Government Arts College (Grade-I), (Affiliated to Bharathidasan University), Ariyalur 621713, India

V. Vijayalakshmi

Assistant Professor & Head, Department of Computer Science, Government Arts College (Grade-I), (Affiliated to Bharathidasan University), Ariyalur 621713, India

Organization that in 2050, the number of persons over 65 years will be nearly 2.5 billion. The increasing of elderly population results in more number of falls. Due to the increase of age, the rate of fall and fall injuries raises gradually. Interest of fall detection overtime, from January 2010 to December 2050, is shown below in the given graph. The data are taken from the Google Search fall detection.



Fall in elders leads to major problem for the whole world [1]. After fall, individual medical care decreases the damages from fall injuries which results the increases of survival rate. The fall detection system functions very quickly and detect the fall [2]. Thus, it gained more importance. At present, researchers draw their attention toward fall detection and take remedial measures. The existing fall detection system is divided into three groups. They are ambience device, camera-based systems and wearable devices [3]. The devices of ambience are fixed in an area. It detects the fall. Some of the ambience devices are Doppler, microphone and accelerometer sensors. Computer vision uses camera which tracks the human movements. In this, fall can be detected when a person is inactive for a long duration. Wearable devices are attached to the body of the user to detect the fall [4]. Some of the wearable devices are accelerometer and gyroscope.

Elderly people who are suffering from injuries, disability or any other reason must practice rehabilitation exercises so that they may perform their daily activities without others help [5]. Physiotherapist in rehabilitation centers identify the problems of the elderly people, and they design suitable rehabilitation therapy to the individual concern. Some people do not adhere to the programs suggested by the physiotherapist, and they do the rehabilitation exercises in their own way. The patient must pay keen attention to both the body postures and the range of motions, repeating the exercises suggested by the therapist so that the patient can avoid unnecessary strain to the joints and muscles and may avoid further injuries.

Kinect is a low-cost sensor box which is used for playing Xbox games in the beginning [6]. Kinect consists of video camera, combined with depth sensor which could be able to measure the distance between an object and the Kinect box. Kinect detects 3D space about 4 m in depth and an angular field of view of 30 degrees to right and left. It is designed in the way that the camera of Kinect tracks human skeleton and joints. The Kinect sensor to some extent record human movements. This device can

be easily portable from place to place [7]. Kinect sensor was introduced in the year 2010. Kinect offers natural user interface and not only tracks the body movements but also records voice commands, interpret gestures and facial expression, speech recognition and environment recognition.

Machine learning has its significant development over the past few years. It is a branch of science which exhibits the ability of machine in classifying the data given to them. Some of the benefits of classification of data are as follows.

- It helps in data protection.
- It improves data security.
- It improves user productivity and decision making.
- It helps to eliminate unnecessary data.
- It also develops algorithm.

Machine learning algorithm helps the computer by converting complex patterns and develops intelligent [8]. In machine learning, data play an important role. It is used to gain knowledge from the data. The learning and prediction performance is affected by the quality of the dataset. Professor Husain-Tien Lin states in his book that machine learning is otherwise called as learning from data. Data occupies an important position in machine learning. Before entering into the study of machine learning, we must know first of all the notations of dataset. There are two types of datasets. First one is labeled dataset and next one is unlabeled dataset. We can group machine learning under three categories. They are supervised learning, unsupervised learning and reinforcement learning.

In supervised learning, the training set is labeled dataset. Supervised learning finds out the similarities between the feature set and the labeled set, that is, the knowledge and properties learn from labeled dataset. In unsupervised learning, the training sets are unlabeled dataset. It deals with clustering, probability density estimation, finding relationship among features and dimensional reduction. The results obtained from unsupervised learning could be used for supervised learning. Reinforcement learning is used to find out solution for salvation in decision making, for example, automatic vehicle driving.

Feature extraction plays a vital role in machine learning. The need for feature extraction are as follows. Extraction of technique of features are useful when there is large dataset. The need for number of resources are reduced. While doing so, there is no loss of relevant information. It helps in the reduction of the amount of redundant data from the dataset. Reduction of data helps to make the model with less effort of machine and the speed of learning is increased.

2 Daily Human Activity Recognition

Activity recognition means the task of recognizing the present physical action done by users in the set of definitive environments. In human activity recognition system, there occurs many challenges. The assistive technology helps the elderly people in fitness

tracking [9]. A fall may be an incident which brings rest to a person on the ground. It causes severe injuries and sometimes event to mortality. Internal and external factors are responsible for fall. Fall may occur, if a person loses his/her consciousness or slips all of a sudden while running or walking. Some of the internal causes are medical conditions like neurological, cardiac or other disabling conditions. Some other causes are medication of side effects, loss of balance, physical inability, especially among elderly peoples, poor movements and impaired vision and of hearing. Some of the external factors are overcrowded housing, poor footpaths etc. Tiredness, weakness in body condition and lack of concentration are other factors which lead to fall. Because of violent attack, fall occurs to some persons (Fig. 1).

Presently, due to the development of technology, human activity recognition reached its peak position. This technology is used to recognize the action of the user. It is also used to assist the task with the use of computing system. In this study, computer vision research as contributed a lot in this aspect. Human activity research indicates physical human activity [10]. In the beginning, research on human activity uses gestures. Gesture recognition is one of the main sub-topics of action recognition. At present, it gained more importance for its role in human machine interaction. In the past, devices like mouse, keyboard or touch screen were used. For elders and disabled peoples, it is difficult for them to use. So researchers are trying to find an alternative source. Table 1 gives approaches such as technology, merits and demerits of different techniques and some application of gesture recognition.

Human research activity is challenged by the traditional medical procedures. It gives great results in the areas of other fields like sports and industries. It also caters to the needs of human activities transportation, brushing teeth and medication etc. It is also used for gaming experience while using Kinect Xbox 360. We can use activity recognition for single person or a group. In order to identify single user, multiple user recognition can be performed, and thereby actions are tracked. This is used to



Fig. 1 Daily activities of elderly people

Table 1 Technology of gesture recognition

S. no.	Approach	Technology	Merits	Demerits	Applications
1	Vision based	Surveillance camera	High accuracy	Privacy issue	Gaming, smart screen interaction
2	Depth sensor	Kinect	Low price	Private issue	
3	Wearable sensors	Smart watch	High accuracy	Difficult to use	
4	Object tagged	Ultrasonic sensor	High accuracy	Device based	

monitor video camera. In the process of single user recognition, there are many ways to collect data.

3 Fall Detection System

An event that leads a person to lie or rest on the ground is described as fall. Fall in elders is a major problem globally. Fall occurs mostly to the elders due to age [11]. Fall causes major risk mainly to male, when comparing with that of female. There are five types of falls. They are

- Backward fall occurs when a person fall backward and rest on the ground.
- Forward fall occurs when a person fall forward and rest on the ground.
- Forward fall on the knees bragging the chair occurs when a subject fall forward remains on the ground and bragging the chair.
- Left side fall occurs when a subject fall on the left side and remains on the ground.
- Right side fall occurs when a person falls on the right side and remains on the ground.

There are different methods available to detect fall events. They can be divided into two kinds. They are

- Non computer vision-based methods and
- Computer vision-based methods.

3.1 Non-Computer Vision-Based Method [12]

In non-computer vision-based methods, different kinds of sensors such as acoustic, acceleration and floor vibration sensors are used to detect sound [13], vibration and data of human body movements [14]. The information is gathered is analyzed to detect the reason for fall. There are two ways by which non-computer vision-based methods are used. They are wearable sensor methods and ambient fusion-based

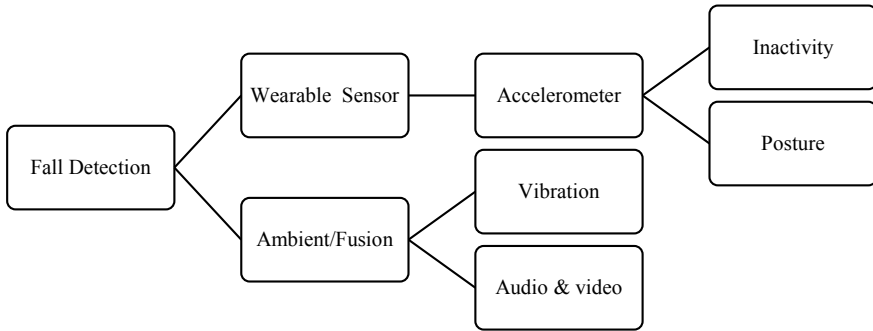


Fig. 2 Classification of non-computer vision-based method

methods. In wearable sensor method, we can use accelerometer, gyroscopes and tilt meter to detect the fall events. There are some disadvantages in this system. The elders always used to carry throughout the day wherever they go. Elders may sometime forget to wear and may forget to charge the battery. They may feel uncomfortable to wear it. Because of these, researchers pay their attention towards an alternative method. Ambient sensors consist of floor sensor, pressure sensor and an infrared sensor. We can fix floor and pressure sensors on the ground so that at time of fall, frequency of vibrations are recorded in the sensor. In addition to above-mentioned sensors, infrared sensor is also attached which records the motions in the surroundings environment. These systems are free from environment, and there may be difficulties in the installation of the above-mentioned sensors because the flooring is different from house to house, so we find difficulties in the installation (Fig. 2).

3.2 Computer Vision Based Method [12]

This system has foremost benefits when compared with that of non-computer vision system. There is no need to wear any device in this system. More cameras are used to track the human movements. There are three ways by which their performance is carried out.

- Using single RGB camera.
- Using 3D-based multiple cameras.
- Using 3D-based depth cameras.

3.2.1 Using Single RGB Camera

We can easily setup a single RGB camera for detecting human fall. It costs very low. In order to detect, fall need features which relates to shape and human motion analysis. Mirmahboub et al. use an ordinary simple method to produce a silhouette

of a person, and multiple features are then obtained from the silhouette area. A classification can be done with the help of silhouette of an individual.

3.2.2 Using 3D-Based Multiple Cameras

By using 3D-based multiple camera, we can also detect the human fall. This 3D multiple camera systems reconstruct the object and at the same time, we pay attention to time-consuming calibration process. Auvinet et al. uses multiple cameras for the reconstruction of 3D shape of the individual. By analyzing volume distribution, events of fall can also be detected. An alarm is setup to alert when event of fall occurs.

3.2.3 Using 3D-Based Depth Cameras

In the beginning, researchers used time-of-flight 3D camera for finding fall detection. It is very expensive. Due to the development of depth sensing technology, a new device called Microsoft Kinect Xbox 360 draws the attention of the researchers. It is with the help of depth camera, we can simply calculate the distance between top of the person and to the floor. This can be used to detect the features of the human fall.

By using normal video camera or by depth video camera, movements of the person are monitored in vision-based system. When a falling posture is detected, an alarm indicates help to prevent fall. A single setup of these system can be supervised more than a person at a time. So, these systems are very convenient to use and free from environment. They can be used in previous installed surveillance and security cameras. There are many improvements in using computer vision method such as detection and recognition of objects, classification of images and segmentation. Researchers made use of these concept in areas of various vision-based application.

According to Perry et al. fall can be grouped under three categories. They are

- Methods that measure acceleration.
- Methods that measure acceleration combined with other methods.
- Methods that do not measure acceleration.

All fall detection systems are similar. The main objective is discrimination between fall event and daily activity living (DAL). Certain activities of daily living are not an easy task like sitting down or going from standing position to lying down has strong similarities to falls. To test a fall detector, we are in need of data pertaining to fall and daily activities of living. Sensors recorded the data. It is recorded in the form of acceleration signal, images, pressure signals etc. They are processed and classified using detection techniques. They are expressed in the form of sensitivity and specificity. Fall is used for detecting sensitivity and daily activity living is used for detecting specificity.

4 Fall Rehabilitation

Rehabilitation belongs to a branch in medical science which deals with restoring the ability of an affected individual and make him to do his daily routine work. Patients suffering from injuries [15] or disability have negative effects often in their day-to-day activities. We have to give rehabilitation process to these types of people. In rehabilitation centers, the cost is very high. In order to support the patient, we have to make the patients to do the rehabilitation exercises in homely environment. The effect of the age differs in time from person to person, but there appears some in differences. E.g., slowness, balance disorders which results in physical limitations.

At the time of walking, elders move their hip largely when compared with that of younger generation. Youngers have very slow hip movements and the knee movements. Some of the falls are caused due to intrinsic factors. It is because of weakness in lower extremity muscles and limitation in lower limb joints mobility results in impaired gait pattern. They are called as significantly as risk factors. Death is considered to the sixth factor for elders who are above 65 years of old, and death is considered to the second factor for those who are in between 65 and 75 years old and for the people who are above 75 years of old, it is considered to the first factor. Recent study considers physical activities are very essential for maintaining good health and independence. Exercises pertaining to home games can be used successfully for preventing fall. Health games and therapy combined games contribute more care effectively (Fig. 3).

4.1 General Concept for Fall Prevention in Elders

- We must take steps to improve the life style of the elderly people.
- Elders must be given independence by giving rehabilitation exercises.
- We must support the elders through telecommunication with their families.

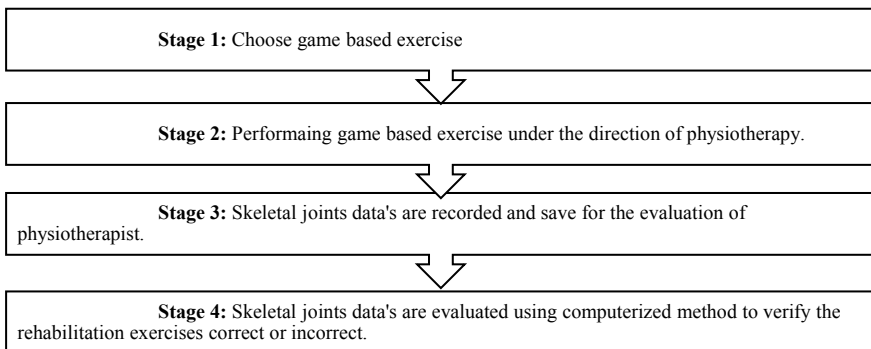


Fig. 3 Execution of game-based rehabilitation exercises

- We have to develop guidelines for elders regarding physical activities.
- We must offer pleasant opportunities for their physical activeness.
- Elders are to be educated how to live a safety life in their old age.
- We must ensure elders the good health care.
- Due weightage should be given to words of elderly people.
- Caretakers are educated how to take care of elderly people.

We can develop the health and quality of life of affordable actions. More than 50 trails are made to investigate the benefits of exercises. It is because of variation in the exercise types, intensity, frequency and time taken. The program of rehabilitation should be started on the day of operation or accident. By this way we can save time, money, individual suffering. Delay in this process leads to unnecessary mental and physical problems. The steps to be followed in this method are

- Evaluation.
- Physical medicine.
- Psychological supportive training.
- Vocational training.

The technique includes positive diagnosis and patient's rehabilitation potential. The team members who have attended the patient's recorded the collected information and data regarding the patient and evaluation is made. The team members made evaluation along with patient and members of the team make their individual contribution to rescue the patient. The part of the physiotherapist is vital because he has to explain briefly about the exercises pertaining to the patient. The physiotherapist helps in the rehabilitation planning by considering the nature and distribution of abnormal muscular activity.

5 Technology in Kinect

Kinect is used to track human body movements [16] and also used in homely environment for doing rehabilitation exercises [17]. It consists of a set of sensors, and it was developed as an input tool for Xbox 360 and Xbox one gaming console. There is an infrared projector, IR camera which is used for obtaining precise depth maps, a RGB color camera, a four microphone array and a motorized tilt [18]. In both, the camera images are produced at 30 frames per second (fps). The IR camera captures 3D video data with the help of structured light technology. The depth values portray to the imaginary image plane. An irregular pattern of dots is shouted by the IR projector with a wavelength comprising of 700 nm–1 mm. In addition to this, Kinect tracks human skeleton joints. The windows SDK for Kinect gives skeleton tracking, and it allows the users to identify people and track their actions. The depth sensor helps the Kinect to recognize six users standing from 2.6 to 13.1 feet. Moreover, two of the detected skeletons are tracked with the aspect of twenty joint positions. Each skeleton joint is calculated in a three dimensional such as X, Y and Z plane [19].

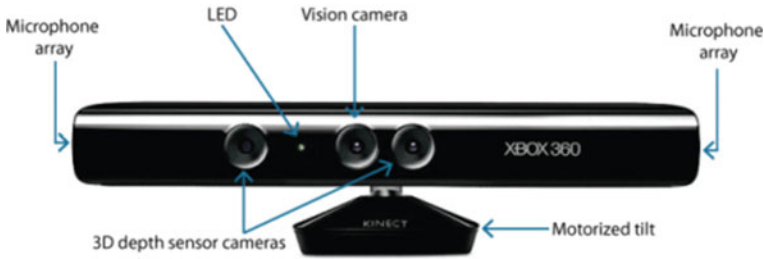


Fig. 4 Kinect Xbox 360

When the joints move from right to left, the X-axis varies. When the joints move up and down Y-axis differs. When the joints move back and forth in relation to Kinect sensor axis varies [20] (Fig. 4).

5.1 Use of Kinect in Fall Detection

The Kinect system detects the activities of the elderly people and track the events. The elderly people is perceptible. The sensor of the Kinect read, and the movements of the elders is restrained [21]. The method of fall detection is portrayed below designing of system (Fig. 5).

There are two stages in operation of the proposed system. They are

1. Acquisition stage.
2. Recognition stage.

The first stage consists of Windows personal computer joined with Kinect sensor by both physiotherapist and patient. Computer is in charge running the machine learning program. A program is used by the physiotherapist, and exercises are recorded and patient should practice it. The patient performs the exercises before the sensor, and it is translated into a machine main program. The patient uses a system for his performance, and the progress is recorded in the database. By using the system, the performer could retrieve the exercise. Regarding skeletal feature

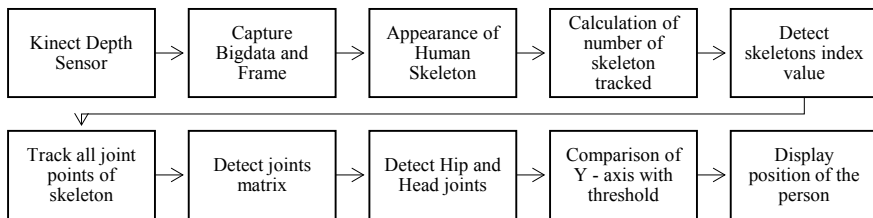


Fig. 5 Method of fall detection

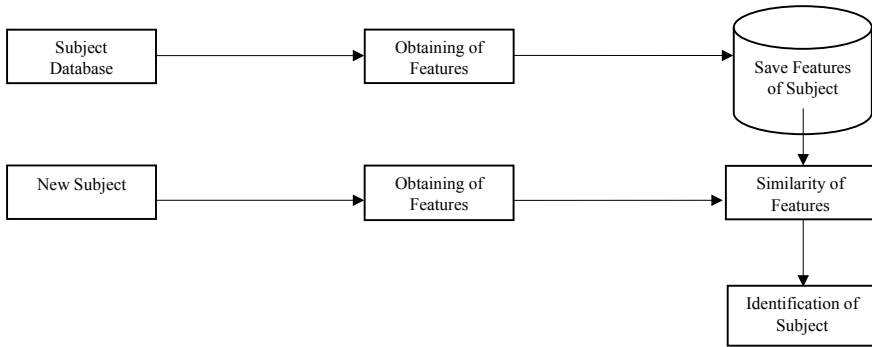


Fig. 6 Operation of acquisition and identification stage

recognition, the Kinect sensor capture RGB video and depth information and extract skeletal data. A skeletal image provides information about the human body as depth maps. The sequence of skeleton provides information about shape, structure, position and angles (Fig. 6).

The second stage is recognition stage. After selecting the exemplar, the user must follow the real time feedback pointing out the body joints. On the basis of comparison, analyze the skeleton points in correspondence with the save image and the target image. Motion instructions are given by the system for correcting in accurate joint. Apart from this, the system plays back the learning exemplar while the user fails to follow the movements. In this, the skeleton detection of the user before the device is obtained from the depth image. At the end of the process, the result is exhibited and is recorded in database.

5.2 Rehabilitation and Kinect

It is with the help of games; the customized real-time system is developed with the help of Kinect. It allows the elderly people to perform physical rehabilitation exercises by using Kinect [22]. This type consists of serious games which stimulates body mobility by means of immersive experience. These technology helps the rehabilitants to do their exercises in the homely environment. The different types of exercises uploaded in the platform help to do several aspects like strength and aerobic capacities. This system does not need the presence of physiotherapist while the rehabilitants do their exercises. In this system, the rehabilitants are monitored and an audio visual feedback is given during these session so that the user may know if he is performing the exercises correctly which are designed by the physiotherapist to them. There are different kinds of games designed for elders. Some of them are Wii, PlayStation, Wii balance, Xbox. Kinect is chosen because it can be used very easily by the elders. Using Kinect is a natural interaction; hence, it is less intrusive.

The flow chart mentioned below gives information about Kinect-based user activity (Fig. 7).

There are two different types of games included in this platform.

1. Games related with aerobic capabilities [23].
2. Improving strength skills [23].

The first type of game is designed to develop aerobic capabilities. A patient is asked to walk in front of the Kinect with different landscapes [24]. The patients are advised by the system when they should increase their speed and when they stop. At the time of game, they are asked to pick up colorful balls which are scattered on the ground. Four different kinds of activities are developed in aerobic games. They are training of upper limb and lower limb, training both limbs at the same time and training of any one of the limbs.

Strength skills are developed in the second type; the patients are taken to the scenic places to have beautiful view and relaxing sounds which might encourage them. The system observes the number of repeated exercises and series the performance of the patient. In this game, the patient is asked to practice some games in the gym. The patients are instructed what kinds of exercises they can perform. The movements of the patients are detected to verify whether the performer performs correct movements. If the patient is not able to perform the exercises within a stipulated time, the system identifies it. It instructs the doer to switch over to the next exercises. Time taken by the patient while doing different exercises are recorded, and it will be brought to the notice of the physiotherapist.

6 Machine Learning

Machine learning algorithm is a science, which brings out the machine's ability in clear understanding of the data given to them. Algorithm development is included in it. Machine learning algorithm makes the computer to have the clear view of the complex patterns and brings out intelligence at the time of machine learns. It includes numerous processing. If there is any challenge for the machine, it changes its structure, and their buy programs are developed. Alp Aydin et al. defined machine learning as optimizing a criterion of a performance by using example data and previous experience. In the process of machine learning, data plays vital role. The learning algorithm learns knowledge from the data. The quality as well as quantity of the dataset affects the learning and performance prediction. The other name for machine learning is deep learning. The benefits of deep learning are

- Maximum use of unstructured data.
- Removal of need for feature engineering.
- Ability to produce quality results.
- Unnecessary costs may be eliminated.
- Elimination of the use of data labeling.

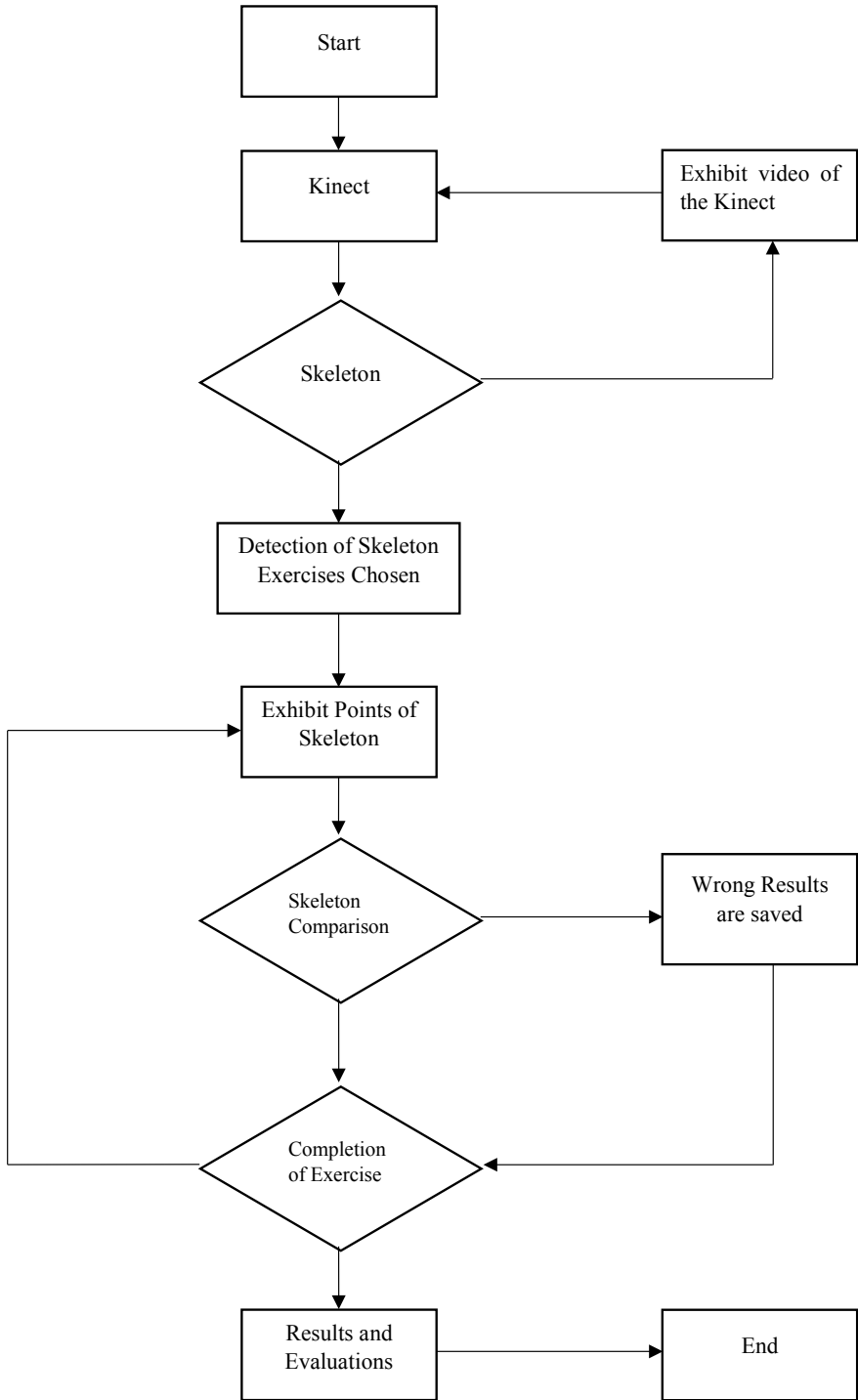


Fig. 7 Kinect-based user activity

Datasets are of two types. They are

- Labeled dataset and
- Unlabeled dataset.

Labeled dataset is a piece of data which is tagged with one or more labels. It identifies certain properties or contained objects. Labels make the data, which has been used for certain machine learning commonly known as supervised machine learning setups. It needs experts to annotate. It is very expensive, very hard and time-consuming to get and store. It is used for complex predicting tasks.

Unlabeled datasets consist of pieces of data. It has not been tagged with labels that identifies characters, properties or classification. Unlabeled data has no labels. It predicts features to represent them. Basically, it is raw data. It is used in unsupervised machine learning. It is obtained by observing and collecting. It is very easy to get and store. It is used as a preprocessing dataset.

An unknown universal dataset assumed to be exist in the machine learning. It contains almost all possible data and the probability distribution appearance in the real world. But in reality, what we see is only a subset of the universal dataset. This is because of memory loss or some other reasons. The other name for acquired dataset is training data. It learns all the properties of the universal dataset. In no free lunch rule, the learned properties can explain the training set, so machine learning is infeasible. The techniques involved in machine learning are statistics and computer science. Statistics is explained as learning the statistical properties from given data. Computer science involves optimization of efficient algorithms, model representation and evaluation of performance.

6.1 Classification of Machine Learning

We can classify the machine learning into three categories. They are (Fig. 8)

1. Supervised learning.
2. Unsupervised learning.
3. Reinforcement learning.

6.1.1 Supervised Learning

The supervised machine learning algorithms consist of dependent variables which are predicted from independent variable. By using the variables, we run a function that creates input from desired outputs. There is continuity in the training process until we achieve accuracy on the training data. In supervised machine learning algorithm, labeled dataset plays as an orientation for data training and testing exercises. Some of the popular supervised machine learning algorithms are linear regression, decision tree, random forest, k-nearest neighbor and logistic regression.

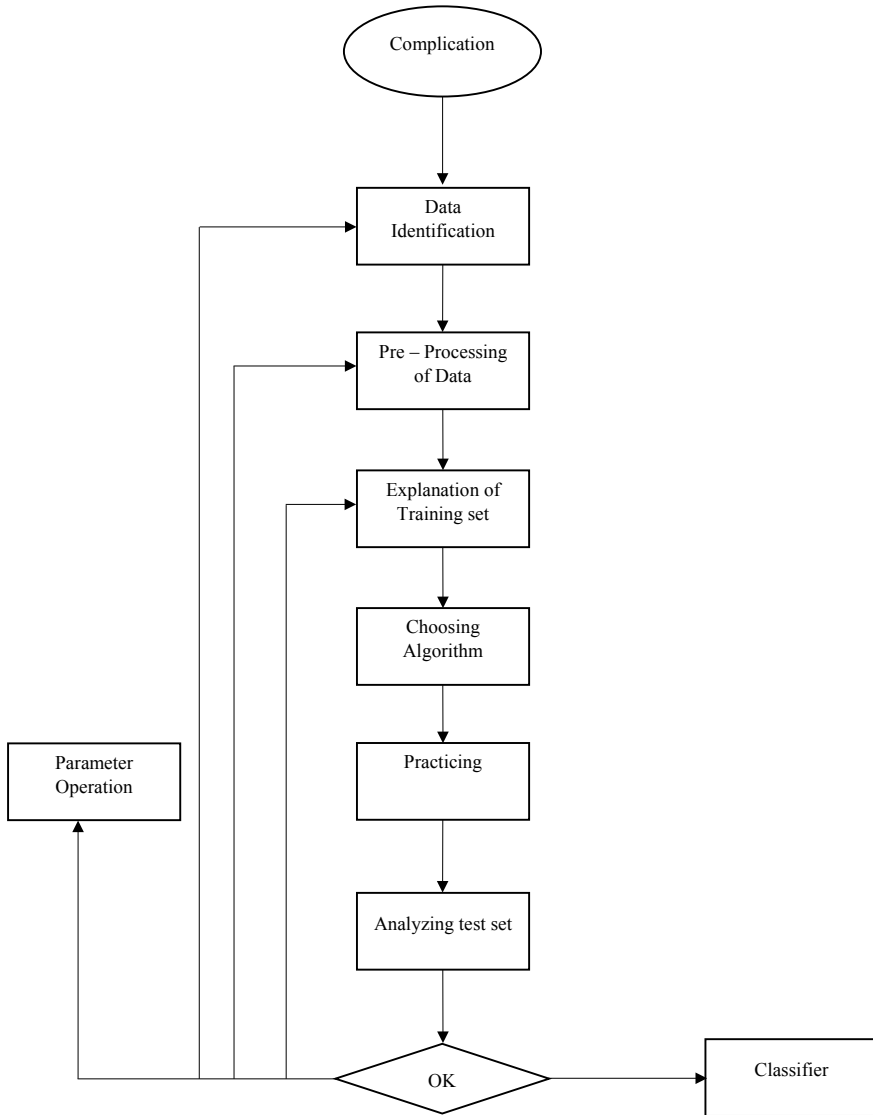


Fig. 8 Process of machine learning

Linear Regression

It is used to estimate real values. We establish relationship between independent and dependent variables by a fitting a best line. The classifiers of these algorithms are

$$\text{Precision} = \frac{\text{FP}}{\text{TP} + \text{FP}} * 100 \tag{1}$$

$$\text{Recall} = \frac{\text{FN}}{\text{TN} + \text{FN}} * 100 \quad (2)$$

$$F1 = \frac{\text{Precision}}{\text{Recall}} * 100 \quad (3)$$

Decision Tree

Apart from other supervised machine learning algorithms, decision tree algorithm solves regression and classification problems. The main of this algorithm is to create a training model that predicts the value of the target variables by simple decision rules. The classifiers of these algorithms are

$$\text{Particularity} = \frac{\text{TN}}{\text{FP} + \text{TN}} * 100 \quad (4)$$

$$\text{Susceptibility} = \frac{\text{TP}}{\text{TP} + \text{FN}} * 100 \quad (5)$$

$$F1 = \sqrt{\text{Particularity} * \text{Susceptibility}} \quad (6)$$

Random Forest

Random forest is a trademark term for an ensemble of decision trees. Random forest consists of collection of decision trees. In order to classify the new object based on attributes, each tree gives a classification, and we say the tree “votes” for the class. The classifiers are

$$F1 = \frac{\text{TP} + \text{TN}}{\text{FP} + \text{FN}} * 100 \quad (7)$$

K-Nearest Neighbor

KNN algorithm stores all available cases and classifies new cases on the basis of similar measures. It is used in statistical estimation and pattern recognition. KNN is also used for classification as well as regression predictive problems. KNN is otherwise called as lazy learning algorithm or non-parametric learning algorithm because it is not having specialized training phase, and it uses all the data for training while classification. The classifiers included in this algorithm are

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} * 100 \quad (8)$$

$$\text{Recall} = \frac{\text{TN}}{\text{TN} + \text{FN}} * 100 \quad (9)$$

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

Logistic Regression

Logistic regression is a classification not a regression algorithm. It is used for estimating discrete values (yes or no) based on given set of independent variables. It predicts output values lies between 0 and 1. The classifiers of these algorithms are

$$\text{Precision} = \frac{\text{TP}}{\text{TN} + \text{FP}} * 100 \quad (11)$$

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FP}} * 100 \quad (12)$$

$$F1 \text{ score} = 2 * \frac{\text{Precision} * \text{Sensitivity}}{\text{Precision} + \text{Sensitivity}} \quad (13)$$

6.1.2 Unsupervised Learning

In unsupervised machine learning, everything is in order. The main aim is to make the computer learns something which we cannot teach it and the computer is not instructed the process of doing. Unsupervised machine learning has two approaches. In the first approach, the agent is taught to use rewards system to denote success. The second approach of unsupervised machine learning algorithm is clustering. This approach finds similarities in the training data. Data is driven in this approach. In the view of Ghahramani, unsupervised algorithm is designed to receive structures from the data samples. Some of the unsupervised machine learning algorithms are

Apriori Algorithm

The Apriori algorithm uses frequent item sets. It produces association rules, and it is made to work on the databases which contain transaction. It is with the cooperation of association rule, it determines the strength or weakness of two connected objects. This algorithm uses breadth-first search and hash tree for calculation of item set association efficiently. It is an iterative process for finding the frequent item sets from the large dataset. The main use of this algorithm is market basket analysis. It helps to find products which can be bought together. It is also used in the field of healthcare. The different steps used in this algorithm are

Step 1: Determining the support of item sets in the transactional database.

Step 2: All higher support value or lower support values are taken into account.

Step 3: Determine the rule of these subsets that have higher values than threshold or lower confidence.

Step 4: Sort out the rules in decreasing order of lift.

K-Means

K-means clustering belongs to unsupervised machine learning algorithm. It is used to clear the problem relating to clustering in machine learning. It forms the unlabeled dataset into different groups. It is a centroid-based algorithm. The clusters related with centroids. Its main aim is to reduce the distance between the data point and their corresponding clusters. It takes the unlabeled dataset as input and separates the dataset into *k*-number of clusters. The process is repeated till the best clusters are identified. *K*-Values is predetermined here. There are two main tasks that this algorithm performed.

- Determining best value for the *k* center points.
- Each data points are given nearest to *k*-center. The points nearer to the *k*-center forms clusters.
- **Step 1:** *K* is selected to decide total number of clusters.
- **Step 2:** Choose random *K* points.
- **Step 3:** Each center points is assigned near to centroids, which performs earlier *K* clusters.
- **Step 4:** The variance is determined and fix a new centroid of every cluster.
- **Step 5:** The third step is continued.
- **Step 6:** If there is any reassignment appears, go to 4th step to complete.

6.1.3 Reinforcement Learning

In this algorithm, the machine takes specific decisions. It works by exposing itself to an environment by training itself by using trial and error method. The machine knows from past experience and get more information and thereby accurate decision is taken by the machine. Reinforcement learning differs from supervised machine learning algorithm. In supervised learning, the training data has the key answer. In reinforcement learning, no answer is found whereas the reinforcement agent determines what to do next. Reinforcement learning divided into two types. They are positive and negative. We can explain positive reinforcement learning as when an event occurs because of particular behavior which increases the strength of the behavior. Negative reinforcement learning strengthens the behavior because the condition of the negative is avoided. We can use reinforcement learning in the field of robotics and for industrial automation, machine learning and data processing. It is also used to create training system which gives custom instruction and material distribution according to the student's strength.

7 Conclusion

Presently, elderly population faces the problem of falling all over the world. They have to be saved. It is because of new technological development elders are rescued from their risk of falling. Rehabilitation is one of the processes by which elder's disability, injuries are healed. Physiotherapist plays a vital role in this aspect. Kinect is a low-cost device helps the elderly people to do rehabilitation exercises suggested by the physiotherapist in the homely environment. It tracks the human movement of the skeleton joints. There are many causes for elders to fall. Rehabilitation makes their elders to get from the bed and do their daily activities independently. Some of the techniques used in machine learning are statistics and computer science. Supervised machine learning algorithms are useful in many ways to detect the fall of the elders.

References

1. S. Tiangang, L. Zhou, D. Xinyang, W. Yi, 3D surface reconstruction based on Kinect, in *2013 IEEE Third International Conference on Information Science and Technology (ICIST)* (2013), pp. 986–990. [Online]. Available at: <https://ieeexplore.ieee.org/document/6747702>. Accessed 30 Jan 2019
2. E. Auvinet, F. Multon, A. St-Arnaud, J. Rousseau, J. Meunier, Fall detection using body volume reconstruction and vertical repartition analysis, in *International Conference on Image and Signal Processing* (2010), pp. 376–383
3. E. Auvinet, F. Multon, A. St-Arnaud, J. Rousseau, J. Meunier, Fall detection with multiple cameras: an occlusion-resistant method based on 3D silhouette vertical distribution. *IEEE Trans. Inf. Technol. Biomed.* **15**, 290–300 (2011)
4. H. Powell, M.A. Hanson, J. Lach, A wearable inertial sensing technology for clinical assessment of tremor, in *2007 IEEE Biomedical Circuits and Systems Conference (BIOCAS)* (IEEE, 2007), pp. 9–12
5. J. Large, N. Gan, D. Basic, N. Jennings, Using the timed up and go test to stratify elderly inpatients at risk of falls. *Clin. Rehabil.* **20**(5), 421–428 (2016)
6. M.-C. Shih, R.-Y. Wang, S.-J. Cheng, Y.-R. Yang, Effects of a balance-based exergaming intervention using the kinect sensor on posture stability in individuals with parkinson's disease: a single-blinded randomized controlled trial. *J. NeuroEng. Rehabil.* **13**(1), 78 (2016)
7. G. Mastorakis, D. Makris, Fall detection system using Kinect's infrared sensor. *J. Real-Time Image Proc.* **9**(4), 635–646 (2014)
8. E. Liberty, K. Lang, K. Shmakov, Stratified sampling meets machine learning, in *International Conference on Machine Learning* (2016), pp. 2320–2329
9. T. Frenken, B. Vester, M. Brell, A. Hein, aTUG: fully-automated timed up and go assessment using ambient sensor technologies, in *2011 5th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth)* (2011), pp. 55–62
10. P. Paul, T. George, An effective approach for human activity recognition on smartphone, in *2015 IEEE International Conference on Engineering and Technology (ICETECH)* (2015), pp. 1–3
11. E.E. Stone, M. Skubic, Fall detection in homes of older adults using the Microsoft Kinect. *IEEE J. Biomed. Health Inform.* **19**(1), 290–301 (2015)
12. J.S. Madhubala, A. Umamakeswari, A vision based fall detection system for elderly people. *Indian J. Sci. Technol.* **8**, 167 (2015)
13. Z.A. Mundher, J. Zhong, A real-time fall detection system in elderly care using mobile robot and Kinect sensor. *Int. J. Mater. Mech. Manuf.* **2**(2), 133–138 (2014)

14. B. Mirmahboub, S. Samavi, N. Karimi, S. Shirani, Automatic monocular system for human fall detection based on variations in silhouette area. *IEEE Trans. Biomed. Eng.* **60**, 427–436 (2013)
15. H.M. Hondori, M. Khademi, A review on technical and clinical impact of Microsoft Kinect on physical therapy and rehabilitation (2014)
16. J.-H. Shin, S.B. Park, S.H. Jang, Effects of game-based virtual reality on health-related quality of life in chronic stroke patients: a randomized, controlled study. *Comput. Biol. Med.* **63**, 92–98 (2015)
17. V. Bevilacqua et al., Fall detection in indoor environment with Kinect sensor, in *2014 IEEE International Symposium on Innovations in Intelligent Systems and Applications (INISTA) Proceedings* (IEEE, 2014)
18. Developer resources/Kinect hardware, Kinect for Windows SDK 2.0/Kinect hardware key features and benefits/Microsoft (2014)
19. Microsoft research/Teaching Kinect for Windows to Read Your Hands/direction in the evolution of Kinect for Windows. TechFest (2013)
20. T. Wei, Y. Qiao, B. Lee, Kinect skeleton coordinate calibration for remote physical training, in *Proceedings of the International Conference on Advances in Multimedia (MMEDIA)* (2014)
21. M. Alnowami et al., Feasibility study of markerless gait tracking using Kinect. *Life Sci. J.* (2014)
22. W. Zhao et al., A Kinect-based rehabilitation exercise monitoring and guidance system, in *2014 5th IEEE International Conference on Software Engineering and Service Science (ICSESS)* (IEEE, 2014)
23. H.-T. Chen et al., Computer-assisted self-training system for sports exercise using Kinects, in *2013 IEEE International Conference on Multimedia and Expo Workshops (ICMEW)* (IEEE, 2013)
24. S. Saha et al., A study on leg posture recognition from Indian classical dance using Kinect sensor, in *2013 International Conference on Human Computer Interactions (ICHCI)* (IEEE, 2013)

A New Approach for Optical Image Encryption Standard Using Bit Swapping and Fractional Fourier Transform



L. Anusree and M. Abdul Rahiman

Abstract Because of the exponential evolution of digital knowledge, cloud, Internet of things (IoT) technologies, personal information protection and security have gained increased attention. A novel scheme for optical encryption of two-dimensional images is proposed in this paper by integrating image bit swapping strategies in fractional Fourier domains. The secret data hiding process is performed based on LSB replacing technique using secret bits. Image encryption is performed in fractional Fourier transform (FRFT) domain blocks based on the random selection. 3D bit swapping technique is used to perform swapping of randomly selected FRFT blocks. To perform complete encryption, entire blocks will be collapsed based on the randomly generated keys. To evaluate the performance, standard performance measures such as peak signal-to-noise ratio (PSNR) of 41.16, mean square error (MSE) of 0.0056, and correlation coefficient (CC) of 0.99 are presented in various noise conditions. From the performance measures, it is clear that this work achieves better quality when compared to classical techniques.

Keywords Scramble · Encryption · Fractional Fourier transform · Optical information · Security · Least significant bit

1 Introduction

Given the large volumes of data exchanged in digital networks, information security is an increasing issue. This has given rise to a variety of cryptographic schemes designed to ensure that unauthorized users cannot access the information. Among these cryptographic schemes are those that make use of optical system properties [1]. Following the invention of the double random phase encoding by Réfrégier and Javidi, so many optical information encryption protocols are being suggested double random phase encoding (DRPE). DRPE-related strategies like fractional Fourier

L. Anusree (✉)

LBSITW, Thiruvananthapuram, Kerala, India

M. Abdul Rahiman

LBSCST, Thiruvananthapuram, Kerala, India

domain DRPE and Fresnel domain DRPE have been used in optical information encryption to date [2].

DRPE indicates that the encrypted image is stagnant normal distribution if the two-phase masks have significant differences. The DRPE solution was later proposed to the Fresnel domain and fractional Fourier domain due to its notable advantages, such as large keyspace and stability toward blindness encryption operation. Even so, it has two clear flaws that prevent it from being used for longer. The first is its susceptibility to multiple threats, which stems from the DRPE scheme's linearity. As a result, an increasing number of people are focusing on designing nonlinear optical encryption schemes [3].

Cryptanalysis of optical encryption was successful, demonstrating that the DRPE was vulnerable to a variety of attacks, including chosen plain text (CPA), chosen cipher text (CCA), and known plain text (KPT). The vast majority of these functions can be accomplished with a single-image encoding. Recently, a novel methodology based on the convergence of sparse strategy and image encoding for two times was created. The integration allowed for an improvement in signal hiding capability while reducing the amount of the same data. Furthermore, these systems used sophisticated knowledge to execute encryption and decryption procedures that necessitated the fabrication of complex optical components [4].

The fractional Fourier transform is a general statement of the order parameter Fourier transform. FRFT extends this differential operator by allowing it to be dependent on a control variable. A function's Fourier transform (FT) can be analyzed. The sequence FRFT is the strength of the FT operator, and when an is 1, FRFT becomes FT. Because of its approximation among the function part and associated Fourier domain, the FRFT has been widely used in digital signal processing, image processing, optical encoding, and quantum optics. FRFT has recently been used in optical cryptography to encrypt hyperspectral data and multiple images. It can significantly decrease the number of private key bits exchanged between allowed users by incorporating FRFT into a CGI-based optical encryption scheme [5].

The rest of this work is organized as follows: The reported optical encryption methods are presented in Sect. 2. The proposed optical encryption technique is discussed in Sect. 3. The result and discussion of the proposed method, comparative studies, and analysis are explained in Sect. 4. At last, discussed conclusion in Sect. 5.

2 Literature Survey

A vast amount of works have been proposed previously to implement the strategy for optical encryption. These different methods are targeting to reduce the complexity of design by optimizing the architecture of the algorithm. This section presents some of the works which are proposed previously to perform the implementation of optical encryption.

Zhang et al. proposed deep learning-based optical image cryptography and authentication transmitting mechanism, and JPEG first normalizes the image to produce a properly organized, and the components are distributed as an image to optimize data transmission; ghost imaging, which employs a point-to-face data transfer method to mitigate the impact of unstable medium and movement on the contact channel and combines it with machine learning can improve recovered image quality; Then, for transmission, ghost imaging is used to improve anti-interference capabilities and transmission control. Lastly, the problem of decreased image quality after ghost-imaging processing is solved through using neural networks for reconstruction, which improves image resolution. The modeling experiment included analyses of viability, stability, and robustness, and the experimental findings were statistically analyzed using methods of correlation such as PSNR, reciprocal information, and so on [6].

Chen et al. proposed an optical hyperspectral image encryption algorithm that encrypts both spatial and spectral information at the same time using sophisticated Chirikov mapping and the gyrator transform. The initial hyperspectral image is translated to binary format before being expanded into a one-dimensional sequence. After which, a place list is created using an improved Chirikov mapping, in which the image's binary sequence can be scrambled according to the position sequence. The image is then scrambled and shared before being converted with the gyrator transform. With the gyrator transform parameters and the enhanced encryption algorithm, the step data acts as the primary key Chirikov mapping serving as secondary keys to increase security [7].

Liu et al. proposed an optical encryption system based on a joint transform correlator (JTC) that allows for parallel encoding of multi-channel images. After which, a place list is created using an improved mapping of Chirikov, in which the image's binary sequence to the position sequence. The image is shared before being converted with the gyrator transform. With the gyrator transform parameters and the enhanced encryption algorithm, the step data acts as the primary key. Using optimized phase masks to confine to a specific area and isolate multiple joint power spectrum (JPS) by regulating the location of a single JPS with linear step shifts to prevent cross-talk among multi-channel images. Both of these procedures are carried out by refining and constructing phase masks that must be modified on the spatial light modulator (SLM), and as a result, there is a feasible optical implementation that does not require any extra optical encryption or complexity [8].

Chen and Chen proposed a novel method focused on iterative phase retrieval that was suggested in interference-based optical encryption for concurrently obtaining two phase-only masks with silhouette removal. In encryption, the proposed process retrieval algorithm needs only a few variations and also no external keys or compatible techniques. During the ongoing process, the two phase-only masks are variously modified; however, neither of the phase-only masks are set during the encoding g process [9].

Li et al. proposed a completely optical image-encoding algorithm that depends on compressive sensing the cover image firstly encoded to a Caucasian stationary noise sequence in a Mach-Zehnder interferometer using a dual random encoding technique. Using single-pixel compressive holographic projection, after that, the

ciphertext is heavily compressed to an optical domain signal. The encrypted image is recovered at the receiving terminal using a compressive sensing technique, and the cover image decoded using three repeated valid keys [10].

Akhshani et al. proposed a two-dimensional piecewise nonlinear chaotic map hierarchy with an unchanging measure. Such maps contain intriguing properties such as invariant measure, ergodicity, and the ability to calculate K-S entropy. Then, by utilizing important features of these chaotic maps such as ergodicity, sensitivity to initial condition and control parameter, one-way computation, and random like activity [11].

Jolfaei and Mirghadri presented a novel picture encryption method that combines pixel shuffling and AES. Through S-box architecture, the chaotic baker's map is employed for shuffling and increasing AES efficiency. In the image, chaos causes dispersal and confused growth. Because of its sensitivity to beginning circumstances, the chaotic baker's map has great promise for constructing dynamic permutation maps and S-boxes [12].

Eisa and Sathesh proposed based on the voltage-driven pattern, it consists of an adjacent drive technique, cross method, and an alternate opposite current direction approach. The technique of assessing biomedical electrical impedance tomography (EIT) and investigating the impedance imaging of existing substances. The importance of the alternate opposing current direction approach in the biomedical system and the EIT image reconstruction techniques [13].

Manoharan proposes and implements an effective chaotic-based biometric authentication method for the cloud's user interface layer. This approach employs the fingerprint as the biometric feature and differs from traditional methods in that it employs an N-stage Arnold Transform to safely validate the claim of the so-called genuine user [14].

From the above discussion, it is very clear that previously several works have been proposed to perform to improve the robustness. The main disadvantage of the previous works is reduced quality and security. The following is the primary goal of this work:

1. To increase the robustness of optical encryption.
2. To preserve the quality of the image.
3. To develop a technique that is flexible for real-time application development.

The details of the implementation of the proposed work are given below sections.

3 Optical Image Encryption Standard Using Bit Swapping and Fractional Fourier Transform

In this work, the cover image is converted to a bitplane, extract the LSB bits, and replace with these LSB bits to combine. After combine 3×3 blocks are obtained and then swapping operation is performed then combine all the blocks to FRFT then swapping using alpha also random secret pattern matrix multiplication is performed

then swapping using beta finally the encrypted image is obtained. The reverse process is applied for the decryption purpose; first, the encrypted image is swapped using beta and then secret matrix multiplication and swap using alpha, then doing IFRT and convert to 3×3 matrix and then 3D bit swapping operation is performed. Bitplane conversion is used to extract the LSB bits, and then secret text image is converted from the image to text conversion and the secret text is extracted from the image.

3.1 Bitplane Extraction from an Image

An image is simply made up of individual pixel data. When an image is 256×256 in size, it means that in total, there are 256×256 pixels, and each pixel contains information about the image. Grayscale images are simply black and white images. Each pixel in a grayscale image has a value between 0 and 255, indicating where the image can be black or white. If the pixel value is 0, the color is completely black; if the pixel value is 255, the color can be completely white; and pixels with intermediate values have colors of black and white. Figure 1 shows the block diagram of optical image encryption standard using bit swapping and fractional Fourier transform. In this work, to generate a grayscale image, since the pixel size of a grayscale image ranges from 0 to 255, the detail is stored in 8 bits. As a result, it can split the image into eight planes. Binary images are those of which the pixel value may either be 0 or 1.

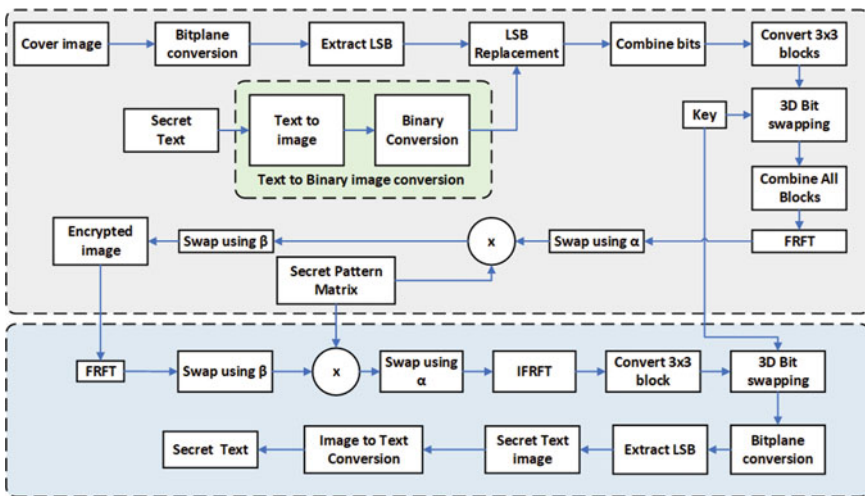


Fig. 1 Block diagram of optical image encryption standard using bit swapping and fractional Fourier transform

3.2 Bitplane Conversion

A bitplane of a digital discrete signal for an image is a group of bits that correspond to a specific bit location in each of the binary numbers that describe the signal. Grayscale images are almost the same as binary images. Each pixel in a gray image can have a value between 0 and 255, which defines where the image would be the gray image. If the pixel value is 0, the color is black; if the pixel value is 255, the color is completely white; and pixels with intermediate values have black and white colors. Figure 2 displays Lena's image of the bit conversion. Figure 3 shows 3D block swapping for Lena image. Figure 4 shows Lena image bitplane extraction.

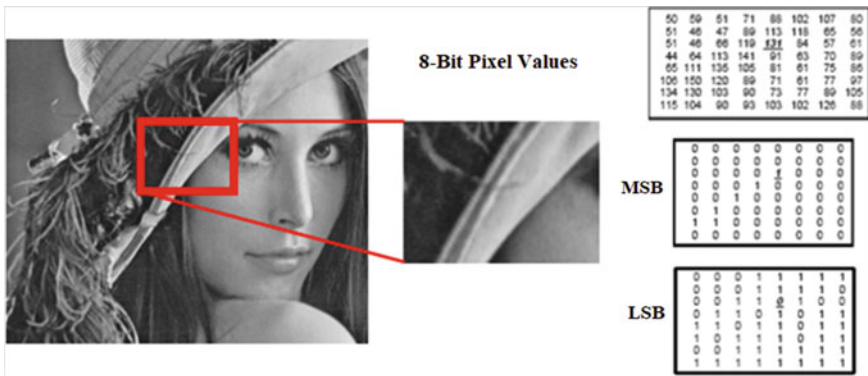
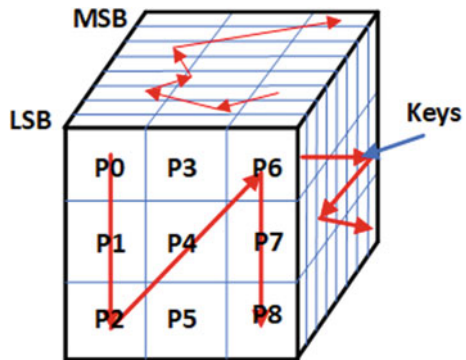


Fig. 2 Bit conversion of Lena image

Fig. 3 3D Block swapping



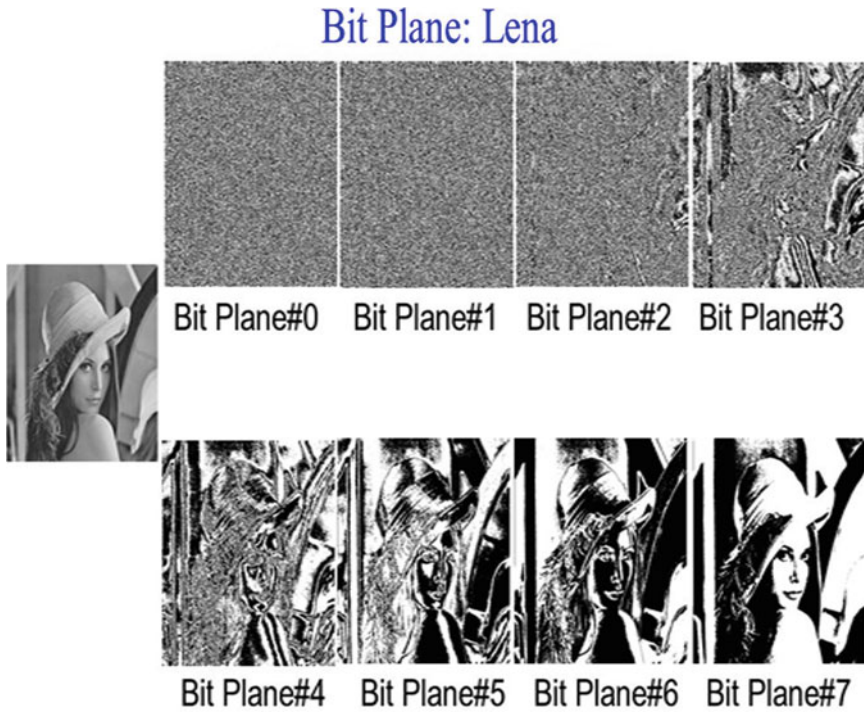


Fig. 4 Bit plane extraction of Lena image

3.3 Extract LSB

In the case of an 8-bit representation, the gray image of bitplane 7 is created to run the input plain text image through an adaptive threshold gray-level transformation algorithm, which maps all the levels among 0 and 127 to one level and all levels starting from 129 to 253 to the other.

3.4 LSB Replacement

The LSB solution substitutes the host image's least important bits with hidden data bits. Based on the image pixels of LSB and the hidden data bits, this LSB method raises or lowers the pixel value by one. The resolution of Lena's host image is (256 × 256 pixels), following the substitution of LSB of every pixel with the message bits.

3.5 Fractional Fourier Transformation

The FRFT is a kind of normal distribution that extends the Fourier transform. The Fourier transform is a linear transformation that allows a signal collected in the position or time domain to be rotated via 90-degree radians into the orthogonal frequency domain or spatial frequency domain. It may be demonstrated that four consecutive Fourier transform applications are similar to the function of identity. It is analogous to the Fourier transform to the n -th number, except it does not have to be an integer, and it is a function transform to some mediatory part between frequency and time. Filter architecture, signal processing, step retrieval, and pattern recognition are among its applications [11]. The FRFT is a process of defining fractional convolution, interaction, and other expressions, as well as the linear canonical transformation (LCT). A random permutation is used to scramble one primitive image. A DRPE scheme is used to encrypt the synthesized complex-valued signal. The proposed scheme is particularly vulnerable to fractional orders of FRFTs, but it is also resistant to data loss and noisy attacks. However, in Tao's algorithms, the amplitude-based image has a smaller main space than the phase-based image, making the amplitude-based image easier to decode.

The FRFT is a common mathematical technique due to its simple optical implementation and a wide variety of applications. The FRT domains theorem forms the foundation of our scheme. A continuum of domains known as fractional domains occur between the generally examined time and frequency domains, which are often found in signal processing (optics). The FRFT function can be expressed mathematically as

$$\hat{f}(\mathfrak{K}) = \int_{-\infty}^{\infty} f(i)e^{-2\pi i p \mathfrak{K} x} dx \quad (1)$$

The f is determined by \hat{f} via the inverse transform

$$f(i) = \int_{-\infty}^{\infty} \hat{f}(\mathfrak{K})e^{2\pi i p \mathfrak{K} x} d\mathfrak{K} \quad (2)$$

4 Results and Discussions

This section discusses the outcomes of numerical simulations performed to determine the efficacy and robustness of the suggested solution. This work is done by MATLAB R2020b using a computer with CPU Intel(R) Core(TM) i5-3320 M CPU @ 1.60 GHz, 1.60 GHz, and 4 GB of RAM. The dataset images with size 256×256 pixel with

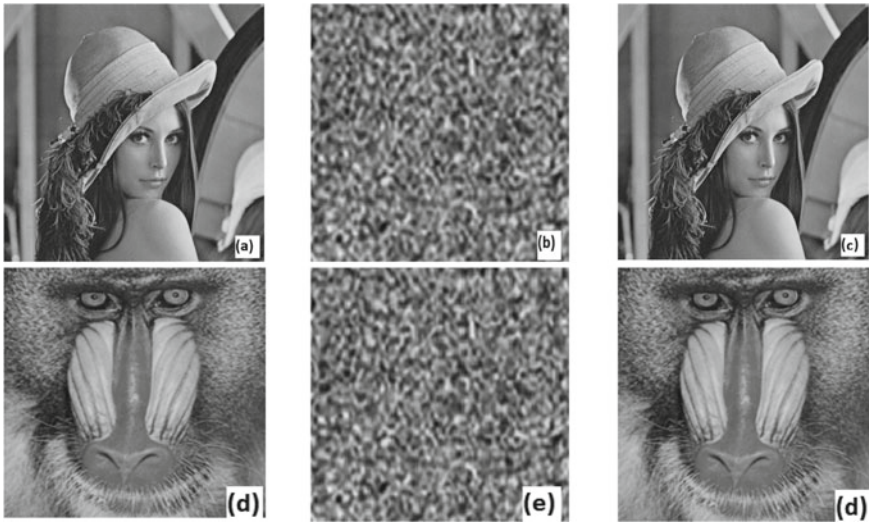


Fig. 5 a, d Original image b, e encrypted image c, f decrypted image

any image format as input sample used as the Lena and mandrill. Figure 5 shows the sample cover image, encrypted image, and decrypted image.

4.1 Mean Square Errors

The MSE is a measure of an estimator’s consistency; it is often non-negative, with values closest to zero being greater. The distinction between the original and decrypted images is represented by MSE [13]. It can be expressed as

$$MSE = \frac{1}{P_x * P_x} \sum_{i=1}^{P_x} \sum_{j=1}^{P_x} |\hat{I}(i, j) - I(i, j)|^2 \tag{3}$$

where $P_x * P_x$ denotes the number of image pixels, $\hat{I}(i, j)$, $I(i, j)$ signify original image values, decrypted image values, and at that pixel value (i, j) .

4.2 Peak Signal-to-Noise Ratio

The PSNR is the proportion of the signal’s maximum potential strength to the power of completely corrupted input, which influences the accuracy of which it is represented [12].

$$\text{PSNR} = 10. \log_{10}(\text{MAX}_{PY}^2/\text{MSE}) \quad (4)$$

$$\text{PSNR} = 20. \log_{10}(\text{MAX}_{PY}/\sqrt{\text{MSE}}) \quad (5)$$

$$\text{PSNR} = 20. \log_{10} \text{MAX}_{PY} - 10. \log_{10} \text{MSE} \quad (6)$$

where MAX_{PY} represents a maximum image pixel value.

4.3 Correlation Coefficient

The correlation coefficient (CC) is a graphical representation of a type of correlation, which is a statistical relationship between these two variables. The variables may be two columns from a given set of data or two components of a quantitative probability distribution with good distribution.

$$\text{CC}(N, n) = \frac{M\{[N - M(N)][n - M(n)]\}}{M\{[N - M(N)]^2\}M\{[n - M(n)]^2\}} \quad (7)$$

Here, F and f represent the plain image and decrypted image.

4.4 Structural Similarity Index Measure (SSIM)

SSIM is a perspective paradigm that treats image loss as a perceived shift in structural details while often integrating core visual effects including intensity of light masking and intensity masking concepts.

$$L(i, j) = \frac{2k_i k_j + r1}{k_i^2 + k_j^2 + r1} \quad (8)$$

$$C(i, j) = \frac{2l_i l_j + r2}{l_i^2 + l_j^2 + r2} \quad (9)$$

$$S(i, j) = \frac{l_{ij} + r3}{l_i l_j + r3} \quad (10)$$

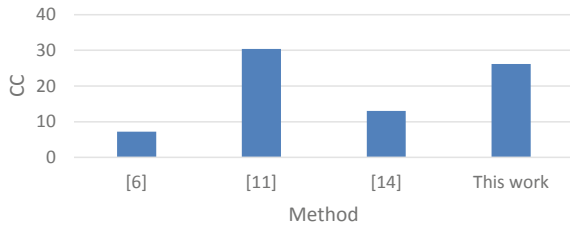
$$\text{SSIM}(i, j) = [L(i, j)^\alpha . C(i, j)^\beta . S(i, j)^\gamma] \quad (11)$$

The weights α, β, γ reduced to 1, the formula can be written as

Table 1 Comparative performance of previous works

Method	CC	PSNR	MSE	SSIM
DI [3]	0.87	–	–	–
GI [6]	–	7.2	–	–
FRFT [11]	–	24.40	–	–
PTDMPFRFT [14]	–	13.03	–	–
JST [15]	0.98	–	–	–
This work	0.99	41.16	0.0054	0.98

Fig. 6 Comparative performance of CC



$$SSIM(i, j) = \frac{(2k_i k_j + r1)(2l_{xy} + r2)}{(k_i^2 + k_j^2 + r1)(l_i^2 + l_j^2 + r2)} \tag{12}$$

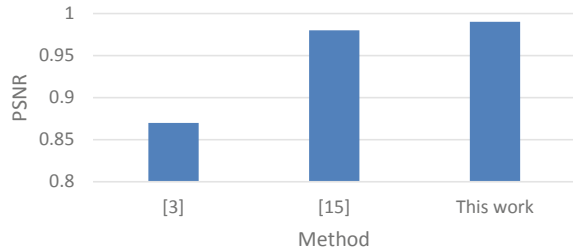
where k_i is the average of i , k_j is the average of j , k_i^2 is the variance of i , k_j^2 is the variance of j , L, C, S are luminance, contrast, and structure, $r1, r2, r3$ are contrasted level, α, β, γ are weighted combinations.

From Table 1, the better comparative performances are CC, PSNR, and MSE in previous works. This work has higher CC, PSNR, and lower MSE when compared with other previous methods. Diffractive imaging (DI) returns only a CC value like 0.87. FRFT has PSNR 24.40. The ghost imaging (GI) method returns 7.2 PSNR. The phase-truncated discrete multiple-parameter fractional Fourier transform (PTDMPFRFT) method has 13.03 PSNR only. Jigsaw Transform (JST) has 0.98 CC and which is 0.01 lower than the proposed method. This new approach for optical image encryption standard using bit swapping and fractional Fourier transform improve 0.1% from previous existing better method. Figure 6 shows comparative performance of CC, and Fig. 7 shows comparative performance of PSNR.

5 Conclusion

A highly secure optical image encryption standard is proposed in this work for highly secure image transmission process. Results analysis of this works shows the efficiency of this work when compared with the conventional works. It has also been

Fig. 7 Comparative performance of PSNR



discovered that the current algorithm's performance is proportional to the FRFT order. Furthermore, the FRFT's privacy efficiency. It is possible to study a DRPE-based optical cryptosystem and demonstrate that this optical encryption scheme has a security weakness due to the FRFT's simplicity. It has been shown that a research approach can take two fractional-order keys and that the phase extraction strategy in the FRFT domain can obtain two-phase keys. From the result analysis, it is observed that obtained PSNR is 41.16 and MSE is 0.0054 which is better when compared with conventional works.

References

1. A.V. Zea, J.F. Barrera, R. Torroba, Experimental optical encryption of grayscale information. *Appl. Opt.* **56**(21), 5883–5889 (2017)
2. D. Peng, Z. Huang, Y. Liu, Y. Chen, F. Wang, S.A. Ponomarenko, Y. Cai, Optical coherence encryption with structured random light. *PhotonIX* **2**(1), 1–15 (2021)
3. Y. Qin, Q. Gong, Z. Wang, Simplified optical image encryption approach using single diffraction pattern in diffractive-imaging-based scheme. *Opt. Express* **22**(18), 21790–21799 (2014)
4. E.A. Mohammed, H.L. Saadon, Sparse phase information for secure optical double-image encryption and authentication. *Opt. Laser Technol.* **118**, 13–19 (2019)
5. S. Zhao, X. Yu, L. Wang, W. Li, B. Zheng, Secure optical encryption based on ghost imaging with fractional Fourier transform. *Opt. Commun.* **474**, 126086 (2020)
6. L. Zhang, R. Xiong, J. Chen, D. Zhang, Optical image compression and encryption transmission-based on deep learning and ghost imaging. *Appl. Phys. B* **126**(1), 1–10 (2020)
7. H. Chen, C. Tanougast, Z. Liu, W. Blondel, B. Hao, Optical hyperspectral image encryption based on improved Chirikov mapping and gyrator transform. *Opt. Lasers Eng.* **107**, 62–70 (2018)
8. J. Liu, T. Bai, X. Shen, S. Dou, C. Lin, J. Cai, Parallel encryption for multi-channel images based on an optical joint transform correlator. *Opt. Commun.* **396**, 174–184 (2017)
9. W. Chen, X. Chen, Iterative phase retrieval for simultaneously generating two phase-only masks with silhouette removal in interference-based optical encryption. *Opt. Commun.* **331**, 133–138 (2014)
10. J. Li, J.S. Li, Y.Y. Pan, R. Li, Compressive optical image encryption. *Sci. Rep.* **5**(1), 1–10 (2015)
11. A. Akhshani, S. Behnia, A. Akhavan, H. Abu Hassan, Z. Hassan, A novel scheme for image encryption based on 2d piecewise chaotic maps. *Opt. Commun.* **283**, 3259–3266 (2010)
12. A. Jolfaei, A. Mirghadri, in *An Applied Imagery Encryption Algorithm Based on Shuffling and Baker 39 Maps*. Proceedings of the 2010 International Conference on Artificial Intelligence and Pattern Recognition (AIPR-10), Florida, USA, pp. 279–285 (2010)

13. E.E.B. Adam, Survey on medical imaging of electrical impedance tomography (EIT) by variable current pattern methods. *J. ISMAC* **3**(02), 82–95 (2021)
14. J.S. Manoharan, A novel user layer cloud security model based on chaotic arnold transformation using fingerprint biometric traits. *J. Innovative Image Process. (JIIP)* **3**(01), 36–51 (2021)
15. D. Zhang, X. Liao, B. Yang, Y. Zhang, A fast and efficient approach to color-image encryption based on compressive sensing and fractional Fourier transform. *Multimedia Tools Appl.* **77**(2), 2191–2208 (2018)
16. H. Chen, X. Du, Z. Liu, Optical spectrum encryption in associated fractional Fourier transform and gyrator transform domain. *Opt. Quant. Electron.* **48**(1), 1–16 (2016)
17. L. Sui, H. Lu, X. Ning, Y. Wang, Asymmetric double-image encryption method by using iterative phase retrieval algorithm in fractional Fourier transform domain. *Opt. Eng.* **53**(2), 026108 (2014)
18. H. Li, X. Bai, M. Shan, Z. Zhong, L. Liu, B. Liu, Optical encryption of hyperspectral images using improved binary tree structure and phase-truncated discrete multiple-parameter fractional Fourier transform. *J Opt.* **22**(5), 055701 (2020)
19. W. El-Shafai, I.M. Almomani, A. Alkhayer, Optical bit-plane-based 3D-JST cryptography algorithm with cascaded 2D-FrFT encryption for efficient and secure HEVC communication. *IEEE Access* **9**, 35004–35026 (2021)

Ensemble Model Ransomware Classification: A Static Analysis-based Approach



Shanoop Johnson, R. Gowtham, and Anand R. Nair

Abstract The growth of malware attacks has been phenomenal in the recent past. The COVID-19 pandemic has contributed to an increase in the dependence of a larger than usual workforce on digital technology. This has forced the anti-malware communities to build better software to mitigate malware attacks by detecting it before they wreak havoc. The key part of protecting a system from a malware attack is to identify whether a given file/software is malicious or not. Ransomware attacks are time-sensitive as they must be stopped before the attack manifests as the damage will be irreversible once the attack reaches a certain stage. Dynamic analysis employs a great many methods to decipher the way ransomware files behave when given a free rein. But, there still exists a risk of exposing the system to malicious code while doing that. Ransomware that can sense the analysis environment will most certainly elude the methods used in dynamic analysis. We propose a static analysis method along with machine learning for classifying the ransomware using opcodes extracted by disassemblers. By selecting the most appropriate feature vectors through the tf-idf feature selection method and tuning the parameters that better represent each class, we can increase the efficiency of the ransomware classification model. The ensemble learning-based model implemented on top of N -gram sequence of static opcode data was found to improve the performance significantly in comparison to RF, SVN, LR, and GBDT models when tested against a dataset consisting of live encrypting ransomware samples that had evasive technique to dodge dynamic malware analysis.

Keywords Tf-idf · N -gram · Random forest · Opcode · SVM · Ransomware · Voting classifier

S. Johnson (✉) · R. Gowtham · A. R. Nair
Department of Computer Science and Engineering, Amrita School of Engineering, Amrita
Vishwa Vidyapeetham, Coimbatore, India
e-mail: cb.en.p2cse19021@cb.students.amrita.edu

R. Gowtham
e-mail: r_gowtham@cb.amrita.edu

A. R. Nair
e-mail: r_anand@cb.amrita.edu

1 Introduction

Around three hundred and fifty-seven million new malware samples were revealed by early 2017, according to the Internet Security Threat Report conducted by Symantec Broadcom [1]. Malicious software (malware) is capable of eroding or stealing data and compromising computer systems. The earliest form of malware was observed to have appeared in the beginning of the 1970s. The Creeper Worm in the 70s was a program that replicates itself and attaches itself in the remote system and locks it down [2]. The tremendous leap in technological advancements has contributed to massive growth in malware risks with respect to the aspects such as magnitude, numbers, type of operation. The wide-scale development in interconnected devices like smartphones, systems, and IoT devices allows smart and sophisticated malware to be developed. In 2013, more than 250,000 computers were infected with the CryptoLocker ransomware, which then dramatically increased to 500,000 devices during 2014 [3]. In 2017, with an average attack growth rate of 350%, global organizations financial losses due to this malware attack surpassed five billion dollars [4]. By the year 2021, it is expected to reach 20 billion dollars [5]. Global ransomware security agencies noted that the existing solutions are not successful in protecting against attacks by ransomware and suggested the need for active solutions to be developed.

Ransomware analysis attempts to give information on the functionality, intent, and actions of a specific type of software. Two kinds of malware analysis are available: static analysis and dynamic analysis. Static analysis implies analyzing the source code and related artifacts of an executable or malignant file without execution. Dynamic analysis, on the contrary, requires analyzing the behavioral aspects of an executable in action by running it. Both approaches have their own advantages and pitfalls which we need to identify. Static analysis is quicker, but the method could prove to be ineffective if ransomware is effectively hidden using code obfuscation techniques. Machine learning approaches to detect such variants are still evolving. However, conventional identification and analysis of ransomware are hardly able to keep pace with the latest variants of evolved malware and their attacks.

Compared to all other malware attacks, ransomware attacks are time-sensitive. We need to identify and establish the files behavior within seconds in order to prevent havoc to the system. Unlike other attacks, if we allow the ransomware attack to manifest and then monitor and remove the ransomware, the effect of the attack will be irreversible as is the intent of the attack. There are several dynamic analysis models available at present, and Windows Defender service identifies a whole part of it within its databases extensively. However, fresh attacks are seemingly hard to detect as it might not correspond to any existing signature. For this, we need to establish the files intent based on the opcode it executes in the system. We believe if we can train the model with a satisfactorily large enough number of samples, the effect of code obfuscation techniques and random opcode injection into the executable files employed by the malware in order to confuse the model can sufficiently be overcome.

We have identified that static analysis of opcodes extracted from ransomware samples using disassemblers can perform well in classification. Compared to the tedious dynamic analysis setup, our method is easily operable. We use open-source disassemblers to get the disassembled opcode sequences corresponding to the ransomware executable. This study proposes a better ransomware detection model established on static analysis of opcode sequences, which offers more efficient countermeasures against ransomware attacks by detecting them early. The paper proposes a binary classification of benign and malignant samples.

2 Related Work

Since ransomware is a particular genre of malware, to give more insight to the scope of our research, several pieces of research on malware are also cited in this section. First, we present the most recent research initiatives that focus on time-sorted application programming interface (API) call sequences and opcodes on the classification of malware families, which are comparable to the classification methodology that we propose.

In the early stages of malware detection and mitigation, signature-dependent detection methods were suggested. At that point in time, it was assumed that automatic generation of signatures of malware as much as possible was necessary, and that it would increase the pattern-matching speed and performance [6, 7].

With respect to dependence on time-sorted API call sequences, in order to accomplish the malicious purpose intended by the malware, a particular sequence of API calls, in that very order must be executed by attackers, and a study of time-sorted API call sequences will help us better understand the intent of malicious codes [6]. A. Kharrazand and S. Arshad suggested methods based on sequences of API such as dynamic analysis in a sandbox setting, and Kwon et al. [8] extracted time-sorted API call sequences as characteristics and then using RNN to classify and so on.

Hansen et al. [9] considered time-sorted API call sequences as features. The average weighted F1-measure came out around eighty percent. Pekta [8] took N -grams as features from time-sorted API call sequences and measured term frequency-inverse document frequency (tf-idf). The highest accuracy obtained was 92.5%, suggesting that in terms of selecting meaningful sequences, N -grams have better ability to pinpoint on malicious behavior. Some malware could, however, mostly fingerprint the dynamic analysis environment [8] and enforce steps, such as code stalling [10] to avoid detection by an AV engine or human, making research more strenuous. So, dynamic analysis does not detect malware/ransomware that can identify the system or the sandboxing environment in which its being tested, which is the drawback of dynamic malware analysis. Gowtham Ramesh and Anjali Menen proposed a finite-state machine model [11] to dynamically detect ransomware samples by employing a set of listeners and decision-making modules which identify changes in the system within the specific set of use cases defined in the underlying system.

Samples of ransomwares belonging to seven families and samples of applications (apps) associated to categories within the benign class were used in their technique to perform basic classification [12]. Not only did the system predicts whether the sample belonged to the safe class or the malignant class, but also grouped the sample into the corresponding family. During dynamic analysis, the authors considered almost one hundred and thirty-one time-sorted API call sequences obtained by executing the samples in a sandbox and leveraged deep learning to build a model and tested it. The experiment, thus, conducted outperformed almost all the existing models based on machine learning algorithm classifiers. The precision was almost 98% in multi-layer perceptron (MLP); the strongest true positivity rate (TPR) is just 88.9; and 83.3% is observed for several families of ransomware like CryptoWall. Because of the limitation of dynamic analysis mentioned above, if ransomware is able to identify the sandbox used, researchers will not be able to extract meaningful time-sorted API sequences.

Classifying malwares based on machine learning requires extracting features from malware samples first. For that, we have static and dynamic analysis methods available. Static analysis does not require the sample to be executed. It basically extracts opcodes, scans the format of PE headers, and disassembles the sample. Disassemblers like PEiD [6], IDA Pro [7] have their own databases containing file signatures to identify packers and headers. VirusTotal [8] can detect known malware samples using 43 antivirus engines. In this paper, we disassemble the samples using IDAPro for the purpose of static analysis. In dynamic analysis, the sample which we execute has complete access to the system resources. But, the environment will be a controlled one, probably a sandbox. In this, the software can modify registry keys also. At the termination of execution, the sandbox reverts to its original state, and the environment logs the behavior of the software.

In 2017, Chumachenko [13] proposed malware classification based on dynamic analysis using a scoring system in Cuckoo sandbox. They identified features such as registry keys, mutexes, processes, IP addresses, and API calls and formed the feature vector to perform machine learning algorithms. But, this method was time-consuming and had a limited dataset. In 2015, Berlin and Saxe [14] used a histogram of entropy values of opcode bytes as feature vector for a neural network-based classification. Vu Thanh Nguyen [15] used an artificial immune network for classification. But, both these methods leveraged a highly imbalanced dataset. Both comprised of over 80% malicious samples. Even with synthetic oversampling, the model will be highly biased to one class.

3 System Design

Gathering a balanced dataset is the crux of any machine learning-based application. Cleaning and pruning the data to be precise for the intended application can aid in creating a very efficient model. We have to create a balanced dataset containing the assembly language code (.asm files) corresponding to each sample belonging to the

benign and malignant classes. We use IDA Pro to disassemble the executable (.exe) files and convert them into .asm files. We have a command line feature available with IDA Pro to perform this activity conveniently in batches so that we can obtain a dataset corresponding to a large set of samples.

3.1 Dataset

For the purpose of this study, we collected ransomware samples from virusshare.com repository which had a collection of over thirty-five thousand samples along with their hash values. Around eight thousand samples were selected from this based on size of the files. This is a relatively good number of samples used for study when compared against other studies made in machine learning classification area which is going to be discussed in Section IV. These samples were from various cryptographic ransomware families, including TeslaCrypt, WannaCry, Petya, CryptoWall, and Cerber. Table 1 gives the complete information about these families and samples such as encryption technique, when the malware was released, and the sample count from each family that is presenting the dataset.

To make a balanced dataset for binary classification, almost eight thousand benign .exe files were taken from the local systems available at the various computer laboratories and staff departments at the University. We included several types of benign applications like basic file manipulation executables, DLLs, common tools, video players, browsers, drivers Both benign and malignant files were disassembled using IDA Pro into .asm files using command line interface.

Table 1 Ransomware families used

Family	Year	Techniques	Target	#Samples
CryptoLocker	2013	RSA	User files	741
CryptoWall	2014	RSA 2048 bit	User files	706
Cryrar	2012	RAR-sfx	User files	583
Locky	2016	RSA 2048 bit	User files	567
Petya	2016	AES-128MBR	User files	593
Reventon	2012	N/A	User files	617
TeslaCrypt	2015	ECC	Games and multimedia files	398
WannaCry	2017	RSA 2048 bit	User files	436
Cerber	2016	RSA 2048 bit	User files	263

3.2 Implementation

Figure 1 shows the complete system design. After creating a balanced dataset from benign and malignant files, we generate N -grams of opcode sequences from disassembled files. Then, N -gram sequences obtained are ranked, and top feature vectors are chosen. We can also set a threshold to limit the number of feature vectors to be chosen. In the next step, we compute the tf-idf for the chosen feature vectors. That is, each N -gram sequence is given a probability factor which represents how well they represent their classes.

3.3 Preprocessing

The collected .exe files need to be disassembled using any popular disassemblers. We used IDA Pro and a snippet of IDA pro as shown in Fig. 2. In order to batch process, we created a script file which runs sequentially and obtained disassembled .asm files. The method proposed here takes advantage of static analysis; that is, the samples are not needed to run on physical machines.

From the .asm files, we need to extract the opcode sequences in time-sorted order so that we can confirm which sequences will cause harm. We then record these time-sorted opcode sequences of all the benign and malignant files into a single text file along with the corresponding class label.

From each ransomware sample .asm file, various continuous sequences of opcodes of varying length (N grams of opcodes) are extracted from the original opcode sequence. In malware, any one opcode execution may not harm the device, but it may be detrimental to a machine if a sequence consisting of more than one opcode in a particular order is executed. In addition, a short, contiguous series of N -opcodes is called an N -gram. Hence, the N -gram can grab a few meaningful opcode sequences.

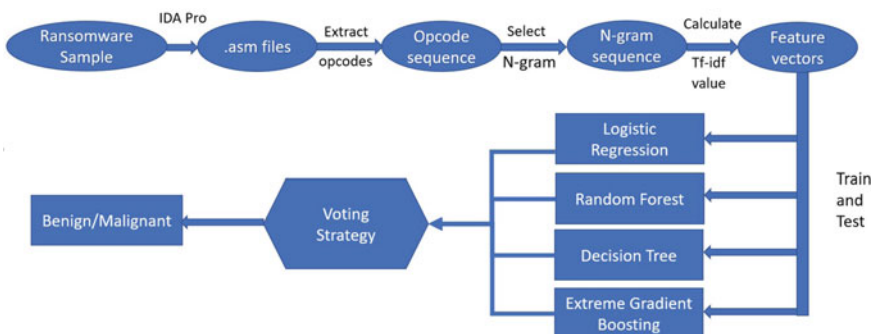


Fig. 1 Model diagram

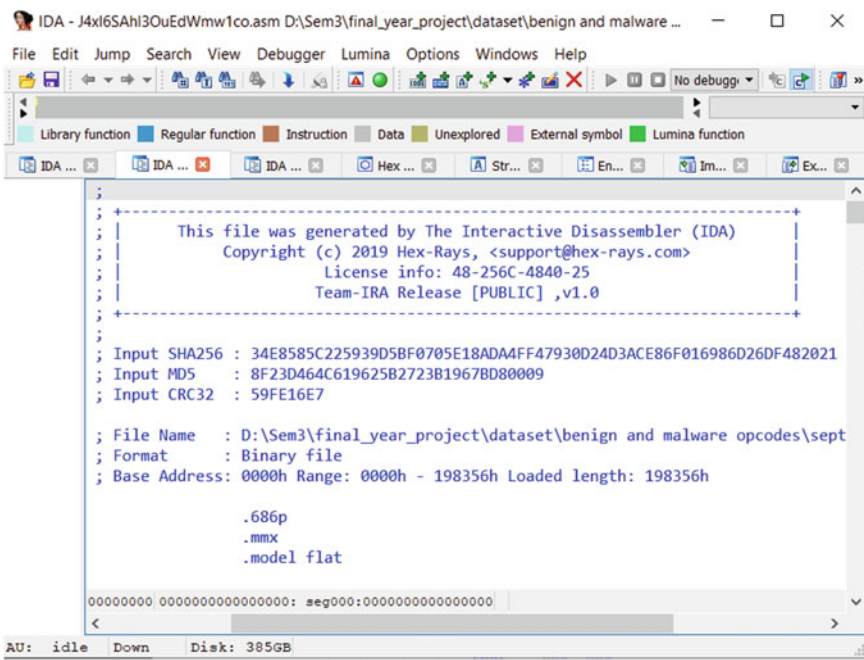


Fig. 2 Disassembled source code of a ransomware sample

The N -grams have the power to predict certain phenomena occurring if we obtain enough samples to train. It is also prudent to have an equal mix of benign and malignant samples to reduce bias toward any one class.

3.4 Computing Tf-idf Value

Out of all the ransomware families used in the study and the benign files, we obtained a significant number of N -gram sequences. Now, we need an efficient method to calculate the importance of each feature vector and rank them. Tf-idf uses a common approach that utilizes principle of language modeling so as to classify essential words. The heuristic intuition of tf-idf is that a word that is appearing in several corpuses is probably not a good indicator of a specific corpus and may not be allotted more points than the sequences that are found to turn up in fewer corpuses. By using Eqs. (1), (2), and (3), we determine the tf-idf for every N -gram.

$$TF_{t,d} = f(t, d)k_f(t, d) \tag{1}$$

Here, $TF_{(t,d)}$ represents the total number of times the N -gram t occurs in d th N -gram sequence. $f_{(t,d)}$ shows the frequency of N -gram t occurring in N -gram sequence d , and $\sum_k f_{(t,d)}$ denotes the total of N -grams in d . Then,

$$IDF(t) = \log 2|N| |\{d: t \in d | d \in N\}| \quad (2)$$

Here, $IDF_{(t)}$ specifies if N -gram t is that uncommon in all sequences of N -gram, whereas N is the collection of all sequences of N -grams. $|d: t \in d | d \in N|$ represents the count of sequences of N -grams that actually has N -gram t .

$$TF-IDF_{(t,Df)} = TF_{(t,Df)} \times IDF_{(t)} \quad (3)$$

In above Eq. (3), we use every sequence of N -gram within malignant files f so as to form a lengthy N -gram sequence Df . $TF - IDF_{(t,Df)}$ indicates the value of tf-idf in N -gram t of the lengthy N -gram sequence Df .

In a ransomware family, tf-idf measures the value of an N -gram, which increases as the count of a N -gram increases in a class. In addition, it is negatively proportional to how many times the exact N -gram is found to crop up in other ransomware families. To a certain degree possible, tf-idf will differentiate each class from other classes since the N -gram with a larger tf-idf score means that the number times it exists in one class is more and the frequency of occurrences across the latter classes is very low.

In each class, we arrange the N -grams as N -grams function in the descending order of their respective tf-idf score and pick only the most potent N -grams from both classes. To get the feature N -grams for one classifier, we combine $b * t$ feature vectors, i.e., t N -grams from each of the b classes. Since there are many similar N -grams features derived from different classes, the repeating N -grams features are eliminated.

By using Eq. 1, compute the term frequency (TF) score of each N -gram feature vector. In an N -gram, the values of the N -gram features are represented as one single vector, which is the corresponding ransomware feature vector for the next step. In the N -gram sequence, if the number of features present in the N -gram is 10, and the number of occurrences of a particular N -gram, say (del, mov, sub) is 2; then, using Eq. 1, we compute the TF score of (del, mov, sub). In order to form one vector, we calculate the TF score of each feature vector like this. We acquire all feature vectors by performing the same procedure for all N -grams in order to obtain the feature vector. The feature dimension is the count of all N -grams chosen. Table 2 shows the sample feature vector:

3.5 Training, Validation, and Testing

In the previous steps, for each class, we obtained the most indicative features. These have the class label information as well that needs to be fed into the ML algo-

Table 2 Sample 3-g sequence

Sample ID	1	2	3	4
Class label	1	1	0	1
add lock add	0.003764	0	0.00244	0.00276
add cmp mov	0.00084	0.01534	0.00140	0.00739
add mov mov	0.237791	0	0.03269	0.02501

rithm. During the training phase, we have trained the model using four machine learning models, namely SVM, random forest (RF), logistic regression (LR), and gradient boosting decision tree (GBDT) algorithms. We were able to conclude that the random forest algorithm gave much better classification accuracy when trained and tested with various feature lengths and N -gram sizes. We divided the dataset into 80 to 20 ratios for training and validation. We employed a randomized search on hyperparameters of all classification algorithms employed to find the optimum hyperparameter values and then use those for final model.

We obtained validation accuracy as high as 99% with the random forest classifier. Then, we use an ensemble learning model called voting classifier which predicts the class based on their highest probability of chosen class. It simply aggregates the findings of each classifier passed into voting classifier and predicts the output class based on the highest majority of voting. At last, the trained classifier model can, with a high degree of accuracy, predict the labels of unknown ransomware samples. By measuring the classification accuracy, false positive rate (FPR), and false negative rate (FNR), the performance of the model can be tested and compared. We tested the model using various other active ransomware samples provided by SERB and benign files, and it performed above par, yielding a testing accuracy of 94%.

4 Results and Discussions

Ransomware classification methods that rely upon dynamic analysis cannot fathom the samples that can detect the sandboxing environment in which they are analyzed. To cope with this limitation and to produce a model with a better binary classification accuracy, here, we made use of a static analysis-based approach based on opcode execution to classify ransoms. We employ ensemble learning using voting classifier on top of four major machine learning algorithms (RF, SVM, LR, and GBDT) to model classifiers. We also experimented with varying lengths of N -gram sequences (2, 3, and 4-g) and different threshold values to limit the number of features from each class. Tf-idf is utilized to pick the most relevant features, which led to a very good performance by the classification model. The model was used on diverse combinations of the dataset, and it proves that random forest (RF) has a better performance compared to other algorithms followed by support vector machine (SVM). We also

Fig. 3 Confusion matrix voting classifier

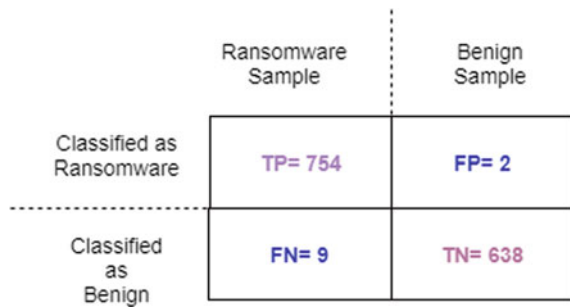
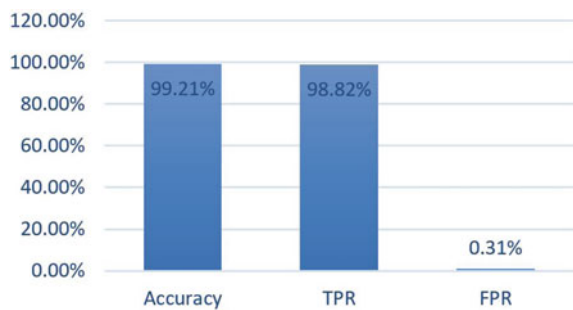


Fig. 4 Assessment of the experimental results



used randomized search cross-validation method to find the optimum hyperparameter values for each algorithm used.

The highest validation accuracy obtained using ensemble model voting classifier was noted to be 99.21% as shown in Fig.3. The experimental results clearly show that this model can detect ransomwares with a false positive rate of just 0.31% as shown in Fig.4.

We performed an extensive testing using different *N*-gram and feature lengths on all the chosen classification algorithms. Owing to the excellent feature selection technique, we got splendid results with an average of 98% classification accuracy in all algorithms combined. Results are shown in Table 3.

Table 3 Training accuracy

Classifiers	2-g	3-g	4-g
Random forest (RF)	99.07	98.64	98.98
Support vector machine (SVM)	97.22	96.86	97.50
Extreme gradient boosting (GBDT)	98.64	98.43	98.21
Voting classifier	98.76	99.21	98.86

Table 4 Testing accuracy

Classifiers	2-g	3-g	4-g
Random forest (RF)	77.28	90.37	84.93
Support vector machine (SVM)	69.13	86.17	77.28
Extreme gradient boosting (GBDT)	61.97	89.87	80.24
Voting classifier	93.33	91.11	85.43

Table 5 Time required in seconds

Voting classifier	2-g	3-g	4-g
Calculating tf-idf value	76.54	79.04	83.53
Model fitting	82.14	120.23	168.37
Model fitting	1.96	2.16	3.08

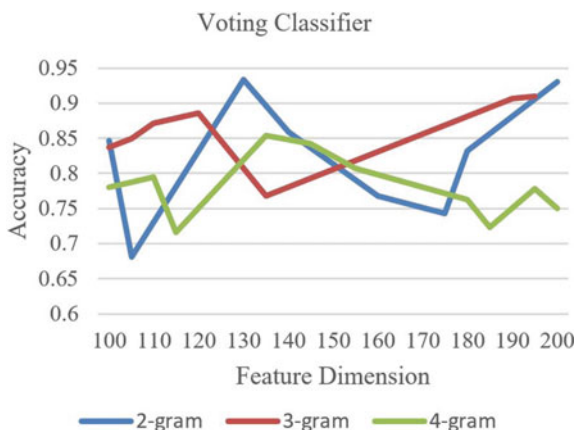
After training the model, we tested it using different sets containing a diverse combination of ransomware samples and benign files. We were able to obtain classification accuracy of 90% using the random forest (RF) algorithm. The ransomware samples used for testing were observed to contain different packers and had plenty of obfuscation methods employed in them which were ascertained by testing the samples using dynamic analysis methods. These results were further enhanced by employing the ensemble voting method after optimizing the hyperparameter values. We obtained model accuracy as high as 93.33%. Results are shown in Table 4.

Training this machine learning model using any one classifier algorithm takes almost negligible time. The time taken for each step, in seconds, is presented in Table 5. Since the disassembling step is common for all variations, it need not be considered.

Loading the dataset created using disassembled opcodes into the Python notebook takes an average 12s. Vectorization and selection of feature vectors were observed to be the most time-consuming step during the implementation phase. It was noted to have consumed over 3 min for each variation. Time required for model fitting or training using a voting classifier increases drastically with the increase in the number of N -grams. Once we fit the model, it takes negligible time for making the prediction. We trained and tested the model with various feature dimensions for all three chosen N -gram sequences to select the optimum number of features. Then, further tests were conducted based on the chosen feature dimension and N -gram size. The results are shown in Fig. 5.

We can see from the results in above figures that in voting classifier, both 2 and 3-g performances are better in classifying samples.

Fig. 5 Classification accuracy of voting classifier in varying feature dimensions



We have also compared our work with existing dynamic analysis-based methods as listed in Table 6. The results tabulated are from the respective published papers. The test samples that we have used in this study are active and live.

From the table, it is clear that our method proves to be more efficient than existing dynamic analysis models. This exempts us from designing complex sandboxes to analyze ransomware behavior. This method proved to be less time-consuming at the prediction phase. It will save countless working hours for any analyst who tries to dissect and figure out the heuristic signatures of a ransomware sample and then classify it manually. We were able to drastically reduce the false positive rate (FPR), which means very few benign files were wrongly classified. Though the false negativity rate (FNR) was negligible, while analyzing these samples that were wrongly classified as benign using various malware analysis tools, it was inferred that the samples employed code injection techniques. Those misclassified instances also exhibited advanced encryption techniques because of which the disassembler could not unpack the file properly.

5 Conclusion

Ransomware that can detect the virtualized sandboxing environment used to perform dynamic analysis will dodge detection by employing various evasive tactics. We proposed a static analysis method which uses natural language processing techniques to classify ransomware in order to address this disadvantage. In order to construct classifiers, we use an ensemble learning model called voting classifier consisting of four machine learning models (SVM, RF, LR, and GBDT). We use different lengths of N -gram opcode sequences (2, 3, and 4-g) with various threshold values to limit the number of features that represent each class. Tf-idf is used to determine the most potent N -gram features that are highly indicative with respect to each class, which

Table 6 Comparison with existing methods

Ransomware detection model	Method	Dataset used	TPR (%)	FPR (%)	FNR (%)
Automated dynamic analysis of ransomware: benefits, limitations, and use for detection [16]	Algorithm: regularized logistic regression feature selection: MIC	582-ransomware 942-benign	96.34	0.16	3.66
Deep learning for ransomware detection [17]	Algorithm: deep natural network. Detection before encryption begins	155 ransomware Unknown benign	93.92	38	7.08
Leveraging machine learning techniques for windows ransomware network traffic detection [18]	Algorithm: J48 decision tree. Feature selection: Tshark extractor	210-ransomware 264-benign	97.1	1.6	2.9
Our method	Algorithm: ensemble model. Feature selection: <i>N</i> -gram, tf-idf	2983-ransomware 2682-benign	98.82	0.31	1.18

results in yielding a high validation accuracy. Detailed tests on real datasets show that the other algorithms also perform as well as random forest, which prompted us to employ the voting classifier strategy. The findings conclude that classifiers using *N*-gram feature vectors of different lengths are a better model to effectively classify ransomware. The results then compared with existing ransomware detection methods that rely on dynamic analysis model prove that our static analysis model performs better.

As the future enhancements to this model, we can introduce into our dataset, malignant samples which have advanced code obfuscation capabilities such as code rearranging, injecting garbage/benign opcodes into the source code in order to avoid detection. We can also move to deep learning methods, probably, RNN for even better feature selection and improved classification efficiency.

Acknowledgements This work was carried out as part of a research project sponsored by the Science and Engineering Research Board (SERB), the Department of Science and Technology (DST), Government of India (GoI). We express our sincere gratitude to DST-SERB for the support they extended to us through the “Early Career Research” Award (Sanction No. ECR/2018/001709). We are much obliged to the Department of CSE for having facilitated the seamless research environment even during the adverse circumstances caused by COVID-19 pandemic. We sincerely thank all the faculties at the Department of CSE and CTS Laboratory for their meticulous support.

References

1. J. Fu, J. Xue, Y. Wang, Z. Liu, C. Shan, Malware visualization for fine-grained classification. *IEEE Access* **6**, 14510–14523 (2018)
2. A. Ali, Ransomware: a research and a personal case study of dealing with this nasty malware. *Issues Inform. Sci. Inf. Technol.* **14**, 087–099 (2017)
3. B. Eduardo, D. Morat Oss, E. Magana Lizarrondo, M. Izal Azcarate, A survey on detection techniques for cryptographic ransomware. *IEEE Access* **7**, 144925–144944 (2019)
4. B.A. Al-rimy, M.A. Maarof, S.Z. Shaid, Ransomware threat success factors, taxonomy, and countermeasures: a survey and research directions. *Comput. Secur.* **74**, 144–166 (2018)
5. K. DaeYoub, J. Lee, Blacklist versus whitelist-based ransomware solutions. *IEEE Consumer Electron. Mag.* **9**(3), 22–28 (2020)
6. A. Pekta, T. Acarman, Classification of malware families based on runtime behaviors. *J. Inf. Secur. Appl.* **37**, 91–100 (2017)
7. J.O. Kephart, W.C. Arnold, Automatic extraction of computer virus signatures, in *Proceedings of the 4th Virus Bulletin International Conference* (Abingdon, UK, 1994)
8. A. Kharraz, S. Arshad, C. Mulliner, W. Robertson, E. Kirda, UNVEIL: a largescale, automated approach to detecting ransomware, in *Proceedings of the 25th USENIX Conference on Security Symposium* (USENIX Security, 2016), pp. 757–772
9. S.S. Hansen, T.M.T. Larsen, M. Stevanovic, J.M. Pedersen, An approach for detection and family classification of malware based on behavioral analysis, in *Proceedings of 2016 International Conference on Computing, Networking and Communications. ICNC* (IEEE, 2016), pp. 1–5
10. C. Kolbitsch, E. Kirda, C. Kruegel, The power of procrastination: detection and mitigation of execution-stalling malicious code, in *Proceedings of the 18th ACM Conference on Computer and Communications Security* (ACM, 2011), pp. 285–296
11. G. Ramesh, A. Menen, Automated dynamic approach for detecting ransomware using finite-state machine. *Dec. Support Syst.* **138**, 113400 (2020)
12. R. Vinayakumar, K.P. Soman, K.K.S. Velany, S. Ganorkar, Evaluating shallow and deep networks for ransomware detection and classification, in *Proceedings of 2017 International Conference on Advances in Computing, Communications and Informatics, ICACCI* (IEEE, 2017), pp. 259–265
13. H. Zhang, X. Xiao, F. Mercaldo, S. Ni, F. Martinelli, A.K. Sangaiah, Classification of ransomware families with machine learning based on N -gram of opcodes. *Future Gener. Comput. Syst.* **90**, 211–221

14. I. Kwon, E.G. Im, Extracting the representative API call patterns of malware families using recurrent neural network, in *Proceedings of the International Conference on Research in Adaptive and Convergent Systems* (ACM, 2017), pp. 202–207
15. A. Mohaisen, A.G. West, A. Mankin, O. Alrawi, Chatter: classifying malware families using system event ordering, in *Proceedings of 2014 IEEE Conference on Communications and Network Security*. CNS (IEEE, 2014), pp. 283–291
16. D. Sgandurra, L. Muoz-Gonzlez, R. Mohsen, E.C. Lupu, Automated dynamic analysis of ransomware: benefits, limitations and use for detection (2016). [arXiv:1609.03020](https://arxiv.org/abs/1609.03020)
17. A. Tseng, Y. Chen, Y. Kao, T. Lin, Deep learning for ransomware detection. Internet Archit. IA2016 Workshop Internet Archit. Appl. IEICE Techn. Rep. **116**(282), 87–92 (2016)
18. O.M. Alhawi, J. Baldwin, A. Dehghantanha, *Leveraging machine learning techniques for windows ransomware network traffic detection*, in *Cyber Threat Intelligence* (Springer, Cham, 2018), pp. 93–106
19. D. Bilar, Opcodes as predictor for malware. Int. J. Electron. Secur. Digital Forensics **1**(2), 156–168 (2007)
20. R. Moskovitch, C. Feher, Y. Elovici, Unknown malcode detection: a chronological evaluation, in *IEEE International Conference on Intelligence and Security Informatics, 2008. ISI* (IEEE, 2008), pp. 267–268
21. R. Moskovitch, et al., Unknown malcode detection via text categorization and the imbalance problem, in *2008 IEEE International Conference on Intelligence and Security Informatics* (IEEE, 2008), pp. 156–161
22. R. Moskovitch et al., *Unknown malcode detection using opcode representation*, in *Intelligence and Security Informatics* (Springer, Berlin Heidelberg, 2008), pp. 204–215
23. I. Firdausi, et al., Analysis of machine learning techniques used in behavior-based malware detection, in *2010 Second International Conference on Advances in Computing, Control and Telecommunication Technologies (ACT)* (IEEE, 2010)
24. L. Yi-Bin, D. Shu-Chang, Z. Chao-Fu, B. Gao, Using multi-feature and classifier ensembles to improve malware detection. J. CCIT **39**(2), 57–72 (2010)
25. I. Santos et al., *Opcode-sequence-based semisupervised unknown malware detection*, in *Computational Intelligence in Security for Information Systems* (Springer, Berlin, Heidelberg, 2011), pp. 50–57
26. Z. Zhao, A virus detection scheme based on features of control flow graph, in *2011 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC)* (IEEE, 2011 Aug 8), pp. 943–947
27. Y. LeCun, Y. Bengio, G. Hinton, Deep learning. Nature **521**, 436 (2015). <http://dx.doi.org/10.1038/nature14539>
28. Y. Lecun, Generalization and network design strategies, in *Connectionism in Perspective* (Elsevier, 1989)
29. S. Hochreiter, J. Schmidhuber, Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
30. K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using RNN encoderdecoder for statistical machine translation, in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (Association for Computational Linguistics, Doha, Qatar, 2014), pp. 1724–1734. <http://dx.doi.org/10.3115/v1/D14-1179>
31. M. Schuster, K.K. Paliwal, Bidirectional recurrent neural networks. IEEE Trans. Signal Process. **45**(11), 2673–2681 (1997)
32. N. Harini, T.R. Padmanabhan, 2CAuth: a new two factor authentication scheme using QR-code. Int. J. Eng. Technol. **5**(2), 1087–1094 (2013)
33. G. Ramesh, I. Krishnamurthi, K. Sampath Sree Kumar, An efficacious method for detecting phishing webpages through target domain identification. Dec. Support Syst. **61**, 12–22 (2014)
34. N. Harini, T.R. Padmanabhan, 3c-auth: a new scheme for enhancing security. Int. J. Netw. Secur **18**(1), 143–150 (2016)

Flood Prediction Using Hybrid ANFIS-ACO Model: A Case Study



Ankita Agnihotri, Abinash Sahoo, and Manoj Kumar Diwakar

Abstract Growing imperviousness and urbanization have increased peak flow magnitude which results in flood events specifically during extreme conditions. Precise and reliable multi-step ahead flood forecasts are beneficial and crucial for decision makers. Present study proposes adaptive neuro-fuzzy inference system (ANFIS) combined with ant colony optimization (ACO) algorithm which optimize model parameters for predicting flood at Matijuri gauge station of Barak River basin, Assam, India. Potential of hybrid flood forecasting model is compared with standalone ANFIS based on quantitative statistical indices such as coefficient of determination (R^2), Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). Analysis of results generated by models indicated that ANFIS-ACO model with $RMSE = 0.0231$, $R^2 = 0.96014$ and $MAE = 0.0185$ performed better with more accuracy and reliability compared to standalone ANFIS model. Also, results demonstrated ability of proposed optimization algorithm in improving accurateness of conventional ANFIS for flood prediction in selected study site.

Keywords Flood · ANFIS-ACO · Barak River · Prediction

1 Introduction

Change in climatic conditions are a major reason for increase in extreme hydrological events such as heavy rainfall and flooding. Because of temporal and spatial variations in rainfall distribution with highly nonlinear and inordinately complex nature of relationship between rainfall and runoff, forecasting flood events remain one of the most important and challenging tasks for effective hydrological study

A. Agnihotri · M. K. Diwakar
Department of Civil Engineering, MNIT Jaipur, Jaipur, Rajasthan, India
e-mail: 2020rce9520@mnit.ac.in

M. K. Diwakar
e-mail: manoj.ce@mnit.ac.in

A. Sahoo (✉)
Department of Civil Engineering, NIT Silchar, Silchar, Assam, India

[3]. Precise forecasts of flood time series is crucial for development of flood damage mitigation, flood warning system, soil conservation, soil erosion reduction, and flood prevention. Forecasting flood events represent a multifaceted nonlinear problem and therefore is tough for modelling. In last two decades, several soft computing techniques like artificial neural network (ANN) and fuzzy inference system (FIS) have been successfully implemented in modelling hydrological problems. Among these techniques regression analysis and wireless sensor networks have been used for various purposes [2, 5]. Such models are capable of ‘learning’ nonlinear relationship of a specified process. FIS are qualitative modelling systems motivated from human perceptive capability where system behaviour is defined utilising a natural language. Lately, a rising number of FIS applications in hydrology have been reported [7, 13, 15, 19, 24, 25, 27].

Mukerji et al. [10] utilised ANN, ANFIS, and adaptive neuro Genetic Algorithm Integrated System (ANGIS) for carrying flood forecasts at Jamtara gauge station of Ajay River, India. Bisht and Jangid [1] used ANFIS to develop river stage–discharge models at the Dhawalashwaram barrage spot in Andhra Pradesh, India. Based on comparison of observed and estimated data, outcomes revealed that ANFIS performed better in predicting river flow discharge compared to customary models. Rezaeianzadeh et al. [14] used ANFIS, ANN, multiple linear regression, and multiple nonlinear regression to estimate peak flow of Khosrow Shirin catchment, positioned in Fars region, Iran on a daily basis. Predictive capabilities of the proposed model were evaluated and observed that ANFIS performed superior for predicting daily flow discharge at the proposed site with spatially distributed rainfall as input. Pahlavani et al. [12] presented the applicability of ANFIS to model flood hydrograph at the Shirindarreh basin positioned in the northern Khorasan Province, Iran. However, ANFIS has significant limitations like training complexity and curse of dimensionality that limits its application on problems having huge datasets.

Also, ANFIS is related with a major shortcoming that is implementation of tuning parameters for finding optimal membership functions. Thus, combination of nature-inspired optimization algorithms with standalone ANFIS acts as a new alternative modelling approach to improve its performance in resolving challenging problems [6, 11]. Several hybrid artificial intelligence models have been employed to forecast floods in various rivers across India [17, 18]. Development of flood forecasting models using hybrid wavelet-based ANFIS models were investigated [20, 21]. Obtained results indicated that W-ANFIS provided more efficient flood forecasts than simple ANFIS. Similarly, there are many applications of hybrid ANFIS-ACO model in several fields of science and engineering, they are, mammogram classification [26], classification of gene expressions of colon tumour and lung cancer [28]. Furthermore they exhibited a noticeable application in varied civil engineering problems [8, 22, 23] (Azad et al. 2018). In present study, a hybrid ANFIS-ACO technique is established to predict flood events, making it a new application in flood forecasting study. Combination of ANFIS and ACO has enhanced the performance simple ANFIS model in flood prediction.

2 Study Area and Data

The River Barak flows through Nagaland, Manipur, Assam and Mizoram states in India and finally converges in Bay of Bengal via Bangladesh. It falls between geographical location of $24^{\circ}8'$ to $25^{\circ}8'$ N latitudes and $92^{\circ}15'$ to $93^{\circ}15'$ E longitudes with a drainage area of nearly $41,157 \text{ km}^2$ in the Indian subcontinent. This river basin lying in tropical region of India is characterised by many reckless meandering twists and shows changes and shift in channel at several places through time and space. Climate in Barak valley is to some extent humid and is divided into four season, i.e., pre-monsoon, monsoon, post-monsoon and winter. Monsoon starts from June and lasts till August when the basin receives maximum precipitation. Present study focuses on developing a flood prediction model for Matijuri gauging station of Barak Valley as shown in Fig. 1.

3 Methodology

3.1 ANFIS

ANFIS is a machine learning modelling approach combining human knowledge, adapting ability of ANN with reasoning capability of FIS. FIS denotes to theory which employs fuzzy sets having classes of incomplete, unclear and imprecise information. Here, membership functions (MFs) are stipulated to execute numerical calculations by utilising linguistic labels [9, 16]. A typical ANFIS architecture consisting of five layers is represented in Fig. 2. Each input's linguistic descriptions and number of descriptions which depends on nature of applied fuzzy MFs is found in the first layer. Using additional number of fuzzified values intricacy of network structure is intensified. Second layer finds product of all associated membership values. Normalisation of incoming values occurs in third layer. Fourth layer is utilised for defuzzification, whereas in final layer that is the fifth layer, output is computed. For a first-order Sugeno-type fuzzy model, a common set of if-then rules are defined below:

$$\text{IF } (x \text{ is } A_1) \text{ and } (y \text{ is } B_1) \text{ THEN } (f_1 = p_1x + q_1y + r_1)$$

$$\text{IF } (x \text{ is } A_2) \text{ and } (y \text{ is } B_2) \text{ THEN } (f_2 = p_2x + q_2y + r_2)$$

where A_1 , A_2 and B_1 , B_2 are membership values of input variables x and y , in respective order; p_1 , q_1 , r_1 and p_2 , q_2 , r_2 are constraints of output function f_1 and f_2 .

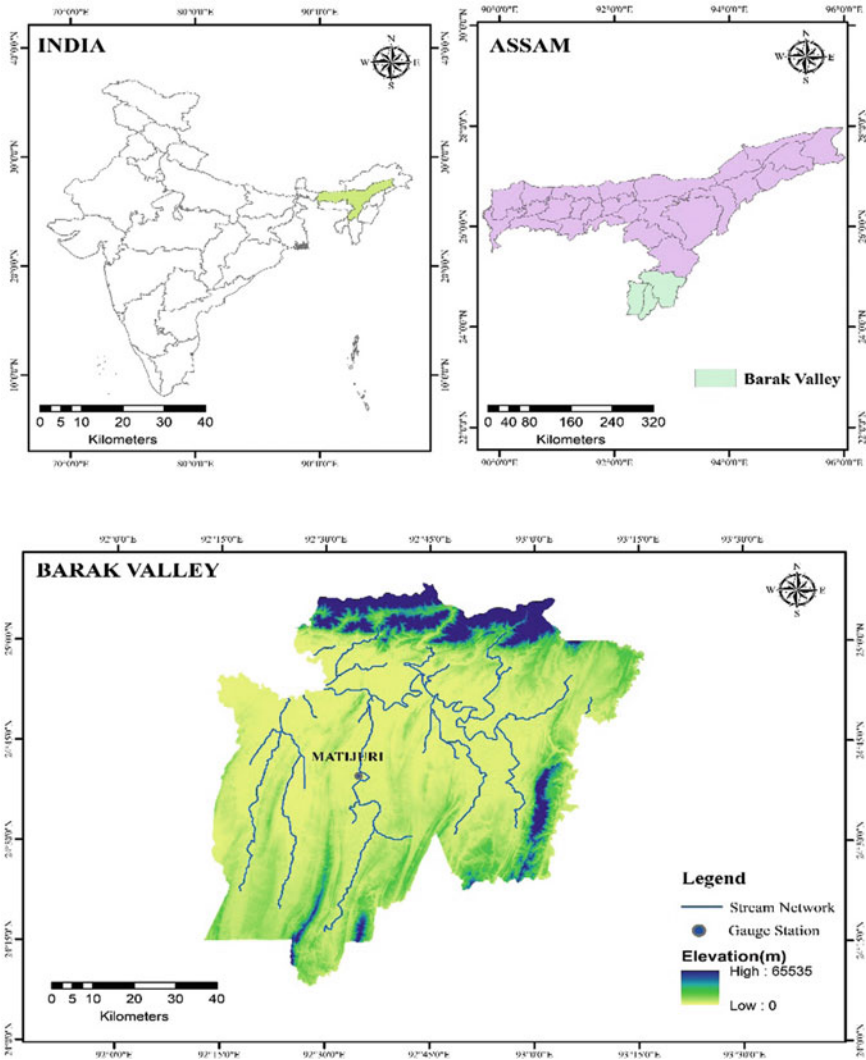


Fig. 1 Study area

3.2 ACO

Dorigo and Di Caro [4] first introduced ACO as a multi-feature solution to solve various optimization problems. Fundamental knowledge about ACO algorithm is that it works on basis of behaviour of natural ant colonies. It is a parallel search using a number of computational threads based on dynamic memory structure and problem data. The dynamic memory comprises information on feature of different solutions obtained preciously to considered optimization problem. Communication of various

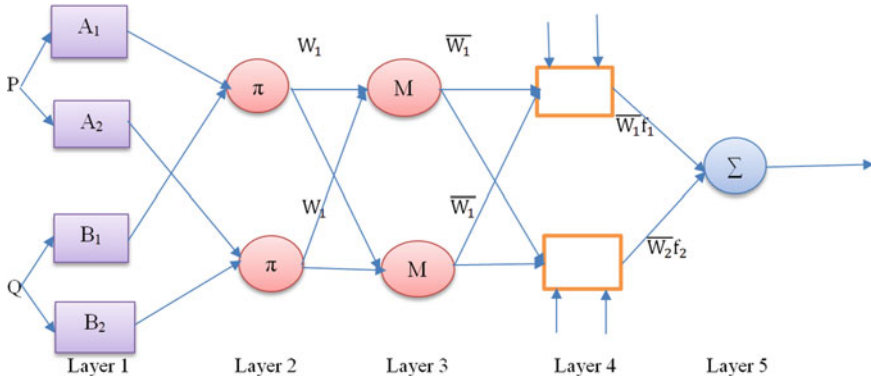


Fig. 2 Architecture of ANFIS

search agents is a combined behaviour that has demonstrated to be highly efficient to solve combinatorial optimization problems. For a specified problem, ants will work in parallel for trying out different solutions until they discover a superior one. Ants communicate with one another utilising pheromone trails which assist them in search for shortest way to the food source. An identical process is utilised in ACO algorithm to find optimum point in search space. Ants move in paths in forward and backward manner. They apply a repetitive process for exploring perfect solution related with the problem. The selection of transition from node ‘i’ to a node ‘j’ is arbitrarily based on a probability as expressed in Eq. (1).

$$P_{ij}^k(t) = \frac{([\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta)}{\sum_{i \in N_i} [\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta} \tag{1}$$

- τ_{ij} : Quantity of pheromones on edge (i, j).
- η_{ij} : Value of visibility amid edge (i, j).
- α : Constant feature controlling impact of τ_{ij} .
- β : Constant feature controlling impact of η_{ij} .
- N_i : Set of node points which haven’t been visited yet.

Using this probability ants will travel from any ‘i’ node to another ‘j’ node. After all nodes are visited, all ants are observed to have completed their iterations or tours. The complete working procedure of ANFIS-ACO model is represented in Fig. 3.

3.3 Evaluating Constraint

Rainfall (P_t) and flood (f_t) data for a period of 1960–2019 from June to October (monsoon season) are collected from CWC, Shillong. 70% of collected data, i.e., 1960–2001 are utilised to train, whereas remaining 30% of data set, i.e., 2002–2019

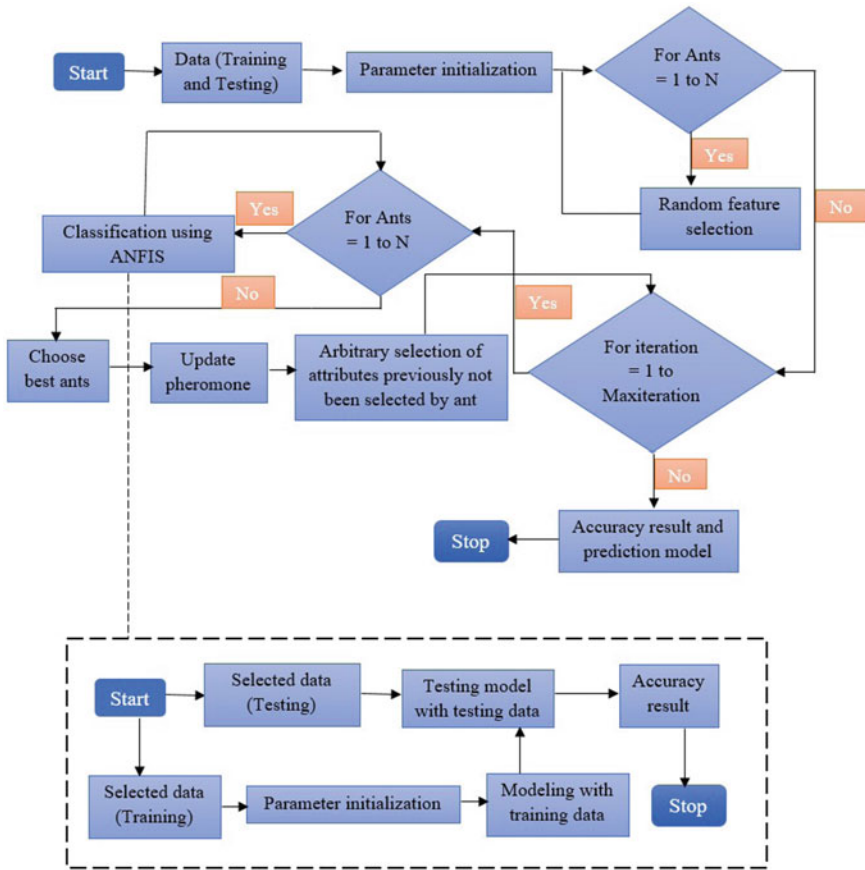


Fig. 3 Flow diagram of ANFIS-ACO

are utilised to test the applied model. Quantitative statistical measures such as R^2 , MAE and RMSE are used for assessing performance of models and finding the best among them.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (R_i - \hat{R}_i)^2} \tag{2}$$

$$R^2 = \left(\frac{\sum_{i=1}^n (R_i - \bar{R}_i)(R^* - \tilde{R})}{\sqrt{\sum_{i=1}^n (R_i - \bar{R}_i)^2 \sum_{i=1}^n (R^* - \tilde{R})^2}} \right)^2 \tag{3}$$

$$MAE = \frac{\sum_{i=1}^n R_i - \widehat{R}_i}{n} \tag{4}$$

where R_i and \widehat{R}_i are observed and forecasted values, R^* and \widetilde{R} are mean of observed and forecasted values, respectively, and n is number of observed values. For finding best performing model, MAE and RMSE must be minimum while R^2 must be maximum.

4 Results and Discussions

Performance of proposed models is assessed for flood forecasting in Barak River considering Matijuri gauging station for monsoon season during which flood possess huge challenge to affected public. In this study, collected data were normalised within a range (0–1) using following expression.

$$R_{norm} = \frac{R_i - R_{min}}{R_{max} - R_{min}} \tag{5}$$

where R_{norm} , R_i ; R_{min} and R_{max} signify normalised, observed, minimum and maximum data values, in respective order.

In Table 1, P_{t-1} represents one month lag rainfall; P_{t-2} —two month lag rainfall; P_{t-3} —three month lag rainfall and P_{t-4} : four month lag rainfall. The performance of hybrid ANFIS-ACO and conventional ANFIS model is summarised in Table 1 for training and testing phases. It can be clearly observed from Table 1 that simulation

Table 1 Performance of various models

Technique	Model	Input scenario	Training period			Testing period		
			R^2	RMSE	MAE	R^2	RMSE	MAE
ANFIS	Model-I	P_{t-1}	0.92383	0.0735	0.0502	0.91381	0.0674	0.0647
	Model-II	P_{t-1}, P_{t-2}	0.93742	0.0642	0.0475	0.92042	0.0615	0.0513
	Model-III	$P_{t-1}, P_{t-2}, P_{t-3}$	0.94167	0.0571	0.0432	0.92635	0.0541	0.0451
	Model-IV	$P_{t-1}, P_{t-2}, P_{t-3}, P_{t-4}$	0.94791	0.0453	0.0386	0.93908	0.0436	0.0399
ANFIS-ACO	Model-I	P_{t-1}	0.96095	0.0409	0.0314	0.94357	0.0382	0.0342
	Model-II	P_{t-1}, P_{t-2}	0.96518	0.0374	0.0253	0.95173	0.0327	0.0274
	Model-III	$P_{t-1}, P_{t-2}, P_{t-3}$	0.97352	0.0331	0.0208	0.95626	0.0283	0.0226
	Model-IV	$P_{t-1}, P_{t-2}, P_{t-3}, P_{t-4}$	0.97824	0.0267	0.0179	0.96014	0.0231	0.0185

(training phase) and forecasting (testing phase) accuracy of ANFIS-ACO model fluctuates w.r.t. input combinations with lowest training and testing RMSE (0.0267 and 0.0231), MAE (0.0179 and 0.0185) and maximum training and testing R^2 (0.97824 and 0.96014) belonging to Model-IV with input of $P_{t-1}, P_{t-2}, P_{t-3}, P_{t-4}$. Similarly for standalone ANFIS, lowest training and testing RMSE (0.0453 and 0.0436), MAE (0.0386 and 0.0399) and maximum training and testing R^2 (0.94791 and 0.93908) belonging to Model-IV with input of $P_{t-1}, P_{t-2}, P_{t-3}, P_{t-4}$.

Figure 4 shows the scatter plot representation comparing actual data with predictions made by proposed models. It is evident from Fig. 4 that ANFIS-ACO has less scattered predictions in comparison with ANFIS model. Time variation plot of observed and forecasted flood by hybrid and conventional models is illustrated in Fig. 5. Results show that, estimated peak floods are 7183.97, 7340.792 m^3/s , for ANFIS and ANFIS-ACO models against actual peak 7649.785 m^3/s for Matijuri gauge station. It can be clearly observed from detailed graphs that forecasts made by ANFIS-ACO model are nearer to corresponding observed flood values. Figure 6 shows the box plot of observed and predicted values by ANFIS and ANFIS-ACO models for testing phase. Figure shows highest similarities of ANFIS-ACO with observed values.

Based on analysis of results from current research, it is relatively clear that ANFIS-ACO performed superior compared to conventional ANFIS. Present work is restricted to a single region, and additional detailed studies are necessary for investigating obtained results. For future investigations, researchers will assess uncertainty of input data using other meta-heuristic algorithms over different time scales and combination of other input parameters. It is anticipated that present and upcoming investigations in this way will deliver a basis for development of efficient real-time flood forecasting models with different data-driven techniques integrated with optimization algorithms. The significance of present study lies in the specified study area. As every year, Assam receives major floods due to glaciers melt in summer which coincides with monsoon rainfall resulting in intensification of downstream, causing

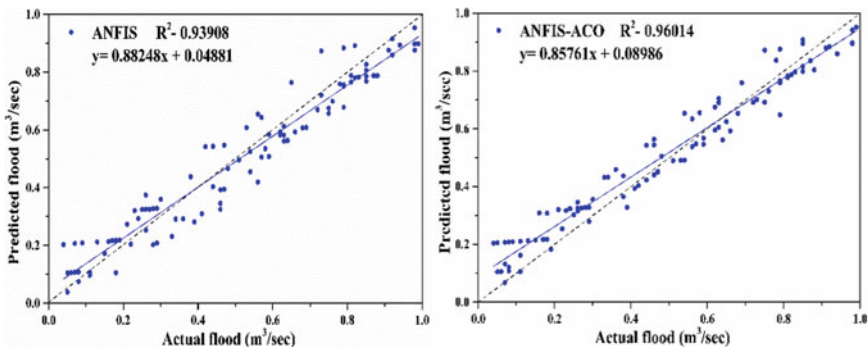
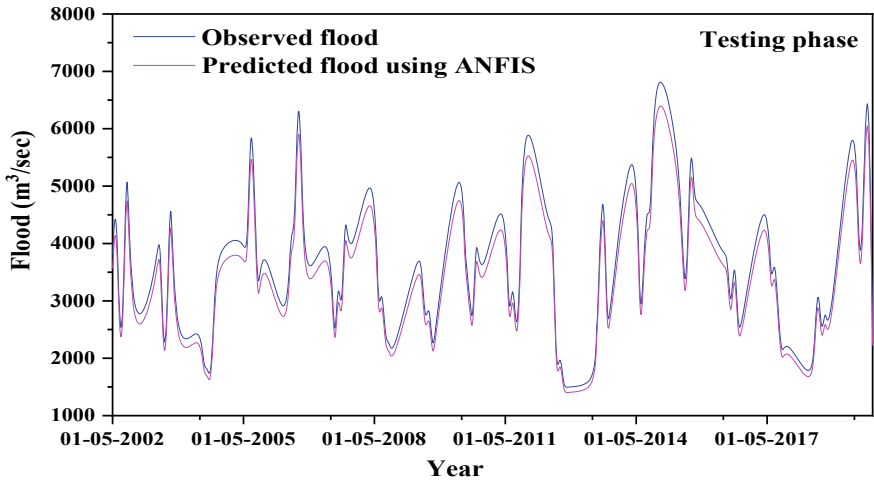
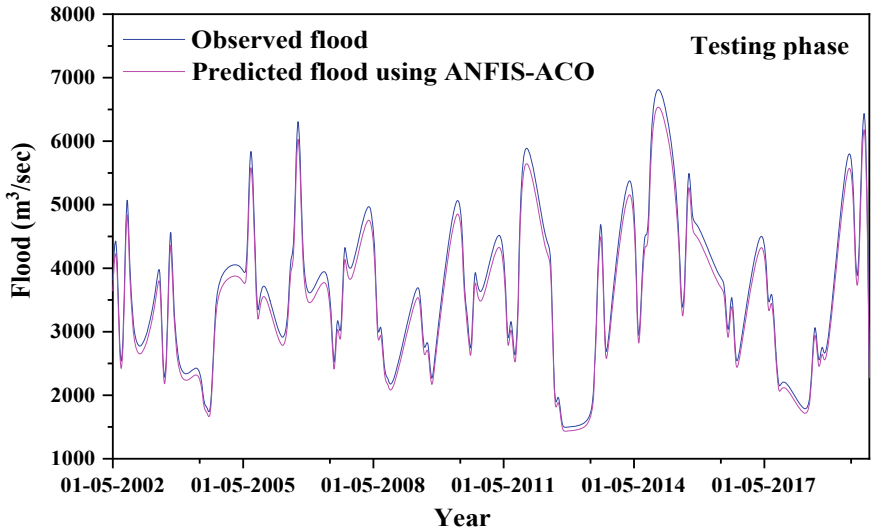


Fig. 4 Scatter plots of actual verses predicted flood



(a)



(b)

Fig. 5 Comparison between simulated and observed flood using a ANFIS and b ANFIS-ACO

the annual floods. Moreover, the use of ANFIS-ACO model is a new application in the field of flood forecasting which makes this study more substantial.

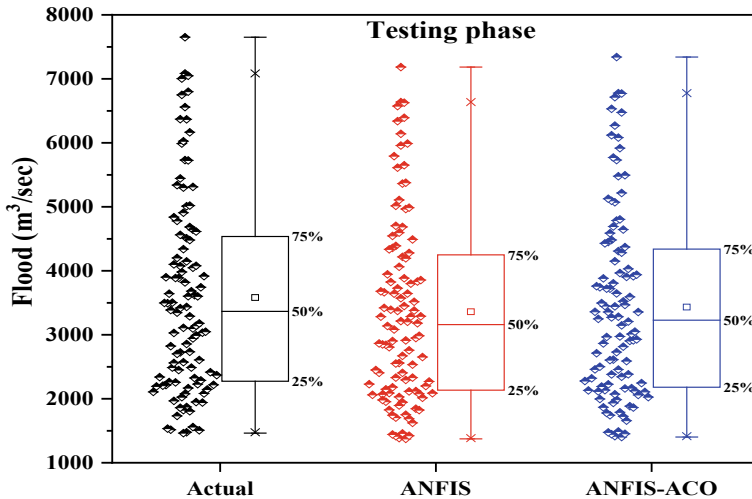


Fig. 6 Box plot of observed and predicted flood at Matijuri

5 Conclusion

Because of complex and uncertain nature of flood, application of data-driven models for flood forecasting is a challenging task for hydrologists, and thus, requires new optimization methods combined with conventional models for boosting accuracy by optimising model's parameters. This study used hybrid ANFIS-ACO model to predict flood at Matijuri gauge station of Barak River basin, Assam and evaluated its performance with standalone ANFIS model by using standard statistical measures. Obtained results indicated that ANFIS-ACO model with $RMSE = 0.0231$, $R^2 = 0.96014$ and $MAE = 0.0185$ performed better than simple ANFIS model for same input combinations. Further study could be carried out for establishing appropriateness of proposed models and performance criterion in different regions.

References

1. D.C.S. Bisht, A. Jangid, Discharge modelling using adaptive neuro-fuzzy inference system. *Int. J. Adv. Sci. Technol.* **31**, 99–114 (2011)
2. N. Chakrabarty, A regression approach to distribution and trend analysis of quarterly foreign tourist arrivals in India. *J. Soft Comput. Paradigm (JSCP)* **2**(01), 57–82 (2020)
3. F.J. Chang, Y.M. Chiang, L.C. Chang, Multi-step-ahead neural networks for flood forecasting. *Hydrol. Sci.* **52**(1), 114–130 (2007)
4. M. Dorigo, G. Di Caro, Ant colony optimization: a new metaheuristic, in *Proceeding of the 1999 Congress on Evolutionary Computation*, vol. 2 (1999), pp. 1470–1477
5. K. Kamel, S. Smys, Sustainable low power sensor networks for disaster management. *IRO J. Sustain. Wirel. Syst.* **04**, 247–255 (2019)

6. D. Karaboga, E. Kaya, An adaptive and hybrid artificial bee colony algorithm (aABC) for ANFIS training. *Appl. Soft Comput. J.* **49**, 423–436 (2016)
7. A.K. Lohani, N.K. Goel, K.K.S. Bhatia, Improving real time flood forecasting using fuzzy inference system. *J. Hydrol.* **509**, 25–41 (2014)
8. M. Mohammed, A. Sharafati, N. Al-Ansari, Z.M. Yaseen, Shallow foundation settlement quantification: application of hybridized adaptive neuro-fuzzy inference system model. *Adv. Civ. Eng.* **7381617** (2020)
9. N.R. Mohanta, N. Patel, K. Beck, S. Samantaray, A. Sahoo, Efficiency of river flow prediction in river using wavelet-CANFIS: a case study, in *Intelligent Data Engineering and Analytics* (Springer, Singapore, 2021), pp. 435–443
10. A. Mukerji, C. Chatterjee, N.S. Raghuvanshi, Flood forecasting using ANN, neuro-fuzzy, and neuro-GA models. *J. Hydrol. Eng.* **14**, 647–652 (2009)
11. D.T. Nguyen, S. Yin, Q. Tang, P.X. Son, L.A. Duc, Online monitoring of surface roughness and grinding wheel wear when grinding Ti-6Al-4V titanium alloy using ANFIS-GPR hybrid algorithm and Taguchi analysis. *Precis. Eng.* **55**, 275–292 (2019)
12. H. Pahlavani, A.A. Dehghani, A.R. Bahremand, S. Shojaei, Intelligent estimation of flood hydrographs using an adaptive neuro-fuzzy inference system (ANFIS). *Model. Earth Syst. Environ.* **3**, 35 (2017)
13. T. Rajaei, S.A. Mirbagheri, M. Zounemat-Kermani, V. Nourani, Daily suspended sediment concentration simulation using ANN and neuro-fuzzy models. *Sci. Total Environ.* **407**(17), 4916–4927 (2009)
14. M. Rezaeianzadeh, H. Tabari, A.A. Yazdi, S. Isik, L. Kalin, Flood flow forecasting using ANN, ANFIS and regression models. *Neural Comput. Appl.* **25**, 25–37 (2014)
15. H.R. Safavi, M.A. Aljaniyan, Optimal crop planning and conjunctive use of surface water and groundwater resources using fuzzy dynamic programming. *J. Irrig. Drain. Eng.* **137**(6), 383–397 (2011)
16. A. Sahoo, S. Samantaray, S. Bankuru, D.K. Ghose, Prediction of flood using adaptive neuro-fuzzy inference systems: a case study, in *Smart Intelligent Computing and Applications* (Springer, Singapore, 2020), pp. 733–739
17. A. Sahoo, U.K. Singh, M.H. Kumar, S. Samantaray, Estimation of flood in a river basin through neural networks: a case study, in *Communication Software and Networks* (Springer, Singapore, 2021a), pp. 755–763
18. A. Sahoo, S. Samantaray, D.K. Ghose, Prediction of flood in Barak River using hybrid machine learning approaches: a case study. *J. Geol. Soc. India* **97**(2), 186–198 (2021b)
19. S. Samantaray, A. Sahoo, D.K. Ghose, Prediction of sedimentation in an arid watershed using BPNN and ANFIS, in *ICT Analysis and Applications* (Springer, Singapore, 2020), pp. 295–302
20. Y. Seo, S. Kim, V.P. Singh, Multistep-ahead flood forecasting using wavelet and data-driven methods. *KSCE J. Civ. Eng.* **19**(2), 401–417 (2015)
21. V. Sehgal, R.R. Sahay, C. Chatterjee, Effect of utilization of discrete wavelet components on flood forecasting performance of wavelet based ANFIS models. *Water Resour. Manage.* **28**(6), 1733–1749 (2014)
22. A. Sharafati, A. Tafarojnoruz, M. Shourian, Z.M. Yaseen, Simulation of the depth scouring downstream sluice gate: the validation of newly developed data-intelligent models. *J. Hydro-Environ. Res.* **29**, 20–30 (2019)
23. A. Sharafati, M. Haghbin, M.S. Aldlemy, M.H. Mussa, A.W. Al Zand, M. Ali, S.K. Bhagat, N. Al-Ansari, Z.M. Yaseen, Development of advanced computer aid model for shear strength of concrete slender beam prediction. *Appl. Sci.* **10**(11), 3811 (2020)
24. S. Sridharam, A. Sahoo, S. Samantaray, D.K. Ghose, Estimation of water table depth using wavelet-ANFIS: a case study, in *Communication Software and Networks* (Springer, Singapore, 2021), pp. 747–754
25. R. Tabbussum, A.Q. Dar, Performance evaluation of artificial intelligence paradigms—artificial neural networks, fuzzy logic, and adaptive neuro-fuzzy inference system for flood prediction. *Environ. Sci. Pollut. Res.* **28**(20), 25265–25282 (2021)

26. K. Thangavel, A.K. Mohideen, Mammogram classification using ANFIS with ant colony optimization based learning, in *Annual Convention of the Computer Society of India*, Dec 2016 (Springer, Singapore, 2016), pp. 141–152
27. L.H. Xiong, A.Y. Shamseldin, K.M. O'Connor, A nonlinear combination of the forecasts of rainfall-runoff models by the first order Takagi-Sugeno fuzzy system. *J. Hydrol.* **245**(1–4), 196–217 (2001)
28. S. Zainuddin, F. Nhita, U.N. Wisesty, Classification of gene expressions of lung cancer and colon tumor using adaptive-network-based fuzzy inference system (ANFIS) with ant colony optimization (ACO) as the feature selection. *J. Phys. Conf. Ser. IOP Publ.* **1192**(1), 012019 (2019)

Evaluation of Different Variable Selection Approaches with Naive Bayes to Improve the Customer Behavior Prediction



R. Siva Subramanian, D. Prabha, J. Aswini, and B. Maheswari

Abstract Study of consumer behavior analysis within the enterprises is considered as paramount to identify how the customers are satisfied with the enterprise's services and also predicate how long a customer will exist in the enterprises in future. To achieve better customer satisfaction and to establish a sustainable relationship with the customers, the need for consumer analysis must be performed out expertly. To perform customer analysis in a better way, NB an ML model is studied and analyzed. But due to uncertainties present in the dataset like redundant, irrelevant, missing, and noisy variables makes the NB classifier to analyze wisely. Also violation of independence assumption between the variables in the dataset causes the NB to execute the customer analysis ineffectively. To improve customer analysis with these datasets and to strengthen the NB prediction, this research aims to use of variable selection approach. The variable selection methodology picks the best optimal variable subset by using some evaluation and search strategies to obviate the associated and unrelated variables in learning set and makes the NB assumption satisfied and enhance NB prediction in customer analysis. Three different variable selection methodology is applied in this research (filter, wrapper and hybrid) In filter seven different approaches—Information gain, Symmetrical uncertainty, Correlation attribute evaluation (CAE), OneR, Chi-square, Gain ratio, and ReliefF are applied and in wrapper five approaches—SFS, SBS, Genetic, PSO and Bestfirst are applied and in Hybrid approach combines both filter and wrapper approach. These three methodology works independently to selects the optimal variable subset and uses

R. Siva Subramanian (✉)
Anna University, Chennai, India

D. Prabha
Department of Computer Science and Engineering, Sri Krishna College of Engineering and Technology, Coimbatore, India
e-mail: prabha@skcet.ac.in

J. Aswini
Sree Vidyankethan Engineering College, Tirupati, India

B. Maheswari
Department of Computer Science and Engineering, Rajalakshmi Institute of Technology, Chennai, India

these variable subsets to enhance the NB prediction in customer analysis data. The experiment performed reveals NB using variable selection methodology gets better prediction compared to NB without variable approach. Also compare to the three variable selection methodology, the Hybrid approach gets better prediction to compare to the other two approaches.

Keywords Naive Bayes · Variable selection · Customer analysis · Machine learning · Hybrid approach · Decision making

1 Introduction

In the competitive dynamic business environment, to be successful in business the need of customer analysis and customer satisfaction toward the enterprise business is considered an important one. Performing customer analysis leads to effective understanding of customer behavior pattern toward the enterprises and customer expectation with the enterprises [1]. Gaining better knowledge of customers will aim to build different plans to enhance the enterprises and in turn plan how to provide better customer business and building a strong relationship with the customers. The role of performing customer analysis sort out different good things about customers such as potential customers within the enterprises, improving customer retention and acquisition, enhancing effective consumer service, optimizing enterprises market share, efficient business planning and overall improving business profitability [2]. But in the fast-moving business world, the data regarding the consumers are generated and gathered is faster than in traditional days. Since in the traditional days, the data regarding the consumers generated are very less in the volume and this may be due to under development of technologies. But today the situation is different since technologies are developed more than expected level the generation of customers data also enormous. By analyzing the customer data in proper methodology makes to sort different perspectives of data and further helps to improve the business goal and consumer satisfaction. Since the customer data are in huge dimensional, there is possible of the occurrences of redundant, noisy and missing variables in the datasets. These irrelevant variables make to degrade the classifier prediction and cannot bring out a better insight dimensional of customer analysis [3]. The problem of removing the irrelevant, noisy, redundant, and missing variables in the dataset is a prominent one to carry effective customer analysis. To address the problem with the huge dimensional dataset and to present the efficient customer analyses, NB a machine learning is deployed. NB an effective probabilistic learner relied on Bayes rule which is an extension of conditional probability. NB is quite simple to implement and with very small learning sets the NB can be performed. The classifier implies two unique conditions, which are variables that are conditional independent of other input predictor variables; and all variables are equal. But, violation of presumption made by NB results in poor prediction of the classifier. Unfortunately, in some real-time datasets NB-conditional independence is mostly violated, and the NB assumption proposed

are practically unrealistic. This happens due to generation of customer data from different possible ways and may hold same correlated variables. This makes the NB assumption practically unrealistic in real-time datasets. If these customer data are examined using NB, then surely poor performance can be witnessed. To improve the NB prediction with these customer data, there need an efficient methodology to identify the best variables that are strongly associated to output class and removing the correlated, noisy and missing variables from the datasets. To overcome the above problem variables selection methodology is proposed. These methods help to identify best variables, which can be further investigated with the NB. In this work different filter, hybrid and wrapper, variable selection are applied to evaluate how these variables selection chooses the best variables subset to improve NB prediction by removing the problem associated with huge dimensional data and satisfying the NB primary assumption made on variables. The performance assessment of NB is examined through variables subset obtained from different variables selection methods is computed and results are compared and presented. The article is followed by related works, variable selection, methodology, experiment and results, result analyses, conclusion.

2 Related Works

The author applies variable selection mechanism to evaluate the optimal variable subset in phage virion data for evaluation using NB [4]. The author uses CAE method to pick the best variables subset and by eliminating the redundant variables in the datasets. The variable method select 38 features which shows remarkable improvement in classifier prediction. The features obtained are examined using seven different classifier and result shows NB obtained best prediction compare to other classifier. The author uses two different variable selection mechanism to consider the best variables subset and applies the subset to examine with different ML techniques [5]. The author considers genetic and greedy search variable techniques to find the good feature group from the two different email spam dataset. The variables subset are further applied with different classifiers like GA, NB, SVM, and Bayesian and empirical results show SVM gets better results compared to others. Likewise from two variables methods, greedy search performs wisely to select a good variable subset when compared to others in the email spam datasets. The author proposes two variable selection approaches to select optimal variables subset in text categorization for the NB classifier [6]. The variable approach depends on Information Theory measures. The variable methods developed are maximum discrimination and $MD - x^2$. The proposed method uses a learning model in variable selection and evaluates selected variables optimality. The experiment is performed using different benchmark dataset and performance comparison is carried using other existing variable methods. Results show the proposed mechanism gets better performance with others. The author applies a variable selection mechanism to optimize the probability estimation of the NB [7]. The author performs the study on SBC methodology using

probability estimation and based upon the SBC, the research proposes improved SBC-CLL. The experiment is conducted using SBC and SBC-CLL and results show SBC-CLL performs better compared to SBC. The research results present a simple approach to enhance the NB for better probability estimation. The author proposes FVBRM variable selection to eliminate the irrelevant variables and to enhance the performance prediction in intrusion detection (IDS) [8]. Also, the research uses other three variable methods like CFS, IG, GR to compare the performance prediction of the proposed system. The study works to construct a computationally better approach to get a reduced variable subset for IDS. The experiment is performed using the CFS, IG, GR and FVBRM and results are compared. The selected variable subset is examined using the NB classifier. The results show FVBRM gets the better performance, but there exists time complexity in the FVBRM method. The author performed a study on cirrhotic patients who have taken the TIPS treatment [9]. In the cirrhotic medical dataset, all the attributes are not relevant to conduct the classification and to remove the irrelevant variables from the dataset variable selection mechanism is conducted. The research applies different filters and wrappers to generate a variable subset to examine using NB, selective-NB, Semi-NB, tree augmented-NB and K-dependence NB. The author applies a variable selection mechanism to compute the prominent features in phishing detection to perform better classification [10]. The research uses a correlation-based and wrapper method to perform variable selection using a phishing dataset. In the wrapper genetic and greedy FS are applied. The attribute subset obtained is evaluated using the NB, LR, and RF. The empirical results reveal the wrapper technique performs better in selecting variables compared to CFS for achieving better prediction in a classifier. The research concludes effective variable selection mechanism improves the classification accuracy significantly. The author applies a hybrid variable selection mechanism to obtain optimal variable subset to enhance classifier performance [3]. The hybrid method is based upon reliefF and genetic variable method. First using the relief method variables are selected and then the selected variable is further examined using a genetic method to get the best variable subset. The methodology is operated using the Wisconsin dataset to remove the unnecessary variables from the dataset. Then, from the subset of the reduced variables obtained are further examined using different ML classifiers like NB, DT, JRIP, KSTAR, RF and J48. The author performs importance of using variable selection in medical datasets [11]. To obtain maximum accuracy in the classification and enhance the model, finding out optimal variable subset is important one. For that purpose, the author applies a procedure named MFFS method. The MFFS procedure consists of four phrases. The variable subset obtained from the methodology is further examined with four various ML models like NB, SVM, MLP and J48. The MFFS procedure is carried using 22 various medical datasets. The results present best optimal performance is achieved compared to with existing schemes. The author conducted a study on improving NB prediction in the text classification domain [12]. Due to existence of unstructured and irrelevant variables in datasets, the need of performing variable selection mechanism is essential to get the relevant attribute group to enhance NB evaluation. To that purpose two phrase variable selection approach is deployed. The first phrase works by using a univariate method to select the subset of attributes and

then the second phrase works by applying variable clustering to choose the independent attribute set from the first phrase variable set. The research concludes NB performance is enhanced using the proposed variable selection mechanism and time complexity is optimal.

3 Naive Bayes

The NB-probabilistic classifier which builds the classifier effectively and the method assigns target class to the instances problem [13]. The model is based upon the principle of Bayes theorem and with a small learning set the parameter estimation is performed for the classification. The model considers input variables are independent of other input variables predictors, and also, variables are equal [14]. For an issue of classifying the given instances $Y = \{y_1, \dots, y_n\}$ where the n represents the input variables which are independent and it mark to the probabilities instances

$$p(O_i|y_1, \dots, y_n) \tag{1}$$

and O_i represents the classes.

In Eq. (1), suppose if there exit large variables set n or if the variables holds high number of values, then there exits the problem in the classifier probability table and becomes infeasible. By considering the Bayes Theorem, Eq. (1) denoted as

$$p(O_i|Y) = \frac{p(O_i)p(Y|O_i)}{p(Y)} \tag{2}$$

Equation (2) is written as

$$\text{posterior} = \frac{\text{prior} \times \text{likelihood}}{\text{evidence}} \tag{3}$$

In practice, denominator is not based upon the O and the y_i variables and denominator is constant.

$$p(O_i, y_1, \dots, y_n) \tag{4}$$

and rewritten as

$$p(O_i, y_1, \dots, y_n) = p(y_1, \dots, y_n, O_i) \tag{5}$$

$$= p(y_1|y_2, \dots, y_n, O_i)p(y_2|y_3, \dots, y_n, O_i) \dots p(y_{n-1}|y_n, O_i)p(y_n|O_i)p(O_i) \tag{6}$$

Based upon the Y input predictors are independent and under this principle [15]

$$p(O_i|y_1, \dots, y_n) \propto p(O_i, y_1, \dots, y_n) \quad (7)$$

$$= p(O_i) \prod_{i=1}^n p(y_i|O_i) \quad (8)$$

From Eq. (8) clearly explains that the X predictors are conditionally independent. But in the case of consumer data, the dataset generated are in high dimensional. This happens due to data collected are from different origin and possibly holds high dimensional variables about a customer. With these high dimensional, some research problem is associated there are 1. curse dimensionality, 2. with high dimensional large variables there possibly exhibit correlation within variables, 3. there exists uncertainties and unstructured in the dataset (noisy and missing), 4. increases the complexity of the model mechanism, 5. also affecting the efficiency of the classifier. Example consider the customer dataset $D = \{y_1, \dots, y_n|O_i\}$ where y presents the input predictor and O_i presents the output label. In the customer dataset, D all variables are not necessarily important, since some variables (y_1, \dots, y_n) in dataset may have chance of holding back irrelevant, noisy, missing and redundant variables [16]. By using these datasets, in Eq. (8) makes a clear violation of NB assumption and also makes the classifier to work wisely. With high dimensional variables there may be increases in time complexity and computational of the classifier and also make classifier to attach with overfitting. Considering the research issues with high dimensional customer data and to satisfy the NB presumptions in the customer data, there need an effective methodology to pick relevant variables by eliminating the unnecessary variables in the datasets. So for that purpose variable selection mechanism is deployed to make use of its advantages to select optimal relevant variable subset to enhance NB classification. Different filter, wrapper and hybrid methods are considered to generate variable subset to improve NB classification and variable selector are compared with the other variable selector.

4 Feature Selection

In ML, feature selection mechanism also can be called an attribute or variable selection. Variable selection is key ideas in ML to reduce data dimensionality and which in turn has significant impacts on the classifier's performance [17]. The objective of the attribute mechanism is to get the good subset of attributes either automatically or manually to examine with an ML classifier and with the aim of maximizing model prediction. The variable selection mechanism is carried to remove the redundant and irrelevant variables from the dataset without causing information loss in the datasets. Consider the dataset $D = \{x_1, \dots, x_n\}$ and the D holds n variables and m dimensions. The aim of variable selection to minimize m to m' and $m' \leq m$. Some advantages of constructing variables selection include 1. reduce the computational and make users to easily to develop classifier, 2. minimize the learning time,

3. eliminate curse dimensionality, 4. reduce variance and 5. enhance classification [1]. The variable selection is considered an important one while analyzing datasets like customer data $D = \{x_1, \dots, x_n | C_k\}$, since the customer dataset generated are in high dimensional and may have the chance of holding redundant, noisy, irrelevant and missing variables in the dataset. Also using these customer datasets the NB classifier should not be performed directly. If this is accomplished, then using redundant and irrelevant variables, NB model is learnt. This makes the NB classifier to perform badly in prediction problems and also time complexity is increased. Also the basic terms of NB are not to hold correlated attributes in dataset and also the variables must be equal. With these customer data the NB presumption cannot be satisfied and to eliminate the correlated and irrelevant with the customer data and to enhance NB performance variable selection carried out. Variable selection is categorized into filter, wrapper, hybrid and embedded approach [18]. The filter approach makes uses of some statistical mechanism to score the variables based upon the correlation with the output label. Then, based upon the cut-off value good subset variables are considered from learning data. The wrapper mechanism uses a learning model to get a good group of attributes from the original learning set. The hybrid approach is constructed by considering the advantage of both the filter and wrapper approach to choose the best variable subset. Mostly variable subset considered from the hybrid approach will exhibit possibly the best improvement in the classifier.

5 Proposed Methodology

To conduct better analysis using the customer data and to enhance the NB with these high dimensional variables, different variable selection mechanism is conducted. The variable selection applied are filter, wrapper and hybrid approach. The overall mechanism is given in Fig. 1.

5.1 Filter Methodology

Filter methods are simple, computationally fast, and depends entirely on the variable characteristics. The filter uses the different statistical measures to score the variables based upon the relevance with the output label. Filter chooses the variables based on 1. variables that are greatly correlated with output label, 2. less correlated with other input predictors 3. with more IG and MI value variables. The primary pros of filter is it would not depends on any ML models to pick the subset of the variables and also efficient in time and computation [19]. For analysis of huge dimensional variables the use of filter approach is considered as best option. Consider the customer learning set $D = \{x_1, \dots, x_n | C_k\}$ and the purpose of filter is to pick the subset of attributes that is relevant variables and remove all unwanted variables that don't trend to enhance NB performance. The variables in the learning set D are scored accordingly to

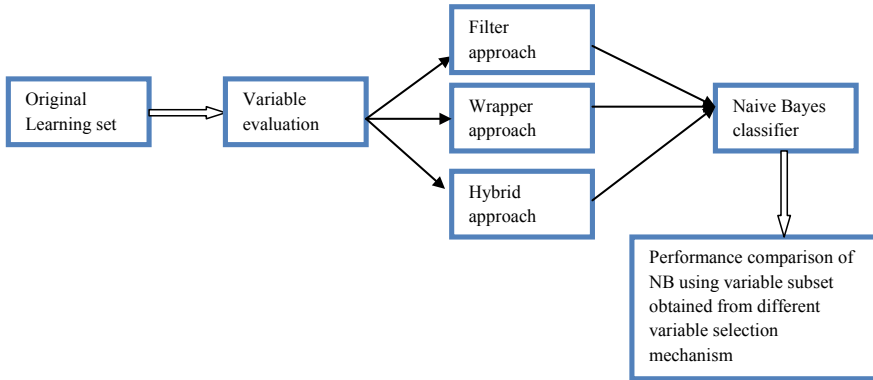


Fig. 1 Describes the overall methodology applied to select the variable subset using different variable selection mechanism to evaluate using Naive Bayes and compute the performance

their relevance or correlation with the output label. Then, from ranked variables list, threshold value is introduced to pick the optimal variable subset. Threshold value considered must be given special attention, since the filter method will only generate the ranked variable list. But selection of the optimal variable subset is based upon choosing the threshold value. In the filter, different methods like Symmetrical uncertainty, IG, ReliefF, CAE, ONE R, Chi-square and Gain ratio are encouraged to evaluate how NB performance improves. The filter mechanism is described in Fig. 2.

A. Symmetrical Uncertainty (SU)

Correlation computing of SU is relied upon Info theoretical entropy and based on this computes the uncertainty in the variable. Entropy E of the A attributes is described as

$$E(A) = - \sum P(a_i) \log_2(P(a_i)) \tag{9}$$

Then entropy of attributes A using another attribute B is described as

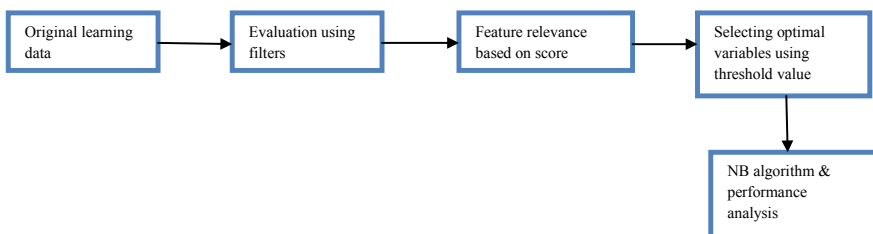


Fig. 2 Describes the filter methodology in selecting optimal variable subset for Naive Bayes

$$E(A|B) = - \sum_i P(b_j) \sum_j P(a_i|b_j) \log_2(P(a_i|b_j)) \quad (10)$$

here $P(a_i)$ represents prior probability A and $(P(a_i|b_j))$ represents posterior probability A using the values B . The entropy A minimizes, reflects further information A given by the B and it is referred to IG and it is denoted as

$$IG(A|B) = H(A) - H(A|B) \quad (11)$$

From Eq. (11), attribute A is greatly correlated to the attribute B compare to other attribute C , if it satisfy the $IG(A|B) > IG(C|B)$.

IG for the two attributes A and the B is symmetrical and it is required measure to find correlation between the variables. IG is biased toward the variables with high scores. Then, symmetrical uncertainty is described as

$$SU(A, B) = 2[IG(A|B)/(H(A) + H(B))] \quad (12)$$

Symmetrical normalizes the value amid (0, 1). The value $IG(A|B) = H(A) = H(B)$ and $SU(A, B) = 1$ represents using one variable completely identifies another variable (A and B are correlated) and the value $SU(A, B) = 0$ represents are uncorrelated [11].

SU Procedure

Consider the learning set $D = \{x_1, \dots, x_n|C_k\}$ and now to use SU approach to rank the variables accordingly to correlation with the output label.

1. Learning set $D(x_1, \dots, x_n, C)$
2. Compute $SU_{i,C}$ for each variables x_i and arrange the variables accordingly to $SU_{i,C}$
3. Get 1st variable x_r and for the next variable x_s
4. if no next variables exit then return variable subset, else
5. if $SU_{r,s} \geq SU_{s,C}$ true then eliminate x_s and proceed to get next variable
6. else false means, proceed to get next variables and
7. if no next variables exit then return variable subset and compute model performance.

B. Information Gain (IG)

Information gain variable selection is based upon the entropy method for evaluation of variable subset. IG computes how much a variable z gives information regarding the class C and compute correlation between the attributes. IG applies entropy to measure the diversity and to calculate the information impurity of target attribute. Consider the z a variable and the information entropy z attribute is described as

$$H(Z) = - \sum P(z_i) \log_2(P(z_i)) \quad (13)$$

$P(z_i)$ represents prior probabilities Z and the IG of the y attribute using the z attribute is described by

$$H(Z|Y) = - \sum_i P(y_j) \sum_j P(z_i|y_j) \log_2(P(z_i|y_j)) \quad (14)$$

here $P(z_i)$ represents prior probabilities of Z and $(P(z_i|y_j))$ represents posterior probabilities. Then, after eliminating the uncertainty, the IG represents the information amount, which is IE difference between the Z and Y variables with Z variable IG is described as

$$IG(Z|Y) = H(Z) - H(Z|Y) \quad (15)$$

The features with good IG are selected and the features with less IG are eliminated [8].

IG Procedure

Consider the learning set $D = \{z_1, \dots, z_n|C_k\}$ and to apply IG to the D learning set

1. Learning set $D(z_1, \dots, z_n, C)$
2. Compute the IG on Learning set D
3. Calculate the IG $IG(Z|Y) = H(Z) - H(Z|Y)$ using Eqs. 13 and 14
4. Select the optimal variables subset based upon threshold value and compute the model performance.

C. Gain Ratio (GR)

GR is non-symmetrical measure developed to overcome the problem of IG toward bias problem. GR is described as

$$GR = \frac{IG}{H(X)} \quad (16)$$

$$IG(X|Y) = H(X) - H(X|Y) \quad (17)$$

For predicating the attribute Y , the IG is normalized using the equation as described. With the normalization approach, the GR values fall between [0, 1]. If value is 1 then exists dependence between X and Y . If value is 0 then exists no dependence between X and Y [8].

D. OneR

OneR is referred as one rule and 1R approach computes each single variable individually. The OneR mechanism is rely upon only the variables and for each feature predictor it generates 1 rule. To make 1 rule for each individual variable, the frequency table is generated for each feature with the output label. The variables that hold small

errors are considered and depending upon the errors the subset of the variables is generated.

OneR Procedure

1. for each individual input variable predictor
2. each value of input variable predictor, rule as proceed
3. count the frequency of class target value
4. identify more frequent class label
5. assign the rule to that class target with respect to predictor value
6. compute total error in the rule of the each variable predictor
7. Then, consider the variable predictor with minimum error.

E. Correlation Attribute Evaluation (CAE)

CAE computes the worthiness of the variables by evaluating the correlation factor between the individual variable with the class. In the CAE, Pearson correlation approach is used to evaluate the correlation for each variables with the class label. The variables which exhibit high correlation with the class label are ranked up and variables which has less correlation with the class label are ranked down. So, based upon the correlation measures the variables are scored [20].

F. Chi-Square

Chi-square variable selection approach is applied to verify the independence between the two variables. Chi-square examine the variables with the class label and select the variables subset based upon the good chi-square scores. Chi-square is described as

$$X^2 = \frac{(\text{Observed frequency} - \text{Expected frequency})^2}{\text{Expected frequency}} \quad (18)$$

When two variables are uncorrelated, then the observed count and expected count are close to each other. The higher the chi-square scores implies variables that are strongly associated to output class, and these variable subset are further applied to NB training.

G. ReliefF

ReliefF is developed by Kononenko 1994 and filter variable selection approach to overcome the issue with the handling of incomplete data and restriction of two class problem. The method is good do handle with the noisy and incomplete variable. The reliefF place weight to each variables based upon the relevance to target class. The approach randomly choose R_i instance and evaluate the k nearest hits H_j for same class. Then approach evaluate the k nearest hits M_j for different classes, next the quality estimation $W[A]$ is carried out for all variables A based on R_i values using H_j and M_j

$$\begin{aligned}
 W[A] := & W[A] - \sum_{j=1}^k \frac{\text{diff}(A, R_i H_j)}{m.k} \\
 & + \sum_{c=\text{class}(R_i)} \frac{P(C)}{1 - P(\text{class}(R_i))} \sum_{j=1}^k \text{diff}(A, R_i, M_j(c))/(m.k) \quad (19)
 \end{aligned}$$

In reliefF initially weights are made to zero and the approach update the weight iteratively. Then at the end, variables with the more weights are considered [3].

5.2 Wrapper Methodology

Wrapper methodology is an effective approach to get optimal variables subset and to eliminate the redundant and irrelevant variables in the high dimensional customer datasets. The approach uses an induction algorithm to pick the right relevant variables in order to perform efficient customer analysis. The pros of wrapper methods are 1. wrapper check the features interaction, 2. provides optimal variables subset [17]. The evaluating of wrapper for selecting optimal variable subset progress through 1. Identify variable subset (go through with search strategy), 2. Construct the ML approach (in this stage, ML model considered to trained with the pre-selected variable subset), 3. Compute the classifier performance, 4. Repeat the above process to obtain a good relevant variable subset. The stopping point of the wrapper methodology depends upon 1. decreases model performance, 2. predefined no of variables reached [10]. Compare to filter this method exhibit high computational time, since this occurs due to involvement of search strategies and induction algorithm and possible of overfitting. In wrapper, totally five different approaches are applied (SFS, SBS, Genetic, PSO and Bestfirst). The wrapper technique is presented in Fig. 3.

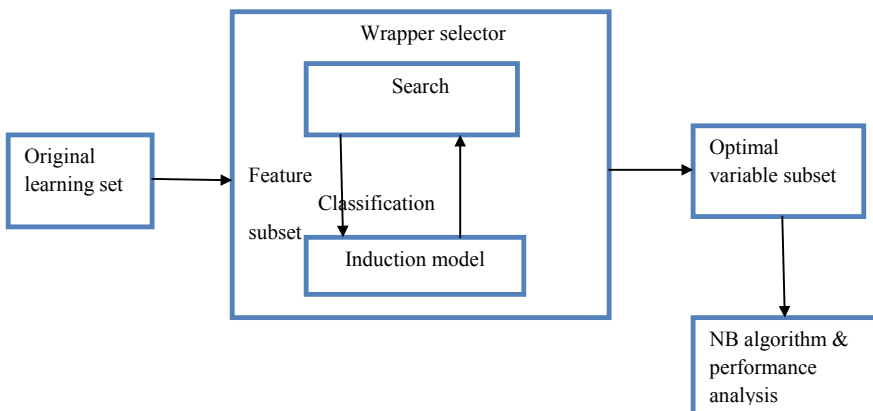


Fig. 3 Describes wrapper method to obtaining optimal variable subset for NB

A. Sequential Forward Selection (SFS)

The SFS approach is a greedy, wrapper-based algorithm that uses the induction model to select the best optimal variable subset. The usage of SFS trends to increase because of its simplicity and speed. SFS approach uses the bottom-up procedure and starting with empty set. From the blank set, SFS uses some evaluation functions to add the variable subset and follow these evaluation function repeatedly to obtain optimal variable subset. These variable subset obtained should possibly reduce the MS error. Further, the selected variable subset from the SFS trends to reduce the classification error. The cons of SFS is the method cannot eliminate the particular variables after the inclusion of other variables [21].

SFS Pseudo Code

1. Proceed with empty set: $Y_k = \{\emptyset\}$, $k = 0$.
2. Select next variable set:

$$x^+ = \arg \max_{x^+ \in Y_k} [J(Y_k + x^+)]$$

3. If $J(Y_k + x^+) > J(Y_k)$
 - a. Update $Y_{k+1} = Y_k + x^+$
 - b. ($k = k + 1$)
 - c. Back to step 2

B. Sequential Backward Selection (SBS)

The SBS approach is a greedy, wrapper-based algorithm that uses the induction model to select the best optimal variable subset. The SBS is quite opposite with the SFS approach. The SBS proceed with the full variable set and using some evaluation function to remove the variable subset which does not contribute more to improve the NB model accuracy. From the SBS procedure, optimal variable subset is generated and further used to optimize the NB prediction [21].

SBS Pseudo Code

1. Proceed with full variable set $Y_0 = X$
2. Eliminate worst variables $x^- = \arg \max_{x \in Y_k} J(Y_k - x)$
3. update $Y_{k+1} = Y_k - x^-$; $k = k + 1$
4. Go to step 2

C. Genetic Algorithm (GA)

GA is based upon search approach and rely on genetics and natural selection principles. GA is fast and efficient, effective parallel capabilities and yields good solutions. The selection of optimal variable subset is obtained through stages in GA [5].

D. Particle Swam Optimization (PSO)

PSO belongs to subset of EC and the approach is developed by Kennedy and Eberhart. PSO is population-based and each single individual of the population is considered as particles in search space.

E. Bestfirst (BF)

The Bestfirst approach is combining of depth and breath search. The best first can proceed through forward or backward or in bi-direction search to get the optimal attribute set [22, 23].

5.3 Hybrid Methodology

Then using the pre-selected variable subset is further applied to the wrapper method to get the best variable subset. The idea of hybrid lies that, filter method works fast in computing variable subset, but the result is not satisfied. Next, in wrapper results are satisfy, but there exhibit high computational. The overcome with selecting optimal variable subset and to perform in efficient time, the hybrid mechanism is proposed [17]. The pros of hybrid are 1. high performance, 2. compare to wrapper better computational, 3. better for large dimensional datasets [24]. Thus by looking at the advantages of filter and wrapper, the hybrid approach is proposed.

6 Experimental Design

The experimental study to enhance the customer analysis using NB is studied and presented detailed in Sect. 4. In this research, three different methodology is performed. For each methodology experimental are performed and compared with NB without applying any methodology.

6.1 Validity Scores and Datasets

The customer data is obtained from UCI and the data holds 45,211 instances. The customer data consists of 17 variables. The experiment conducted is compared by using standard metrics like Accuracy, TPR, Sensitivity, PPV, FNR, FPR [25–27]. List of attributes present in customer dataset are given below:

1. Age (N), 2. Job (C), 3. Martial (C), 4. Education (C), 5. Default (C), 6. Balance (N), 7. Housing (C), 8. Loan (C), 9. Contact (C), 10. Month (C), 11. Day (N), 12. Duration (N), 13. Campaign (N), 14. Pdays (N), 15. Previous (N), 16. Pout come (N), 17. Yes (Output Label). The Character N refers to Numeric Variable and character C refers to Categorical variable.

6.2 Experimental Procedure

1. First the filter procedure is used to select the best variable subset to evaluate with the NB. In this experiment, seven different filter mechanisms are used (Symmetrical uncertainty, IG, GR, OneR, CAE, Chi-square and ReliefF). Using the filter approach, the variables are scored accordingly to relevance with the output label. To chose the best variable subset from the scored variable set, the threshold value is applied. Here, three different threshold value is selected 10%, $\log_2 n$, and 65%. The selection of different threshold value to analysis how the accuracy of NB prediction is achieved with these values (10%, $\log_2 n$, and 65%). The results are tabulated in Tables 1, 2 and 3.
2. Second, the wrapper approach is applied to get the right optimal variable group to evaluate with the NB. Here, five different wrapper mechanism is used (SFS, SBS, Genetic, PSO and Bestfirst). The results are tabulated in Table 4.
3. Third, the Hybrid variable selection is applied to select the best optimal variable subset to evaluate with the NB. The hybrid procedure uses both filter and

Table 1 Accuracy, recall, specificity, FNR, FPR and precision of NB using variable subset obtained from different filter method by setting 10% threshold value and NB (without variable selection)

	Accuracy	Recall	Specificity	Precision	FNR	FPR
SU	89.1951	0.303	0.969	0.571	0.696	0.300
IG	89.1951	0.303	0.969	0.571	0.696	0.300
GR	89.1951	0.303	0.969	0.571	0.696	0.300
ONER	89.1951	0.303	0.969	0.571	0.696	0.300
CHI-SQUARE	89.1951	0.303	0.969	0.571	0.696	0.300
CAE	89.1951	0.303	0.969	0.571	0.696	0.300
RELIEFF	88.1887	0.303	0.969	0.571	0.696	0.300
NB	88.0073	0.528	0.926	0.488	0.472	0.074

Table 2 Accuracy, recall, specificity, FNR, FPR and precision of NB using variable subset obtained from different filter method by setting $\log_2 n$ threshold value and NB (without variable selection)

	Accuracy	Recall	Specificity	Precision	FNR	FPR
SU	88.4077	0.385	0.950	0.506	0.615	0.0498
IG	89.2703	0.394	0.958	0.559	0.606	0.042
GR	88.4077	0.385	0.950	0.506	0.615	0.0498
ONER	89.2703	0.394	0.958	0.559	0.606	0.042
CHI-SQUARE	89.2703	0.394	0.958	0.559	0.606	0.042
CAE	89.7989	0.391	0.965	0.598	0.609	0.034
RELIEFF	88.9452	0.287	0.9693	0.553	0.713	0.0307
NB	88.0073	0.528	0.926	0.488	0.472	0.074

Table 3 Accuracy, recall, specificity, FNR, FPR and precision of NB using variable subset obtained from different filter method by setting 65% threshold value and NB (without variable selection)

	Accuracy	Recall	Specificity	Precision	FNR	FPR
SU	88.4475	0.474	0.938	0.507	0.526	0.061
IG	88.4187	0.470	0.939	0.505	0.530	0.060
GR	88.4475	0.474	0.938	0.507	0.526	0.061
ONER	89.4052	0.439	0.954	0.560	0.561	0.045
CHI-SQUARE	88.4187	0.470	0.939	0.505	0.530	0.060
CAE	88.1334	0.493	0.932	0.493	0.507	0.067
RELIEFF	89.5755	0.463	0.953	0.567	0.537	0.046
NB	88.0073	0.528	0.926	0.488	0.472	0.074

Table 4 Accuracy, recall, specificity, FNR, FPR and precision of NB using variable subset obtained from different wrapper methods and NB (without variable selection)

	Accuracy	Recall	Specificity	Precision	FNR	FPR
PSO	90.0356	0.427	0.963	0.605	0.573	0.036
GENETIC	90.0135	0.414	0.964	0.607	0.586	0.035
SFS	89.852	0.458	0.956	0.548	0.542	0.043
SBS	90.0356	0.427	0.963	0.605	0.573	0.036
BEST FIRST	89.852	0.458	0.956	0.548	0.542	0.043
NB	88.0073	0.528	0.926	0.488	0.472	0.074

wrapper to select the variable subset. Using filter approach variables is ranked accordingly to correlation with the output predictor and the threshold value is set into 65% to select optimal variable subset. The next wrapper approach is applied to the pre-selected variable subset from the filter mechanism to obtain the best optimal variable subset. The results are presented in Tables 5, 6, 7, 8 and 9.

4. Fourth NB is evaluated without considering any feature selection mechanism and results of filter-NB, wrapper-NB, hybrid-NB and standard NB are compared and presented.

Table 5 Accuracy, recall, specificity, FNR, FPR and precision modeling NB using variable subset obtained from HYBRID (PSO and CAE, RELIEFF)-NB method and NB (without variable selection)

	Accuracy	Recall	Specificity	Precision	FNR	FPR
PSO and CAE	90.044	0.424	0.963	0.607	0.576	0.036
PSO and RELIEFF	90.035	0.427	0.9630	0.605	0.573	0.036
NB	88.0073	0.528	0.926	0.488	0.472	0.074

Table 6 Accuracy, recall, specificity, FNR, FPR and precision modeling NB using variable subset obtained from HYBRID (GENETIC and SU, GR, CAE, RELIEFF)-NB method and NB (without variable selection)

	Accuracy	Recall	Specificity	Precision	FNR	FPR
GENETIC and SU	90.0113	0.420	0.9637	0.605	0.580	0.0362
GENETIC and GR	90.0113	0.420	0.9637	0.605	0.580	0.0362
GENETIC and CAE	90.044	0.424	0.963	0.607	0.576	0.036
GENETIC and RELIEFF	90.0356	0.427	0.9630	0.605	0.573	0.036
NB	88.0073	0.528	0.926	0.488	0.472	0.074

Table 7 Accuracy, recall, specificity, FNR, FPR and precision modeling NB using variable subset obtained from HYBRID (SFS and SU, IG, GR, CHI-SQUARE, CAE, RELIEFF)-NB method and NB (without variable selection)

	Accuracy	Recall	Specificity	Precision	FNR	FPR
SFS and SU	89.944	0.409	0.9644	0.604	0.591	0.0355
SFS and IG	89.998	0.414	0.9643	0.606	0.586	0.0356
SFS and GR	89.944	0.409	0.9644	0.604	0.591	0.0355
SFS and CHI-SQUARE	89.998	0.414	0.9643	0.606	0.586	0.0356
SFS and CAE	89.852	0.458	0.956	0.584	0.542	0.0431
SFS and RELIEFF	89.916	0.412	0.9637	0.601	0.588	0.036
NB	88.0073	0.528	0.926	0.488	0.472	0.074

Table 8 Accuracy, recall, specificity, FNR, FPR and precision modeling NB using variable subset obtained from HYBRID (SBS and CAE, RELIEFF)-NB method and NB (without variable selection)

	Accuracy	Recall	Specificity	Precision	FNR	FPR
SBS and CAE	90.044	0.424	0.963	0.607	0.576	0.036
SBS and RELIEFF	90.035	0.427	0.9630	0.605	0.573	0.036
NB	88.0073	0.528	0.926	0.488	0.472	0.074

Table 9 Accuracy, recall, specificity, FNR, FPR and precision modeling NB using variable subset obtained from HYBRID (BF and SU, IG, GR, ONER, CHI-SQUARE, RELIEFF)-NB method and NB (without variable selection)

	Accuracy	Recall	Specificity	Precision	FNR	FPR
BF and SU	90.0113	0.420	0.9637	0.605	0.580	0.0362
BF and IG	89.998	0.414	0.9643	0.606	0.586	0.0356
BF and GR	90.0113	0.420	0.9637	0.605	0.580	0.0362
BF and ONER	89.8719	0.392	0.9659	0.603	0.608	0.0340
BF and CHI-SQUARE	89.998	0.414	0.9643	0.606	0.586	0.0356
BF and RELIEFF	89.9449	0.409	0.9644	0.604	0.591	0.0355
NB	88.0073	0.528	0.926	0.488	0.472	0.074

6.3 Experimental Results

The results analysis from Table 1 shows that compared to standard NB all the filter-NB approach studied in this research improves the customer analysis prediction. Standard NB gets customer analysis up to 88.0073, but other filter-NB gets prediction up to 89.1951 except reliefF-NB. Here, 10% threshold value is considered to get the best variable subset from the variable ranking list obtained after filter evaluation. Except for ReliefF-NB method all other filter approach produces the same prediction analysis. With 10% threshold value is best for all filter approach to choose the variable subset. But for the reliefF 10% threshold is not enough to improve the NB.

The result analysis presented in Table 2 describes filter-NB methodology performs better customer analysis compare to NB (without variable selection). The standard NB achieves prediction up to 88.0073, but filter-NB achieves higher prediction up to 89.7989 and minimum prediction up to 88.4077. Here, the threshold value $\log_2 n$ is set to obtain optimal feature subset. The $\log_2 n$ threshold value choose the best variable subset for chi-square, oneR, IG and SFS compared to ReliefF, SU and GR.

The results analysis from Table 3 shows the filter-NB approach improves better customer prediction compared to standard NB. The NB (without variable selection) achieves prediction up to 88.0073, but in the case of filter-NB approach achieves higher prediction up to 89.5755 and minimum prediction up to 88.1334. Here, the threshold value is set to value 65%. The threshold value is good for filter method oneR and reliefF compared to SU, GR, IG, CHI-SQUARE and CAE.

The results analysis presented in Table 4 shows wrapper-NB methodology achieves better customer analysis compare to standard NB (without variable selection). The PSO and SBS-NB gets a higher prediction of 90.0356, but standard NB achieves only 88.0073. In wrapper, total five methods are applied from that PSO and SBS perform better compare to the other wrapper methods.

The results analysis presented from Tables 5, 6, 7, 8 and 9 reveals hybrid-NB approach performs better prediction compared to the standard NB. The NB (without variable selection) gets only up to 88.0073 prediction analysis. But the hybrid method PSO and CAE archives 90.044. Then, the next hybrid method genetic and CAE achieves 90.044 and genetic and reliefF achieves 90.0356. Then, the next hybrid method SFS and SU gets 89.944 and SFS and IG archives 89.998 and SFS and GR achieves 89.944 and SFS and chi-square achieves 89.998 and SFS and reliefF achieves 89.916. Then, next hybrid method SBS and CAE achieves 90.044 predictions. Then, the next hybrid method BF and SU achieves 90.0113 and BF and IG achieves 89.998 and BF and GR achieves 90.0113 and BF and oneR achieves 89.8719 and BF and chi-square achieves 89.998 and BF and reliefF achieves 89.9449 prediction improvement The hybrid method is proceed using filter and wrapper method. First using the filter method, the variables are ranked and using a 65% threshold value optimal subset is obtained. Then, from the pre-selected variable subset is further applied wrapper method to obtain the best optimal variable subset.

6.4 Results Analysis

The experiment is carried out with three different methodologies, and the findings obtained are summarized. Tables 1, 2 and 3 describes the result analysis of NB using filter methodology with the different threshold value. From the filter-NB methodology, the research work witness CAE at $\log_2 n$ threshold value selects a best variable subset and obtains maximum accuracy prediction of 89.7989. Table 4 describes the result analysis of NB using wrapper methodology. The PSO and SBS approach selects the best optimal variable subset to improve customer analysis using NB and obtains a maximum accuracy prediction of 90.0356. Tables 5, 6, 7, 8 and 9 describe hybrid-NB methodology to improve the NB. The results reveals the hybrid approach PSO and CAE and GENETIC and CAE and SBS and CAE selects the best optimal variable subset and improves NB prediction by 90.044. The study shows the hybrid approach performs better prediction compared to filter and wrapper from the results obtained from the three distinct approach experiments. Also compare to time complexity, hybrid method performs better compare to the wrapper. As fast as the filter method, the hybrid method performs efficiency to select optimal subset and results obtained are better compare to filter.

6.5 Research Findings

1. In the filter method to select the best optimal variable subset three different threshold values are considered to examine how the threshold value is important to select the variable subset. From the three threshold value, $\log_2 n$ is considered as better, since the NB prediction improves up to 89.7989. Second, next best variable subset is generated at 65% threshold value 89.5755 and the third next three best variable subsets are generated at $\log_2 n$ 89.2703. From this analysis shows the setting of threshold value should be given better consideration.
2. From the different filter approaches, CAE performs better in improving customer analysis using NB compared to other filter approaches. Compare to Wrapper and hybrid the filter performs fast, but the prediction is not satisfying compared to others.
3. The wrapper selects the best optimal variable subset by using SBS and PSO and improves prediction accuracy 90.0356 better compared to the filter approach. But the method is computational inefficient while selecting the variable subset, since there exists an induction model with a wrapper to choose the variable subset.
4. The hybrid approach selects the best optimal variable subset by using CAE with PSO, GENETIC and SBS, and improves the prediction accuracy of 90.044 compared to filter and wrapper. The hybrid approach uses the pros of filter and wrapper to select optimal variable subset. Hybrid works computational efficiency and produces the best variable subset compared to wrapper and filter.

Since the method first uses a filter to identify the best variable set and then using the pre-selected variables, the wrapper mechanism is constructed to select the best optimal variable subset. So, time complexity is improved using hybrid compared to the wrapper. Then, fast and best optimal variable subset is obtained in hybrid compared to filter.

5. The findings of the experiment show that hybrid-NB performs better than filter and wrapper methodology.

7 Conclusion

To perform enhanced customer analysis, the Naive Bayes ML approach is studied and analyzed. But due to uncertainties associated with high dimensional data and the infringement of independence assumption between the variables in the dataset imposed by the NB, makes the NB to function inefficiently. This can be made possible by eliminating redundant, noisy and missing variables and to select the most relevant variables highly correlated with a class label. To arrive at the solution for the above-said problem, three different variable selection methodologies are studied and analyzed. Filter, wrapper and hybrid methodology are examined to get the best variable group to optimize customer analysis using the NB. The experimental analyses presents CAE filter-NB improves prediction up to 89.7989. Then, wrapper SBS and PSO and improve the prediction accuracy of 90.0356. Then, hybrid using CAE with PSO, GENETIC and SBS and improves the prediction accuracy of 90.044. Compare to three methodologies to improve customer analysis hybrid-NB performs better with better computational time and selects optimal variable subset.

References

1. J. Liou, G.-H. Tzeng, A dominance-based rough set approach to customer behavior in the airline market. *Inf. Sci.* **180**, 2230–2238 (2010)
2. R. Siva Subramanian, D. Prabha, A survey on customer relationship management, in *4th International Conference on Advanced Computing and Communication Systems (ICACCS)*, Coimbatore (2017), pp. 1–5. Electronic ISBN: 978-1-5090-4559-4. <https://doi.org/10.1109/ICACCS.2017.8014601>
3. I. Sangaiah, A. Vincent Antony Kumar, Improving medical diagnosis performance using hybrid feature selection via relieff and entropy based genetic search (RF-EGA) approach: application to breast cancer prediction. *Clust. Comput.* (2018)
4. P.-M. Feng, H. Ding, W. Chen, H. Lin, Naïve Bayes classifier with feature selection to identify phage virion proteins. *Comput. Math. Methods Med.* 1–6 (2013)
5. S.K. Trivedi, S. Dey, Effect of feature selection methods on machine learning classifiers for detecting email spams, in *Proceedings of the 2013 Research in Adaptive and Convergent Systems on—RACS* (2013)
6. B. Tang, S. Kay, H. He, Toward optimal feature selection in Naive Bayes for text categorization. *IEEE Trans. Knowl. Data Eng.* **28**(9), 2508–2521 (2016)
7. L. Jiang, H. Zhang, Learning Naive Bayes for probability estimation by feature selection, in *Lecture Notes in Computer Science* (2006), pp. 503–514

8. S. Mukherjee, N. Sharma, Intrusion detection using Naive Bayes classifier with feature reduction. *Procedia Technol.* **4**, 119–128 (2012)
9. R. Blanco, I. Inza, M. Merino, J. Quiroga, P. Larrañaga, Feature selection in Bayesian classifiers for the prognosis of survival of cirrhotic patients treated with TIPS. *J. Biomed. Inform.* **38**(5), 376–388 (2005)
10. R.B. Basenet, A.H. Sung, Q. Liu, Feature selection for improved phishing detection, in *Lecture Notes in Computer Science* (2012), pp. 252–261. https://doi.org/10.1007/978-3-642-31087-4_27
11. S. Sasikala, Appavu alias S. Balamurugan, S. Geetha, Multi filtration feature selection (MFFS) to improve discriminatory ability in clinical data set. *Appl. Comput. Inform.* **12**(2), 117–127 (2016)
12. S. Dey Sarkar, S. Goswami, A. Agarwal, J. Aktar, A novel feature selection technique for text classification using Naive Bayes. *Int. Sch. Res. Notices* 1–10 (2014)
13. C.B. Christalin Latha, S.C. Jeeva, Improving the accuracy of prediction of heart disease risk based on ensemble classification techniques. *Inform. Med. Unlocked* **100203** (2019)
14. J.I.Z. Chen, P. Hengjinda, Early prediction of coronary artery disease (CAD) by machine learning method—a comparative study. *J. Artif. Intell.* **3**(01), 17–33 (2021)
15. R. Siva Subramanian, D. Prabha, Prediction of customer behaviour analysis using classification algorithms, *AIP Conf. Proc.* **1952**, 020098 (2018). ISBN: 978-0-7354-1647-5. <https://doi.org/10.1063/1.5032060>
16. J. Abellan, F. Castellano, Improving the Naive Bayes classifier via a quick variable selection method using maximum of entropy. *Entropy* **19**(6), 247 (2017)
17. F. Moslehi, A. Haeri, A novel hybrid wrapper–filter approach based on genetic algorithm, particle swarm optimization for feature subset selection. *J. Ambient Intell. Human. Comput.* **11**, 1105–1127 (2020)
18. D. Prabha, R. Siva Subramanian, S. Balakrishnan, M. Karpagam, Performance evaluation of Naive Bayes classifier with and without filter based feature selection. *Int. J. Innov. Technol. Explor. Eng. (IJITEE)* **8**(10), 2154–2158 (2019). ISSN: 2278-3075. <https://doi.org/10.35940/ijitee.J9376.0881019>
19. R. Siva Subramanian, D. Prabha, J. Aswini, B. Maheswari, M. Anita, Alleviating NB conditional independence using multi-stage variable selection (MSVS): banking customer dataset application. *J. Phys. Conf. Ser.* **1767**, 012002 (2021). <https://doi.org/10.1088/1742-6596/1767/1/012002>
20. S. Gnanambal, M. Thangaraj, V.T. Meenatchi, V. Gayathri, Classification algorithms with attribute selection: an evaluation study using WEKA. *Int. J. Adv. Netw. Appl.* **09**(06), 3640–3644 (2018)
21. R. Panthong, A. Srivihok, Wrapper feature subset selection for dimension reduction based on ensemble learning algorithm. *Procedia Comput. Sci.* **72**, 162–169 (2015)
22. A. Balgun, S. Basri, A. Sobri, S. Jadid Abdulkadir, Performance analysis of feature selection methods in software defect prediction: a search method approach. *Appl. Sci.* **9** (2019)
23. S. Dinakaran, P.R.J. Thangaiah, Comparative analysis of filter-wrapper approach for random forest performance on multivariate data, in *2014 International Conference on Intelligent Computing Applications* (2014)
24. R. Siva Subramanian, D. Prabha, Optimizing Naive Bayes probability estimation in customer analysis using hybrid variable selection, in *Computer Networks and Inventive Communication Technologies*, ed. by S. Smys, R. Palanisamy, Á. Rocha, G.N. Beligiannis. *Lecture Notes on Data Engineering and Communications Technologies*, vol. 58 (Springer, Singapore, 2021)
25. R. Siva Subramanian, D. Prabha, Customer behavior analysis using Naive Bayes with bagging homogeneous feature selection approach. *J. Ambient Intell. Human. Comput.* **12**, 5105–5116 (2021)
26. D. Prabha, K. Ilango, Customer behavior analysis using rough set approach. *J. Theor. Appl. Electron. Commer. Res.* **8**, 21–33 (2013)
27. T. Vijayakumar, Posed inverse problem rectification using novel deep convolutional neural network. *J. Innov. Image Process. (JIIP)* **2**(03), 121–127 (2020)

Personalized Abstract Review Summarization Using Personalized Key Information-Guided Network



Nidhin S. Dharan and R. Gowtham

Abstract We are proposing a personalized summarization model, which generates an abstractive summary of a random review based on the preference of a specific user. The summary will account the user's preference on different aspects present in the review. We put forward a Personalized Key Information Guided Network (PKIGN) that pools both extractive and abstractive methods for summary generation. Specifically, keywords present in the review are extracted which are specific to that user, and these keywords are used as key information representation to guide the process of generating summaries. Additionally, Pointer-Guide mechanism is employed for obtaining long-term value for decoding. We evaluate our model on a new Trip-Advisor hotel review dataset, comprising of 140,874 reviews from 41,600 users. Combining the results from both human evaluation and quantitative analysis, it is seen that our model achieves better performance than existing models on personalized review summarization in case of hotel reviews.

Keywords Extractive and abstractive summarization · Encoded · PKIGN · Pointer-guide

1 Introduction

Review summarization has gained a great success owing to the introduction of models sequence-to-sequence [1], transformers [2] and their variants [3]. The main objective of review summarization is to create a condensed summarization of the single or multiple reviews. With the exponential growth of e-commerce websites, it has been widely researched (Fig. 1).

This paper discusses introducing the concept of personalization to review summarization, which has not been discussed extensively in previous research. A model has

N. S. Dharan (✉) · R. Gowtham

Department of Computer Science Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India

R. Gowtham

e-mail: r_gowtham@cb.amrita.edu

Review : The location is good, the room is cool, the price is reasonable, but the breakfast is pricey. the check in and check out is pretty quick. The staff is friendly. Most of the stuff in the mini-bar is free. The room I stayed in has no view, but that's fine for me, as I don't spend a lot of time in the room	
User	Summary
Sachin	affordable price, room is nice
Rohit	Great location
Virat	friendly staff and nice room

Fig. 1 The main motivation behind personalized key information-guided network is that for the same review different users are likely to create different summaries according to their own preferences

been built on creating personalized summarization based on important words [4] which does not provide good results. In case of a review, different users may care about different aspects according to their personal preferences. In case of our dataset which consists of hotel reviews, different aspects have been identified as location, room, value, facility, service and food. User A may care about the aspects service and room more than price, while user B may care about price. Therefore, we are proposing a model which takes into account user-specific aspects while creating summaries.

Personalized review summarization has a wide range of application across all online consumer products, such as TripAdvisor and Zomato. Users write their reviews across all of these platforms, and one function is to provide summarizes of other reviews according to their preferences or aspects. Using classical models for summarization, every user will view the same summary for every review. Using our proposed model, for the same review, different users will be able to see different summaries, thus providing a personalized service to each and every customer.

To perform personalized review summarization, we propose personalized key information-guided network which is based on sequence-to-sequence and key information guide network. Our model has major updates done in two parts.

Firstly, we create a corpus of all the reviews and summaries from a user and find out the most common words used by each user. Each user will talk more about the aspect they care about. By this method, we find out which aspects each user cares about.

Secondly, for each review, we are extracting keywords using TextRank [5], and we are only keeping keywords containing using specific keywords. By these methods, these user-specific aspects will be given more importance while generating summary.

To validate our approach, we have taken a dataset from paper. With quantitative and human evaluation approaches, we present that our model has achieved better

results for personalized review summarization in the case of hotel reviews. Our contributions for this project are as follows:

- To the best of our knowledge, we are the first ones to propose a personalized key information-guided network by using user-specific aspect keywords from reviews for personalized review summarization.
- For evaluating our model, we have created a novel dataset Hotel Reviews.

2 Related Works

Abstractive summarization has been studied across numerous papers throughout many years. Abstractive text summarizations have been extensively used for review summarization. RNNs are mainly leveraged for most of the natural language processing tasks as a result of very promising results. After the introduction of Encoder Decoder models in neural machine translation [6], a neural attention encoder–decoder model with feed-forward networks was introduced for abstract summarization [7]. In this model, an attention mechanism is added to the framework by taking into account the context vectors in hidden state of the encoder that helps in decoding the target sequences and achieved state-of-the-art results for DUC-2004 and Gigaword, two sentence-level summarization datasets. The basic architecture of our model is inspired from the sequence-to-sequence model. The approach which is focused on the attention mechanism has been augmented with recurrent decoders [3], abstract meaning representations [8], hierarchical networks [9] and variational autoencoders [10] improved performance on the respective datasets. A segment-to-segment neural transduction model [11] for sequence-to-sequence framework. The model introduces a latent segmentation which determines correspondences between tokens of the input text and the output text. Experimentation performed on the proposed transduction model shows good results on the Gigaword dataset.

While the sequence-to-sequence model with attention was getting promising results, some problems still existed. In these models, for each time step, the decoder uses a fixed target vocabulary to the given probability distribution. These types of situations can lead to OOV word errors. One method of solving this is where the size of target vocabulary is increased, but this will result in increase of computational complexity needed to find out the Softmax function across all possible words in the target vocabulary. To solve this, a model which uses soft copy mechanism was introduced in PGN [12] model which uses pointer generation network, which is a hybrid network, which lets us both copy the words from keywords and also generate the words from target vocabulary. PGN was able to achieve state-of-the-art results on CNN/Daily Mail dataset. The soft copy mechanism, mentioned in the above architecture, is added to our model with personalized keyword.

Another drawback of the traditional method is that there is no way to filter out the secondary information. In the existing methods, all the information given at the encoder is passed on to the decoder state for generation without checking if they are useful or not. This can often lead to the model focusing on unimportant information

while generating summaries. SEASS [13] network uses a selective mechanism to control the information flow which highlights important information and release the burden of the decoder which performed significantly better on ROGUE score for English Gigaword, DUC 2004 and MSRATC test sets. Therefore, focusing on the keywords while summarizing will improve summarization, and this principle is used in our model. Other models use dual-encoding [14] propose an LSTM-CNN approach which create new sentences by searching more fine-grained fragments than sentences and semantic phrases. The dual-encoding approach is consisting of two main stages. The first stage extracts phrases from source sentences which is followed by a stage that generates text summaries using deep learning. Existing models which create abstractive summary based on the opinions and arguments [15] also fail to filter the personalized information. There have been other methods which summarize texts [16–18] and analyze product reviews [19] but fail in personalization. Our model is inspired from Guiding Generation for Abstractive Text Summarization Based on Key Information Guide Network [20] which used key information-guided mechanism and soft copy mechanism. This model was validated on CNN/Daily Mail dataset. The main difference in our model is that we have devised a way to extract personalized information from the users' previous reviews and use this information for key information-guided mechanism and soft copy mechanism.

Another downside is that the encoder-decoder methods do not provide better results for longer texts. Transformer model [2] was introduced using stacked self-attention and point-wise fully connected layers for both the encoder and decoder. This model achieved state-of-the-art results in WMT 2014 English–German dataset. The main advantage of this model is parallelizable and requiring significantly less time to train. We tried implementing a guided mechanism in the transformer, but it did not provide better results.

User-aware sequence network [21] is used for personalized review summarization. This model is based on S2S network with user-aware encoder and user-aware decoder, where selective mechanism is used in user-aware encoder to highlight user-specific information in the review. USN achieved state-of-the-art rogue for personalized summarization. The user-aware decoder identify the user writing style and uses a soft copy mechanism to obtain summaries. But our model differs from this user-aware sequence network as we use personalized keywords to guide our summarization model. We have devised a method to extract user preferences which is unique to our model.

3 Problem Formulation

Suppose that we have a corpus with N user-review-summary triplets, where user u writes a review x and a summary y . Review x is a sequential input where $x = \{x_1, x_2, \dots, x_i, \dots, x_n\}$ is a sequence of n number of words, and i is the index of the input words. Summary y is the shorter output where $y = \{y_1, y_2, \dots, y_i, \dots, y_m\}$ of

Table 1 Dataset description for Tripdata

Reviews	140,874
Summaries	140,874
Users	41,745
Reviews/user	3.37

m number of words where $n > m$. The aim of our model is to generate a summary y from review x by attending to u 's aspects preference on summarizing reviews.

4 Dataset

We use hotel review dataset from Trip Advisor [22]. We create a new dataset Tripdata from this data. Tripdata is collected from identifying manipulated offerings on review portals [22], which was collected from TripAdvisor which is a travel review website. The data contains user-generated reviews along with author names and titles. The title of a review provides summarized information of the review. Moreover, there are many noisy samples found in the data. Different filtering techniques are used to obtain a clean data such as:

- i. Review length filter is used to remove reviews less than twenty five words and more than five hundred words. This is done to remove reviews that are too short and too long
- ii. Title length filter is used to remove titles less than five words. This is done to remove titles which are too short
- iii. Aspect-based filter is used to remove titles which do not have any aspects relating to hotel reviews. For hotel review data, we have mentioned six aspects along with their seed words in Table 1. The seeds words mentioned are expanded with boot strapping method by aspect segmentation algorithm [23]. Finally, the samples where the seed words are not present in the title are removed.

Statistics for Tripdata are provided in Table 1. We randomly split the dataset into 5000 user-review-summary triplets for test, 1500 user-review-summary triplets for validation and the rest for training purpose.

5 Our Model

The classical encoder–decoder network works with the review text as the input and the summary text as the output, where there is limited control over generation and key information may be missing in the summary. In addition to this, we also want the summary to be guided by personalized key information. That is where personalized key information-guided network is introduced shown in Fig. 2.

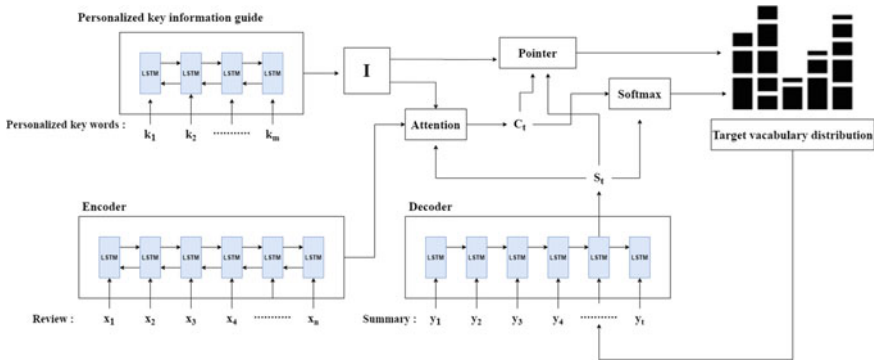


Fig. 2 Personalized key information-guided network architecture

5.1 Preprocessing

In detail, the first step is to identify which aspect each user cares about. Since we are working exclusively on hotel reviews, we can specify which aspects we should look for [21]. In addition to the seed words mentioned [21], we have also included some words with respect to this use case (Table 2). Firstly, we remove the stopwords from all the reviews (to avoid noise) and create a corpus with all the reviews and summaries for each user. Secondly, we find the most commonly used 30 words from each corpus (users will talk more about the aspects they care about). Then, we identify which aspects these words relate to if any. By this method, we identify aspect preference for each user.

The second step is to keywords using TextRank algorithm for each review. Then, we filter out the keywords which are not related to the aspects mentioned for the specific user. By this method, we will be able to get personalized key information for each review.

Table 2 Aspect words and their keywords

Aspect	Keywords
Location	'location,' 'traffic,' 'minute,' 'walk,' 'property,' 'noise,' 'loud'
Service	'server,' 'service,' 'welcome,' 'staff,' 'management,' 'friendly,' 'front desk,' 'helpful,' 'help,' 'courteous'
Room	'room,' 'bed,' 'clean,' 'cleanliness,' 'dirty,' 'bathroom,' 'rooms,' 'suite,' 'décor,' 'spacious,' 'suite'
Value	'value,' 'price,' 'quality,' 'worth,' 'rate,' 'cost,' 'luxury,' 'cheap,' 'bargain,' 'affordable,' 'money,' 'overpriced,' 'expensive,' 'budget'
Facility	'pool,' 'parking,' 'internet,' 'wifi,' 'vibe,' 'facilities,' 'amenities,' 'hotel,' 'fitness,' 'gym'
Food	'delicious,' 'breakfast,' 'coffee,' 'restaurant,' 'eatery,' 'food,' 'restaurants,' 'bar'

5.2 Personalized Key Information Guided Network

The architecture we use here is similar to the one used in guiding generation for abstractive text summarization based on key information guide network [20]. The difference between these models is that while the KIGN used the whole keywords as a guided mechanism, our model used only the personalized key information for guiding the network.

The traditional encoder–decoder model works with source text as input and summary is the output. In this method, the summarization method is hard to control because of the lack of a guided mechanism. So we propose adding two enchantment to traditional sequence-to-sequence model: personalized attention mechanism and pointer mechanism.

Firstly, with the help of TextRank algorithm, we extract personalized keywords for the reviews. As displayed in Fig. 1, the personalized keywords are passed individually to a bi-LSTM present in personalized key information-guided network, and then, we join the last forward hidden state and backward hidden state as the personalized key information representation I :

$$I = \frac{h_1^{\leftarrow}}{h_n^{\rightarrow}} \quad (1)$$

Personalized Attention Mechanism: Traditional attention mechanism uses decoder state to attain the attention distribution of encoder hidden states which makes it hard to have a guided mechanism. The personalized key information representation I as the input onto to tradition attention models (Eq. 2) and the personalized attention mechanism is shown in Eq. 3

$$e_{ti} = v^t \tanh(W_h h_i + W_s s_t) \quad (2)$$

$$e_{ti} = v^t \tanh(W_h h_i + W_s s_t + W_I I) \quad (3)$$

where W_I is learnable parameter. We will be using e_{ti} to obtain the latest attention distribution and context vector c .

$$\alpha_t^e = \text{softmax}(e_t) \quad (4)$$

$$c_t = \sum_{i=1}^N \alpha_{ti}^e h_i \quad (5)$$

An advantage of our personalized key information network is that it makes sure that more focus is given to the personalized keywords. So, more focus will be given to the personalized aspects and prior knowledge is given to the model.

Pointer mechanism: Some of the keywords might be missing from the target vocabulary, which might result in the summaries losing key information. So, we introduce pointer generation network which is a hybrid network which lets us both copy the words from keywords and also generate the words from target vocabulary. We use the personalized key information I , the context vector c_t and decoder state s_t and uses them to calculate a soft switch p_{key} , which makes a decision on whether to generate words from target vocabulary or reproducing words from the input text:

$$P_{\text{key}} = \sigma(w_I^T I + w_c^T c_t + w_{s_t}^T s_t + b_{\text{key}}) \quad (6)$$

where w_I^T , w_c^T , $w_{s_t}^T$ and b_{key} are learnable parameter, σ the sigmoid function.

Our pointer mechanism, where the personalized key information is included, has the capacity to recognize personalized keywords. The attention distribution is used as the probability of the input word a_i and the probability distribution to predict the successive word was obtained:

$$P(y_t = a) = P_{\text{key}} P_v(t = a) + (1 - P_{\text{key}}) \sum_{i:a_i=a} \alpha_{ti}^e \quad (7)$$

Note that if a is an OOV word, then P_v is zero. The main advantage of pointer generation is the ability to produce OVV words with respect to the personalized keyword.

We reduce the maximum likelihood loss at every individual decoding time step during training, which is mainly used for creating sequences. We define y_b^* as the target word for each decoding time step b and the loss is given as

$$L = \frac{-1}{T} \sum_{t=0}^T \log P(y_b^* | y_1^*, \dots, y_{t-1}, x) \quad (8)$$

5.3 Experiments

All experiments has been conducted on Tripdata which has 134,374 training triplets, 5000 test triplets and 1500 validation triplets. Two bidirectional LSTMs of dimension 256 are used in encoder, and an LSTM of embedding 256 is used for decoder. We use GloVe pretrained model for word embedding with dimension 300. A vocabulary size of 63,172 words is used for both source and target texts. W truncate the review to 450 token, personalized keywords to 30 tokens and the summary to 50 tokens for training. The dropout used [24] with probability $p = 0.2$. During training, we use loss on the validation set to implement early stopping and also apply gradient clipping [25] with range $[-4, 4]$. At test time, we use beam search by setting a beam size of 7

for producing summaries. We use Adam as our optimizing algorithm and by setting the batch size to 128. Our model was trained on 300 training iterations.

5.4 *Evaluation Methods*

We exploit ROUGE [26] metric for evaluating our model. ROUGE scores reported in this paper are computed by Pyrouge package.

5.5 *Comparison Methods*

As far as we know, all previous review summarization studies focused on the multi-review summarization scenario, which is essentially different from our task. Here, we compare with several methods which are popular in abstractive text summarization approaches.

- **S2S** is sequence to sequence model with attention. For this model, an attention mechanism is added to the framework by taking into account context vectors in hidden state of the encoder that helps in decoding the target sequences and achieved state-of-the-art results for DUC-2004 and Gigaword datasets.
- **SEASS** [13] adopts a selective network to select important information from review into S2S + Att. This model uses a selective mechanism to control the information flow which highlights important information and release the burden of the decoder which performed significantly better on ROGUE score for English Gigaword, DUC 2004 and MSRATC test sets.
- **PGN** [12] adopts a copy mechanism to copy words from review when generating summarization into S2. This model which uses pointer generation network, which is a hybrid network, which lets us both copy the words from keywords and also generate the words from target vocabulary. PGN was able to achieve state-of-the-art results on CNN/Daily Mail dataset.
- **User-Aware Sequence Network** [21] is used for personalized review summarization. This model is based on S2S network with user-aware encoder and user-aware decoder where selective mechanism is used in user-aware encoder to highlight user-specific information in the review. USN achieved state-of-the-art rouge for personalized summarization.

Table 3 ROUGE $F1$ scores on the test set for various models

Models	Rogue-1	Rogue-2	Rogue-L
S2S	28.15	14.85	28.88
SEASS	29.17	15.22	29.21
PGN	32.53	19.72	33.63
USN	32.89	20.11	33.48
Our model	34.36	21.51	34.28

6 Results

6.1 Review Summarization

Our results are shown in Table 3. Our model is evaluated with standard ROGUE score, taking the $F1$ scores for ROUGE-1, ROUGE-2 and ROUGE-L. We observe that the S2S model has the lowest score since it employs a basic encoder decoder model for summarization. S2S model. For SEASS, we add the selective mechanism, and the scores improve slightly. However, it is important to filter the review text using a selective mechanism for summarization. Therefore, we use guided mechanism with personalized keywords, and it is seen to be efficient. Our model has a gain of 6.21 ROUGE1, 6.66 ROUGE-2 and 5.4 ROUGE-L on. The PGN model, soft copy mechanism is added to the encoder decoder model, performs better than the previous models. The copy mechanism is incorporated on our model as we observed that copying word from the input text is shown to improve the summarization. Our model outperforms PGN by 1.83 ROUGE1, 1.79 ROUGE-2 and 0.6 ROUGE-L.

We also used USN model where modelling is done using user-related characteristics. This performed better than all the other models. So, we know that incorporating user specific information in our model, and using that as a guidance mechanism will improve summarization. Therefore, we have used the personalized key information guide network in our model and we have achieved 34.36 ROUGE1, 21.51 ROUGE-2 and 34.28 ROUGE-L. Our model has exceeded the baseline models with the implementation of personalized attention mechanism and pointer mechanism over a sequence to sequence model.

6.2 Human Evaluation of Personalization

Personalized key information-guided network is a personalized model which also capture aspect preference of individual reviewer. Important aspects for each user are found out in the preprocessing steps. Therefore, we want to identify if these aspects are present in the summaries generated by our model.

We make use of six aspects already mentioned in preprocessing, and for our use case of hotel review, we add a label to describe the overall attitude towards the hotel.

Table 4 Aspect-level precision, recall and $F1$ for various models

Models	Precision	Recall	$F1$ -score
S2S	0.502	0.501	0.501
SEASS	0.512	0.512	0.512
PGN	0.533	0.542	0.5376
USN	0.565	0.572	0.568
Our model	0.615	0.625	0.619

Then, the generated summaries are labeled with the above-mentioned labels. The generated summaries are labeled as follows:

Example 1: excellent customer service (**service**).

Example 2: Great rooms and perfect location (**rooms, location**).

To perform this human evaluation, 1400 user reviews are randomly sampled from our test set. We produce summaries for the reviews using our personalized key information-guided network model also along with those predictions are done using other models such as S2S, SEASS and PGN. Then, we manually label generated summaries and the user preferences. This is done to see how many of the user's preferences are present in the generated summary. While labeling, we check whether the user labels are present in the review and if not, those labels are removed from user aspects. After all the predicted summaries are labeled, we compute aspect level precision, recall and $F1$ score for different models and shown in Table 4. We observe that our model performs better than the other existing models which shows that our model captures personalized features better. This is because of the presence of personalized attention mechanism and pointer mechanism which helps us capture user aspect preferences.

6.3 Case Study

Figure 3 shows an example of review between our model and USN. The output from both the models are compared with the gold summary. The review talks about aspects such as location, room, service and value. But the gold summary is a general text with no aspects. Here, it is observed that our model was able to introduce aspects into the summary and make it more meaningful. Both USN and our model were able to include aspects in our summary. We identified three user-specific aspects namely *service*, *room* and *value* from previous reviews. But it seen that USN covers two aspects, *location* and *service*, of which one, *location*, is not mentioned in user aspects. For our model, it is seen that the summary included information from the three user-specific aspects such as *service*, *room* and *value*. Therefore, our model was able to summarize the review, taking it account the user's preferences.

Review:	I read some of the past reviews and was hesitant to stay here. I usually stay close to the Seattle Center on 5th and Roy. This location is very comfortable for me to travel. The rooms, which are pretty dated, had a modern feel to them. There was a nice sofa bed in the room. The Heavenly bed was indeed nice. The internet was easy to use, but there is a charge for the service. The charges were too high for me. Housekeeping service was great; the attendant was very courteous and professional. Front desk personnel was friendly and cordial. And I did receive a phone call a few hours after I checked in to see if all was in order. I did use the business center and the gentleman running the office was attentive. Room service was surprisingly fast and efficient. Since it was pretty cold to go out at 10pm on Saturday night, I was very impressed with the efficiency and quality of service with the room service. Food was not bad.
Gold summary :	scared of the reviews enjoyed the stay
User aspect :	'service', 'room', 'value'
USN :	good location and great room service
Our Model :	nice room , bad value , great friendly service

Fig. 3 Comparison of the output between the two personalized models on a hotel review are given. Actual summary given by the user is shown as **Gold** summary and the user's preferences are given in **User aspect**

7 Conclusion

In this project, we address personalized review summarization and propose a Personalized Key Information Guided Network to account for user aspect preference into personalized review summarization. At first, we use extractive methods to get personalized keywords as additional input for reviews which accounts for the user preference. Secondly, the important feature of this model is the personalized key information-guided network which helps include the personalized features in the summarization used along with the pointer generation network. To validate our model, we have created a Tripdata dataset containing of hotel reviews from the TripAdvisor website. From experimentation, our model was seen to be performing better than existing and traditional models for the case of personalized summarization.

As future enhancement to the existing model, we can introduce a transformer model with a guided network which would help in summarization of long texts and also introduce parallelize the operations. In addition to that, instead of setting predefined words to identify each aspect, we could introduce a network which identify aspect words from their semantic meaning. This would help in generalizing the model across many domains not just the hotel review dataset.

References

1. M.-T. Luong, H. Pham, C.D. Manning, Effective approaches to attention-based neural machine translation. arXiv preprint [arXiv:1508.04025](https://arxiv.org/abs/1508.04025) (2015)
2. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in *Advances in Neural Information Processing Systems (NIPS)*, Dec

- 2017, pp. 5999–6009
3. S. Chopra, M. Auli, A.M. Rush, Abstractive sentence summarization with attentive recurrent neural networks, in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (Association for Computational Linguistics, 2016), pp. 93–98
 4. R. M6ro, M. Bielikov6, Personalized text summarization based on important terms identification, in *23rd International Workshop on Database and Expert Systems Applications* (IEEE, 2012), pp. 1529–1588
 5. R. Mihalcea, P. Tarau, Textrank: bringing order into texts, in *Proceedings of EMNLP 2004* (Association for Computational Linguistics, Barcelona, 2004), pp. 404–411
 6. D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate. arXiv preprint [arXiv:1409.0473](https://arxiv.org/abs/1409.0473) (2014)
 7. A.M. Rush, S. Chopra, J. Weston, A neural attention model for abstractive sentence summarization. arXiv preprint [arXiv:1509.00685](https://arxiv.org/abs/1509.00685) (2015)
 8. S. Takase, J. Suzuki, N. Okazaki, T. Hirao, M. Nagata, Neural headline generation on abstract meaning representation, in *Empirical Methods in Natural Language Processing* (2016)
 9. R. Nallapati, B. Zhou, C. dos Santos, . G6leħre, B. Xiang, Abstractive text summarization using sequence-to-sequence RNNs and beyond, in *Computational Natural Language Learning* (2016)
 10. Y. Miao, P. Blunsom, Language as a latent variable: discrete generative models for sentence compression, in *Empirical Methods in Natural Language Processing* (2016)
 11. L. Yu, J. Buys, P. Blunsom, Online segment to segment neural transduction, in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing* (Association for Computational Linguistics, Austin, TX, 2016), pp. 1307–1316
 12. A. See, P.J. Liu, C.D. Manning, Get to the point: summarization with pointer-generator networks. arXiv preprint [arXiv:1704.04368](https://arxiv.org/abs/1704.04368) (2017)
 13. Q. Zhou, N. Yang, F. Wei, M. Zhou, Selective encoding for abstractive sentence summarization, in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics* (Volume 1: Long Papers) (2017), pp. 1095–1104
 14. K. Yao, L. Zhang, D. Du, T. Luo, L. Tao, Y. Wu, Dual encoding for abstractive text summarization. *IEEE Trans. Cybern.* (2018), pp. 1–12
 15. W. Lu, L. Wang, Neural network-based abstract generation for opinions and arguments, in *NAACL* (2016), pp. 47–57
 16. D. Krishnan, P. Bharathy, Anagha, M. Venugopalan, A supervised approach for extractive text summarization using minimal robust features, in *2019 International Conference on Intelligent Computing and Control Systems (ICCS)* (2019), pp. 521–527. <https://doi.org/10.1109/ICCS45141.2019.9065651>
 17. K. Shalini, H.B. Ganesh, M.A. Kumar, K. Soman, Sentiment analysis for codemixed Indian social media text with distributed representation, in *2018 International Conference on Advances in Computing, Communications and Informatics, ICACCI* (IEEE, 2018), pp. 1126–1131
 18. N. Lalithamani, R. Sukumaran, K. Alagamnai, K.K. Sowmya, V. Divyalakshmi, S. Shanmugapriya, A mixed-initiative approach for summarizing discussions coupled with sentimental analysis, in *Proceedings of the 2014 International Conference on Interdisciplinary Advances in Applied Computing* (ACM, 2014), p. 5
 19. M.V.K. Kiran, R.E. Vinodhini, R. Archanaa, K. Vimalkumar, User specific product recommendation and rating system by performing sentiment analysis on product reviews, in *2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS)* (2017), pp. 1–5. <https://doi.org/10.1109/ICACCS.2017.8014640>
 20. C. Li, W. Xu, S. Li, S. Gao, Guiding generation for abstractive text summarization based on key information guide network, in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, vol. 2 (2018), pp. 55–60
 21. J. Li, H. Li, C. Zong, Towards personalized review summarization via user-aware sequence network, in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33 (2019), pp. 6690–6697

22. J. Li, M. Ott, C. Cardie, Identifying manipulated offerings on review portals, in *EMNLP* (2013)
23. H. Wang, Y. Lu, C. Zhai, Latent aspect rating analysis on review text data: a rating regression approach, in *SIGKDD* (2010), pp. 783–792
24. N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **15**(1), 1929–1958 (2014)
25. R. Pascanu, T. Mikolov, Y. Bengio, On the difficulty of training recurrent neural networks, in *ICML* (2013), pp. 1310–1318
26. C.-Y. Lin, Rouge: a package for automatic evaluation of summaries, in *Text Summarization Branches Out* (2004)

PKI-Based Security Enhancement for IoT in 5G Networks



Nayeem Ahmad Khan

Abstract The Internet of Things (IoT) is a concept that includes physical devices, web-enabled devices, and the entire network of connections that they use to communicate. The IoT enables these objects to be sensed and controlled remotely across existing network infrastructure, creating opportunities for more direct integration between the physical world and computer-based systems. Nowadays, 5G (fifth generation) systems have the potential to make the IoT concept becomes reality. However, it is difficult for the 5G network against eavesdropping due to the characteristics of 5G networks. In this paper, we have proposed the ElGamal Cryptography with Public Key Infrastructure (PKI) techniques on the communication from the Base Station (BS) to Relay Station (RS) and RS to RS, then finally from RS to Subscriber Station (SS) to against replay attacks, man-in-the-middle (MITM) attack and denial-of-service (DoS). Through the discussion on the proposed mechanism and it has the probability to ensure the confidentiality, integrity, availability, and non-repudiation of the transmitted data. We have discussed the performance analysis of the proposed mechanism and given a conclusion on the discussion. From the result, it shows the mechanism can enhance the security level on the 5G networks.

Keywords Cybersecurity · IoT · 5G · Cryptography · ElGamal · PKI

1 Introduction

With the arising of the Internet of Things (IoT), which is a concept used to define an autonomous communication within a group of physical objects in a system [1], a giant network of objects which has their unique IP address can now be connected on the internet to automate a simple task more productively. To put it into words, IoT is the concept of connecting any devices that have an on and off switch to the Internet [2]. In the giant network of IoT, the relationship between the connected objects (people with people, people with devices, and devices with devices) opens

N. A. Khan (✉)

Faculty of Computer Science and Information Technology, AlBaha University, AlBaha, Saudi Arabia

e-mail: nayeem@bu.edu.sa

potential value for these objects to work together with better efficiency and accurate results.

To connect all the objects in a network, the incoming fifth-generation 5G network is a promising technology in fulfilling the concept of IoT. The 5G cellular network is expected to have a higher system capacity, higher data rates, massive device connectivity, reduced latency, and robustness against unexpected conditions [3]. In the meantime when a node (base station) with a lower power has to communicate with the sink node (subscriber station), the node will have to travel across multi-hop which then requires neighbor's nodes to be used as relays.

With all the IoT data that is transferred through the multi-hop cellular network, the issue in securing the privacy of the data has been a major concern. There is a possibility where the relay nodes may be eavesdropped by an eavesdropper. The eavesdropper may use the information transferred from the node (base station) to other unauthorized nodes.

Before asymmetric-key cryptography, namely Public Key Infrastructure (PKI) is being introduced; the traditional cryptography method which also known as symmetric-key cryptography is commonly used in securing the transmission of data in a network. The symmetric-key cryptography is performed in which the same secret key is used between the nodes (base station) and relay nodes and sink node (subscriber station). As the distribution of secret keys requires complex protocols and architecture, therefore it is difficult for the cryptographic method to be implemented in IoT [4–6], and this leads to possible eavesdropping attacks.

To secure the eavesdropping of data in a multi-hop relay network operation, the ElGamal algorithm with Public Key Infrastructure (PKI) technique is proposed in this paper to secure the data from being attacked. PKI technique involves the use of both public key and private key, which a public key is known by anyone in the network to encrypt the message and to verify signatures while a private key is only known by the sink node (subscriber station) is used to decrypt the message and create signatures. Meanwhile, the ElGamal algorithm which is specifically based on the Diffie-Hellman key exchange algorithm uses a one-way function in which encryption and decryption are done in separate functions. The use of the PKI technique and ElGamal algorithm proposed in this paper helps to prevent security issues such as DoS attack, MITM attack, replay attack, sniffing attack, and spoofing attack to achieve integrity, confidentiality, non-repudiation, and availability in message transmission.

From the network diagram in Fig. 1, BS transmits data to SS using 5G wireless network with the presence of an ad-hoc network. And to enhance the coverage area and performance of the transmission data, multiple numbers of relay stations are used as illustrated in the diagram. This scenario is known as a multi-hop relay network.

First of all, the function of a BS is to send and receive the data transferred. In the meantime, the relay station acts like a booster station or an amplifier in restoring the strength of a transmitted signal. A relay can be categorized into two different groups, transparent relay and non-transparent relay [7]. A transparent relay can only work in a centralized scheduling mode, whereas a non-transparent relay can work in both centralized and distributed scheduling mode. To understand better, we first have to know the difference between centralized and distributed scheduling. Centralized

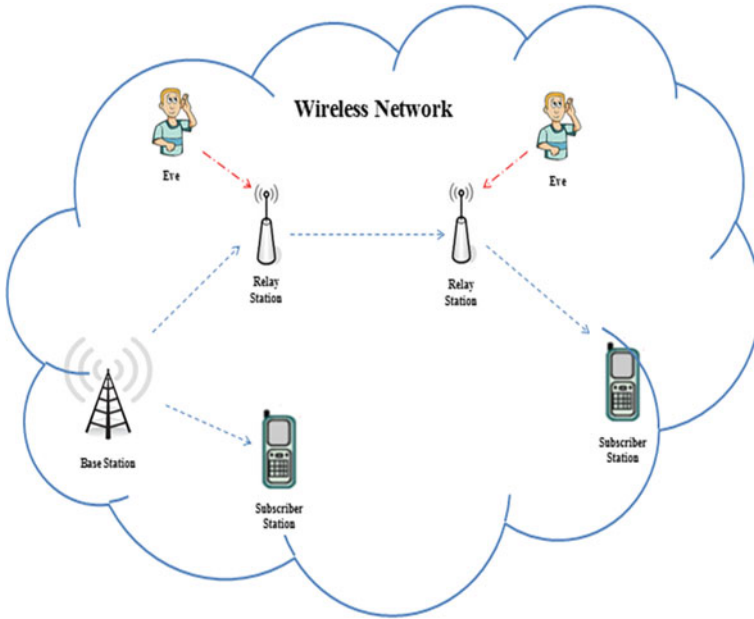


Fig. 1 Wireless network area with multi-hop

scheduling means there is one focus of control. Meanwhile, distributed scheduling distributed their control to other different parts. In this paper, we focus on non-transparent distributed scheduling where the RSS as in Fig. 1 can operates as a BS and transmits the broadcasted message and relays data. Besides, the function of the relays can be categorized into three different strategies which are to estimate and forward, amplify and forward and decode and forward.

The paper is organized as; Sect. 2 studies the wireless network with multi-hop; Sect. 3 presents the suggested solutions using the combination of ElGamal cryptography with PKI. Then, Sect. 4 presents the performance analysis of the proposed cryptography; Sect. 5 is the last section presented in this paper with a conclusion made.

2 Basis Model

As in Fig. 1, there are two RS between BS and SS which according to the authors [8], unreliability increases when the number of hops increases in a distributed and non-transparent scheduling mode. In other words, when the unreliability in an environment increases, the number of attacks such as eavesdropping and tapping eventually increases too. This may lead to attacks like denial-of-service (DoS), man-in-the-middle (MITM), replay, sniffing, and spoofing.

In a wireless network, there is a shared intermediate when transferring a message. This intermediate has caused an easily attack from DoS attacks [8]. DoS attack may occur during the transmission of data from BS to SS where an attacker might try to intercept the data transferred. DoS attack can disconnect the transmission of data or conversation among BS and SS as long as the conversation is available. Besides, MITM occur when an attacker tries to intercept the transferred message by involving himself in the conversation. A typical MITM occurs when the attacker has gained full control over the conversation.

Besides, a replay attack occurs when there is a continuous attack on the conversation. A replay attack will intercept and interrupt the conversation in which the attacker will modify the content of the conversation and send it to the other unauthorized party. The success of the replay attack leads to MITM. In contrast, a DoS attack occurs when an attacker fails in performing a replay attack. A wireless sniffing attack is to use a hardware or software to intercept the transmitted message and decrypt it into a readable format. The attacker will capture the transmitted data without encrypting it like username and password. Spoofing attacks can perform an attack on a wireless network due to the open nature of the wireless medium. Most of the time, an attacker tries to attack the media access control (MAC) addresses by pretending as a legitimate BS.

Therefore, to ensure confidentiality, integrity, availability, and non-repudiation of the transmitted data, the security level on a network should be increased. Securing the transmission of data over a wireless network is not an easy task due to the high number of attacks in the insecure channel. The proposed method in this paper to solve the attacks over the wireless network is by using ElGamal Cryptography with PKI.

3 Proposed Work

3.1 *ElGamal Cryptography with PKI*

Based on the scenario in the network diagram (Fig. 1), data will be continuous transmitted from the BS to the SS through the relay stations where the relay stations can boost the speed of transmission. In this part, we will focus on the way to overcome the attacks in a wireless network with the techniques of ElGamal Cryptography with PKI. ElGamal Cryptography is asymmetric cryptography. It can protect the confidentiality of the transferred data. However, the integrity, availability, and non-repudiation of the data are not able to ensure through the ElGamal Cryptography. In contrast, PKI can protect the confidentiality, integrity, availability, and non-repudiation of the content of the data. PKI only can be used for short messages. Therefore, the exchange of the keys between two parties is needed due to send a large volume of the message. In the integration of ElGamal Cryptography and PKI, the confidentiality of the content can be strengthening. This is because both of the algorithms include the protection

of confidentiality. However, if the information is compromised to an unauthorized party, the content of the data will be no more confidentiality.

In ElGamal Cryptography, it involved the use of a discrete algorithm to get the public key and the exchange of public keys between BS and SS. The details of the process in ElGamal Cryptography with PKI are shown in Fig. 2. First, the BS and SS must agree on picking the prime, “ p ” and generator, “ g .” These two numbers are mainly used in the discrete algorithm process. Without these two numbers, the encryption and decryption between BS and SS cannot be achieved. Next, the SS chooses the private key and keeps within itself. After the private key has been decided, an equation to form the public key is used.

$$PK_B = (g^b) * (\text{mod } p) \tag{1}$$

where

PK_B : Public Key of SS/RS

g^b : generator with Private Key of SS/RR, b

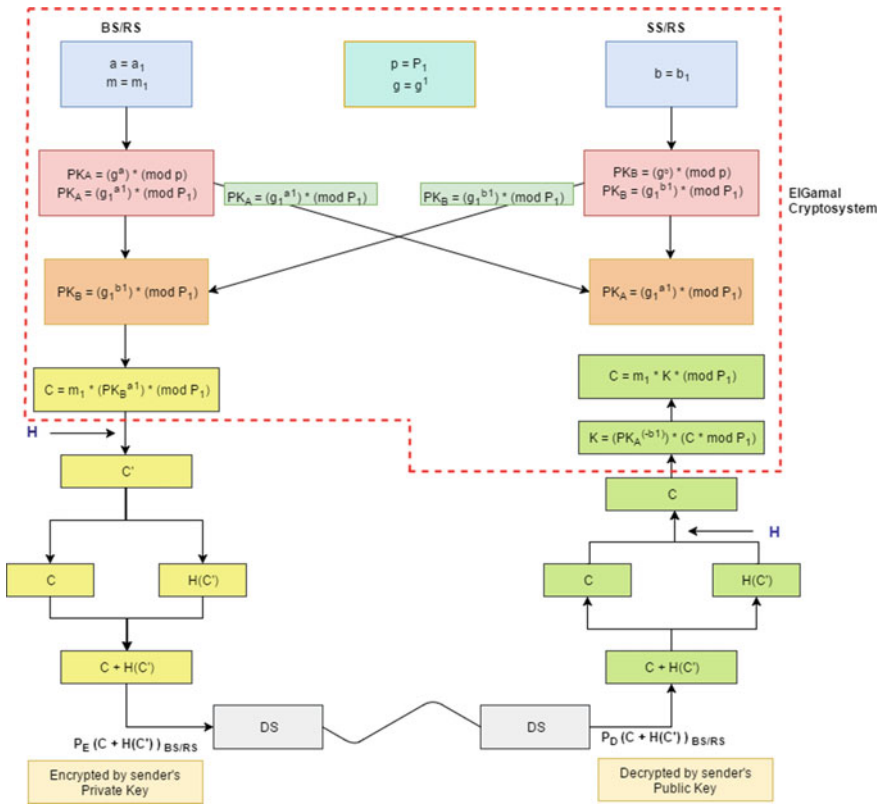


Fig. 2 The process of ElGamal cryptography with PKI

p : Random prime number.

From the above equation, the “ b ” is the chosen private key of SS. By using multiplication, the public key of SS/RS can be known.

At the BS side, the BS needs to choose a random number, “ a ” as BS’s private key and give the message that sends to SS with an annotation as “ m .” The “ m ” includes the content of the message to be sent. The equation to get the public key is the same as the equation in SS but the private key is different as they are using their private key.

$$PK_A = (g^a) * (\text{mod } p) \tag{2}$$

where

PK_A : Public Key of BS/RS

g^a : generator with Private Key of BS/RS, a

p : Random prime number.

From the above equation, the “ a ” is the chosen private key of BS. After the insertion of “ a ” into the equation, the public key of BS is known and exchange the public key, PK_A , to SS. The public keys, PK_A and PK_B will be exchanged between two parties.

The following step is that the BS will use the PK_B to multiply with the message, “ m ,” with the content, “ m_1 .” The “ m_1 ” which is the message will be sent to SS.

$$C = m_1 * (PK_B a_1) * (\text{mod } P_1) \tag{3}$$

where

C : ciphertext of the encrypted message

m_1 : the content of the message to be sent

PK_B : Public Key of SS/RS

a_1 : Private Key of BS/RS

P_1 : Random prime number of p .

The multiplication of these values is to encrypt the message into ciphertext, “ C .” The encryption in the use of a discrete algorithm is to prevent the attacks of the intruder. Without the private keys of both parties, the intruder has a lower probability in eavesdropping.

The concept of PKI will be applied to the following steps. The hash function is used on the C value in order the ciphertext, “ C ,” has the second time encryption. The second encryption of the ciphertext uses the “ C ” to represent in the equation. After that, the first time ciphertext, “ C ” is added with the second time ciphertext with the hash function, “ $H(C')$,” to get the third encryption. The encryption uses the private key of BS/RS to encrypt with the ciphertext.

$$P_E(C + H(C'))_{BS/RS} \tag{4}$$

where

E : Encrypted by BS/RS’s Private Key

C : Cipher text

$H(C')$: Ciphertext with the hash function.

The encrypted message is sent through the insecure channel. When the encrypted message is sent through the channel, the digital signature (DS) of the BS/RS has been used as the authentication in the communication. Therefore, the SS/RS needs to request the digital signature of the BS/RS to verify the message. If the requested digital signature is not able to authenticate, then SS/RS has the right to reject the message to read. Otherwise, the SS/RS can proceed to the decryption process.

After the digital signature is being verified, the encrypted message will be decrypted by using Eq. 5.

$$P_D(C + H(C'))_{BS/RS} \quad (5)$$

where

P_D : Decrypted with Public Key of BS/RS

C : Cipher text

$H(C')$: Cipher text with hash function.

In the decryption process, both the ciphertext and the ciphertext with the hash function will be decrypted with the public key of BS/RS. To eliminate the hash function on the ciphertext, the second time of the hash function will be used. So that the ciphertext only with the first encrypted message. To continue with the decryption, the decryption factor, “ K ,” needs to be found. The equation below shows the calculation to get the decryption factor, “ K .”

$$K = (PK_A^{-b_1}) * (C * \text{mod } P_1) \quad (6)$$

where

K : Decryption factor

PK_A : Public Key of BS/RS

b_1 : Private Key of SS/RS

C : Cipher text

P_1 : Random number of p .

The decryption factor, “ K ,” is important in getting the decrypted message. This is because the “ K ” needs to insert in the equation below to get the decrypted content of the message.

$$C = m_1 * K * (\text{mod } P_1) \quad (7)$$

where

C : Cipher text

m_1 : Content of message

K : Decryption factor

P_1 : Random number of p .

The final output of the discrete algorithm is the content of the message, " m_1 ." When inserting the " K " value into the equation below, the arrangement needs to be changed so that the " m_1 " can be decrypted.

4 Performance Analysis

The main purpose of using the combination of ElGamal Cryptography with PKI is to prevent man-in-the-middle (MITM) attack, DoS, and replay attack. According to the proposed work, with the use of prime p , generator g , random number a , and private keys from both parties, the encryption process that results from the exchange of public keys can ensure the components of data from being interrupted and intercepted.

The chances of an attacker to perform MITM attack, DoS, and replay attack are high when data is transmitted through an insecure channel, intruder can intercept or interrupt the content of the message through these attacks. A replay attack causes the loss of integrity of the message where the authentication of both parties can be stolen. However, the proposed work with the use of a discrete algorithm in the encryption process is likely to prevent the data from being violated. Besides, as the use of private keys on both parties is kept as a secret, thus it helps in securing the message. With the increased level in getting the private key, the attacker will not be able to decrypt the data. Thus, the accuracy of the transmitted information without any falsification can be guaranteed.

Replay attack leads to MITM attack and DoS. In 5G networks, when a MITM attack is been executed, the attacker will be able to manipulate the information transmitted between the authorized party. The attacker gains control of the conversation by impersonating as a legitimate sender to communicate with the receiver. As a result, the sender might be giving in some confidential data that cause the loss of confidentiality and integrity. However, the proposed work provides a higher level of security if compared with the use of PKI or ElGamal Cryptography only. With the use of a discrete algorithm and digital signature in the encryption and decryption processes, the data are only limited for authorized users only. The attacker may fail in performing a MITM attack as a complicated discrete algorithm and authentication between two parties makes the intruder take more time in intercepting the content of the message. Therefore, confidentiality can be ensured as confidential data are only now limited for the intended use.

DoS attacks on the network are performed by flooding a network with traffic. As the network is now full of overwhelming resources, hence it is difficult for the user to access the network which in turn makes the communication becomes unavailable. DoS attack can be prevented with the implementation of the proposed work where when an attacker tries to overflow the network with traffic, the receiver can use a digital signature to identify the legitimate user and the communication can be stopped immediately. Besides that, the public keys of both parties have been encrypted into a numeric message when passing through the insecure channel. It is hard for an intruder to get the information through the encrypted message due to the complicated discrete

algorithm used in the process. For that reason, the availability can be fulfilled whereby the legitimate user can access the network anywhere anytime.

5 Conclusion

This paper addressed a combination of ElGamal Cryptography and PKI for the multi-hop communication from BS to SS through two relay stations in the 5G network. This mechanism is suitable to prevent attacks like MITM attack, DoS, and replay attacks to ensure the confidentiality, integrity, availability, and non-repudiation of the transmitted message. This mechanism provides the complicated discrete algorithm and digital signature in the encryption and decryption processes. However, the proposed mechanism may not have a full guarantee in ensuring the confidentiality, integrity, availability, and non-repudiation of the transmitted message on any networks. Nowadays, the attack techniques have been increasing day by day due to advanced technology. The proposed mechanism has the potential to reduce the chances of attacks from intruders for the multi-hop communication within 5G networks.

References

1. A. Mohammed, Special issue on internet of things: smart things network and communication (2016). Available at <https://www.journals.elsevier.com/journal-of-network-and-computer-applications/call-for-papers/internet-of-things-smart-things-network-and-communication>
2. J. Morgan, A simple explanation of the internet of things, May 2014. Available at <http://www.forbes.com/sites/jacobmorgan/2014/05/13/simple-explanation-internet-things-that-anyone-can-understand/#738c4f236828>
3. T. Nakamura, A. Benjebbour, Y. Kishiyama, S. Suyama, T. Imai, 5G radio access: requirement, concept and technologies. *IEICE Trans. Commun.* **E98-B**(8) (2014)
4. N. Yang, L. Wang, G. Geraci, M. ElKashlan, J. Yuan, M. Di Renzo, Safeguarding 5G wireless communication networks using physical layer security. *IEEE Commun. Mag.* **53**(4), 20–27 (2015)
5. A. Mukherjee, Physical-layer security in the internet of things: sensing and communication confidentiality under resource constraints. *Proc. IEEE* **103**(10), 1747–1761 (2015)
6. Y. Zhang, Y. Shen, H. Wang, J. Yong, X. Jiang, On secure wireless communications for IoT under eavesdropper collusion. *IEEE Trans. Autom. Sci. Eng.* **PP**(99), 1–13 (2015)
7. A.S. Khan, N. Faisal, Z.A. Bakar, N. Salawu, W. Maqbool, R. Ullah, H. Safdar, Secure authentication and key management protocols for mobile multihop WiMAX networks. *Indian J. Sci. Technol.* **7**(3), 282–295 (2014)
8. M. Hangargi, Business need for security: denial of service attacks in wireless networks. Masters dissertation, North Eastern University, Department of Information Assurance, 2012

Wearable Tag for Human Health Monitoring System



A. Jhansi Sri Latha, Ch. NagaSai Manojna, Ch. N. L. Padma Ashalesha,
and K. S. Balamurugan

Abstract With the growing elderly population and the importance of a seamless infant care system, wearable tags are vital for continually monitoring health conditions, researching behaviours, detecting events such as falls, tracking location, and so on. With the advancement of wearable technology, several researchers are developing a solution for establishing a seamless human health monitoring system that can be used both indoors and outdoors. The proposed method has a Convolutional Neural Network (CNN) algorithm for recognizing the human biological signals like Temperature (T), Blood Pressure (BP), ECG (S) and Oxygen level (O) and tracking the location (L) of person. Also, the sensed data gets stored in the cloud for analysing the historical data and predict a future event. is the proposed method has obtained a 99.8% accuracy, less complex and compatible to use over other IoT wearable devices and provide a complete report when the authorized person intends to know the status of health and all by using a mobile phone. The proposed Wearable Tag is required to overcome the impact of present COVID-19 scenario.

Keywords Wearable IoT devices · Sensors · CNN algorithm · Machine learning

1 Introduction

Chronic diseases are the most stranded reason for the unprecedented increase in the death rate. In the current situation, people are being checked using wearable

A. Jhansi Sri Latha · Ch. NagaSai Manojna · Ch. N. L. Padma Ashalesha ·
K. S. Balamurugan (✉)

Department of Electronics and Communication Engineering, Sasi Institute of Technology and Engineering, Tadepalligudem, Andhra Pradesh 534101, India

A. Jhansi Sri Latha
e-mail: srilatha407@sasi.ac.in

Ch. NagaSai Manojna
e-mail: manojna426@sasi.ac.in

Ch. N. L. Padma Ashalesha
e-mail: padma437@sasi.ac.in

devices to detect chronic illnesses such as blood pressure, body temperature, ECG, and oxygen level in future. As time progressed, several chronic disease identification frameworks were developed. Previously established systems are divided into two categories: surveillance-based chronic disease detection systems and wearable sensor-based systems. In the Surveillance-based chronic disease detection system, vibration detection sensors, infrared sensors, depth cameras, cameras, Fibre optic sensors, range-Doppler radar, acoustic sensors, smart tiles, and other technologies have been employed. These gadgets are for the most part situated in a pre-assembled region to screen the old minutes. In a room, cameras and radars are fixed in a particular area and due to some of their fixed characterization; these can screen explicit foreordained zones. Checking individuals, the reconnaissance-based frameworks are agreeable; however, the frameworks are primarily dependent on camera that doesn't cover the client's mystery since it is disallowed in private areas like washrooms, latrines and so forth. On opposite side, wearable sensor frameworks utilize few sensors, for example: pulse sensor, pressure sensor, gyroscope, and tri-axial accelerometer and so on. Attributed to the improvements in miniature electromechanical frameworks, various sensors like pulse sensor, gyroscope and pressure sensors, accelerometers have grown and easily joined into inserted frameworks. A few ongoing chronic diseases identification calculations depend on cell phones that have been projected in current years. The most utilized calculations in chronic diseases recognition incorporate Machine Learning calculations like Decision Trees, Random Forest, Convolutional Neural Networks (CNN).

Kumar [1] examined to screen Heart rate, Respiration rate, ECG, internal heart level. Sensors are associated with microcontroller PIC16F887A. For checking reason made a portable application, website for checking the human wellbeing status. After collecting the required data from sensors, the data is transferred to site physically. Jasses [2] talked about on the temperature of the body assessment utilizing Raspberry pi in distributed computing. In this paper, Raspberry pi monitors the temperature of the body and further communicates the detected information by utilizing remote biomedical sensor innovation, and information established in cloud-based destinations. Dohr [3] screen circulatory strain level using Keep in Touch and shut circle clinical consideration organizations. After reaching the stay in contact, the data is transport off android telephone. A close circle benefit the information obtained from cell phone; by then the information is transport off the secured site. Utilizing this site anyone can screen patient's circulatory strain level. Keep in touch relates to the JAVA-based cell phone with the assistance of close to deal with correspondence and oversees inductive coupling and engaging, from that point the distance is short. Roy [4] screen the ECG surges of patients, AT Mega 16L microcontroller is used for checking ECG levels. This ZigBee module sends data to nearest related structure for ZigBee. The ZigBee module is used for moving ECG levels. Malhi [5] screen internal heat level, heart rate utilizing C8051F020 microcontroller, and ZigBee module is associated with the microcontroller, afterwards that module is move information to closest recipient. Wearable sensors are used to gather information and afterwards ship off

microcontroller. Mansor [6] screen internal heat level utilizing LM35 body temperature sensor. Body Temperature sensor is associated with microcontroller Arduino uno. Utilizing this site anyone can screen body temperature in login measure.

Mohammed [7] screen patient's ECG level at anyplace on the planet utilizing IOIO-OTG Microcontroller. After gathering information, the wave is ship off android application, Android app is made for ECG screening and IOIO-OTG microcontroller is associated with android telephone utilizing Bluetooth dongle or USB link. After gathering information, the level is ship off android app. Screen and it store ECG levels in that android-based app. Jain [8] screen Blood Pressure, Body Temperature, Pulse rate of patients and GSM module is associated to the microcontroller. AT mega 32 microcontroller is utilized to associate the sensors and Subsequent to gathering information, if the worth is low SMS is ship off the specialist. Piyare [9] carry out controlling, observing home machines utilizing android for advanced cell Arduino uno microcontroller, is associated with our home appliances like light's, electrical fans, and so on. Making an android application for this shrewd home Arduino uno board and android app is associated with web and utilizing the android app controlling and checking home machines any place on the planet. We note that while considerable research has been done to find out the health profile of an individual is not much significant has been discovered in existing ML algorithms. Ramachandran [10] applied diverse AI calculations to a public human health discovery dataset. A blend of imperative sign and IMU sensors that are intended to be joined into a wrist-band likewise depicts the information produced. Such wrist groups will be worn in care homes for old grown-ups, where the edge gadget is proposed to be introduced. Singh [11] the human health identification framework is classified into picture, acoustic, PIR, radar, cell phone-based, close to handle imaging, ultrasonic dependent on the ideas of sensor advances. Sensor innovation for various applications was summed up for choosing the appropriate innovation by the analysts and the makers. Chander et al. [12] summed up the difficulties of the current WSS plan, advancement and testing a human wellbeing wearable gadget with a delicate mechanical stretch (SRS) sensor for human wellbeing recognition. The SRS sensor-based series of Parts from I to V articles in "Shutting the Wearable Gap" examine were investigated. Wang et al. [13] proposed multi-class human wellbeing acknowledgement framework. Multi-source CNN Ensemble (MCNNE) structure is utilized for getting the element from various detected information. Information from N number of sensors are pre-handled and arranged as a preparation informational collection independently, and yield highlights map from N number of sensors are joined to incite a total component map. Liu [14] proposed plot enjoys two significant benefits. First and foremost, the proposed framework employments temperature investigation and essential work force situating to distinguish strange fall focuses. Investigating the dubious focuses not just identifies the human wellbeing state in legitimate time, yet furthermore limits the investigation of audacious information. Besides, the layered information handling and the further developed techniques in the extraction period of the component upgrade the exactness. Irregular woodland classifier AI calculation is utilized to accomplish better speculation limit and more exact characterization.

Hashim et al. [15] depending on detected information which one produced by the two-accelerometer product set on the patient’s body, the information occasion calculation (DEA)-based low-power wearable human wellbeing discovery framework (WFDS) utilizing ZigBee was recommended for PD patients. Heading drop occasion (DFE) calculation accommodating to recognize the human wellbeing way of the PD was definitely decided [16]. Lora RF module based sensors wireless network architecture explained clearly in [17].

The rest of the paper is structured as follows: Sect. 2 gives a description of the entire health monitoring system. The experimental results of the suggested system are given in Sect. 4. Finally, a concluding remark such as limitations and stability of the system are given in Sect. 5.

2 System Overview

The chronic diseases prediction and anticipation framework design which was proposed is appeared in Fig. 1. It contains of four blocks: Data collection unit has different wearable Sensors like Body Temperature Sensor, Pulse Sensor, Blood Pressure Sensor, ECG Sensor and a multi-thread for execute the pre-prepared model with authentic information and existing date to produce the reports, Control unit for taking decision from cloud information and sensor information, Wi-Fi module for sending SMS to wrote individual and crisis administrations. Accelerometer Module, Wi-Fi Module, GPS Module GSM Module, is the information modules. These modules are utilized in all bleeding edge cell phones, for these reasons, single-board PCs, for example, Raspberry Pi, and NodeMcu, Arduino Uno are utilized and likewise associate effectively with and utilize these modules in like manner. Raspberry Pi has Wi-Fi on-board, NodeMcu based on Wi-Fi module, namely ESP8266. Interface

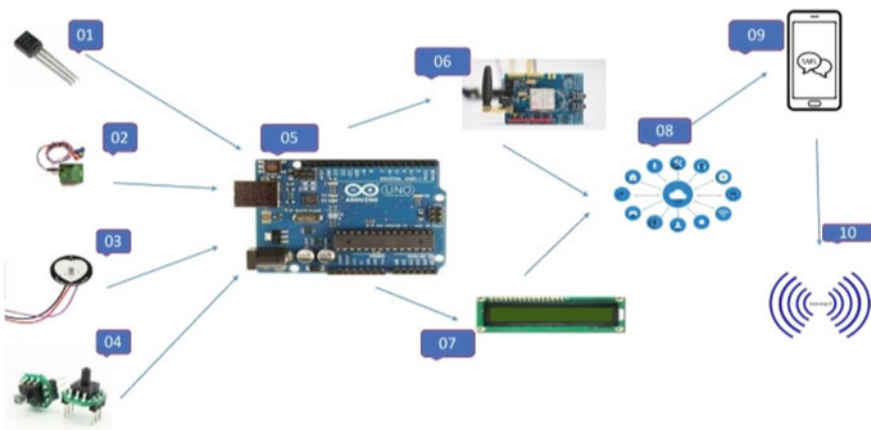


Fig. 1 Overview of a wearable health monitoring system

the analogue sensors with the Raspberry Pi, an extra ADC module is required. The contraption is kept in the pocket on the right or left of the individual. This processor gathers the development information of the client. The information is shipped off the worker for every development. The worker investigations the information and sends back the result of the order to the separate unit. For our plan, we are utilizing heartbeat sensor, blood pressure sensor alongside accelerometer in-built in cell phone for information assortment unit to detect the human information, Bolt IoT regulator and Arduino UNO utilized in regulator unit to take choice either hit the on-going infections counteraction by send SMS to approved individual by Twilio or not from the detecting information and cloud information which one created by Convolutional Neural Network (CNN) and machine learning calculation.

Bolt IoT module: Bolt is a well-integrated IoT platform, for the developers it helps to build projects related to IoT and their products was easily accessible. It is a great platform was designed for developers to make IoT projects. It also quickly accesses the machine learning algorithm to forecast the IoT data of yours and it detects the freak.

Wearable sensors: Wearable sensors are utilized to accumulate physiological and development information subsequently empowering patient status checking and Clinical faculty can distantly screen patient status and be alarmed on the off chance that a clinical choice must be made, Outline of a distant wellbeing checking framework dependent on wearable sensors.

Twilio: Twilio is a cloud interchanges stage that empowers engineers to make universally constructed APIs for voice, SMS, informing applications. Informing, discourse, video, and verification API, Make, get, oversee calls to and from telephone numbers all throughout the planet, send and get SMS, MMS and talk text from any application, and Personalize call streams. The proposed framework sends Twilio's call alert/SMS as opposed to utilizing GSM equipment.

Convolutional Neural Network (CNN): CNN is a series of a pooling layers and convolutional and which allows extracting of the main feature from the images responding to the final object (Fig. 2).

Mathematical equation for convolutional layer:

$$y_{mn} = f \left(\sum_{j=0}^{j-1} \sum_{i=0}^{i-1} x_{m+i,n+j} w_{ij} + b \right). \quad (1)$$

Each neuron in this layer is associated with all neurons in this past layer, and there is no association between neurons in a similar layer. The equation is,

$$y_j^{(l)} = f \left(\sum_{i=1}^n x_i^{(l-1)} \times w_{ji}^{(l)} + b^{(l)} \right), \quad (2)$$

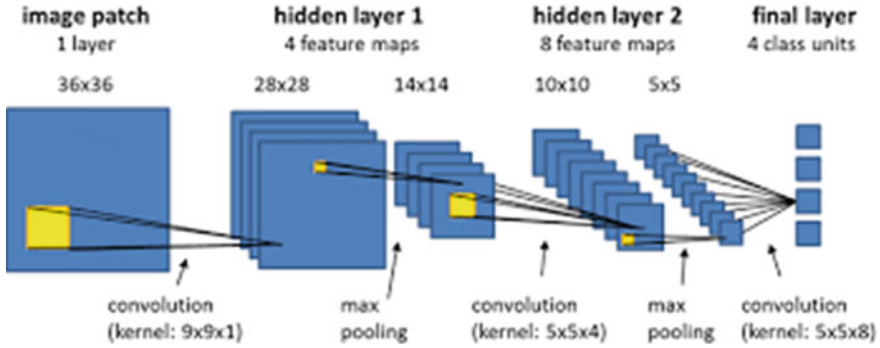


Fig. 2 Architecture of convolutional neural network (CNN)

where n is no. of neurons in the previous layer,

l is the current layer,

$w_{ji}^{(l)}$ connection of neurons weight of j in this layer and neurons i in the previous layer,

$b^{(l)}$ is the bias of neurons j , and f is the activation function.

The yield of the FC layer is produced by a softmax layer which contains of four neurons $[y_1, y_2, y_3, y_4]$, addressing the four classifications. It maps the yield of numerous neurons to the $(0, 1)$ stretch, which can be considered as the likelihood of multi-order. The equation is as per the following:

all actuation capacity of the organizations received the cracked ReLU work:

$$f(x) = \begin{cases} x, & \text{if } x > 0 \\ 0.01, & \text{otherwise} \end{cases} \tag{3}$$

$$z_i^{(l+1)} = w_i^{(l+1)}y^{(l)} + b_i^{(l+1)} \tag{4}$$

$$y_i^{(l+1)} = f(z_i^{(l+1)}) \tag{5}$$

After applying the dropout, forward CNN propagation formula changes to

$$r_j^l \sim \text{Bernouli}(p) \tag{6}$$

$$y^{\sim(l)} = r^{(l)} \cdot y^{(l)} \tag{7}$$

$$z_i^{(l+)} = w_i^{(l+1)}y^{\sim(l)} + b_i^{(l+1)} \tag{8}$$

$$y_i^{(l+1)} = f(z_i^{(l+1)}) \tag{9}$$

any neuron assuming the k th dimension, $\hat{x}(k)$ use the following formula:

$$\hat{w}^{(k)} = \frac{x^{(k)} - E[x^{(k)}]}{\sqrt{\text{var}[x^{(k)}]}} \quad (10)$$

where $x(k)$ is original input data of k th neuron in layer,

$E[x(k)]$ is the mean of the input data in the k th neuron,

and $\sqrt{\text{var}[x(k)]}$ is the standard deviation of the data in the k th neuron.

To re-establish the first information conveyance, change reproduction, and learnable boundaries γ and β are presented in the execution:

$$y^{(k)} = \gamma^{(k)} \hat{w}^{(k)} + \beta^{(k)} \quad (11)$$

where $\gamma^{(k)}$ and $\beta^{(k)}$ are variance and deviation of input data.

The equation of the complete forward CNN normalization process normalized network layer is:

$$\mu = \frac{1}{N} \sum_{i=0}^N X'_i \quad (12)$$

$$\sigma^2 = \frac{1}{N} \sum_{i=1}^N (X'_i - \mu)^2 \quad (13)$$

$$X_i^{\text{norm}} = \frac{X'_i - \mu}{\sqrt{\sigma^2 + \varepsilon}} \quad (14)$$

$$X_i = \gamma X_i^{\text{norm}} + \beta \quad (15)$$

The exhibition of the model on the acknowledgement of four sorts of MI was estimated by accuracy, review and F -score. The bigger the qualities, the better the exhibition of the model. Here, TN: true negatives, TP: true positives, FN: false negatives, FP: false positives.

$$\text{Global average Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (16)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (17)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (18)$$

3 Flow Chart of Proposed Working Model

In the proposed system when the power turns ON, Bolt IoT module sense the Blood Pressure (BP1), pulse rate (P1), temperature (T1), value of accelerometer (A1) from smartphone and send to Bolt Cloud (Fig. 3).

All BP1, T1, P1, and A1 esteems are put away in the foundation of Bolt cloud as a time span signal example and order the edge as blood pressure, temperature, and pulse rate.

CNN calculation foresees the either blood pressure, temperature, heart rate, ECG with assistance of current information outline, authentic information and prepared informational index. In light of worker signal which one produced by CNN calculation and sensor signals, Bolt IoT module produce the trigger flag and convey the alarm message to approved portable number ahead of time if strange conditions anticipated. Following a minute, proposed framework ch sensors esteems are typical or not. On the off chance that typical measure goes to stage one and rehashed. Whenever

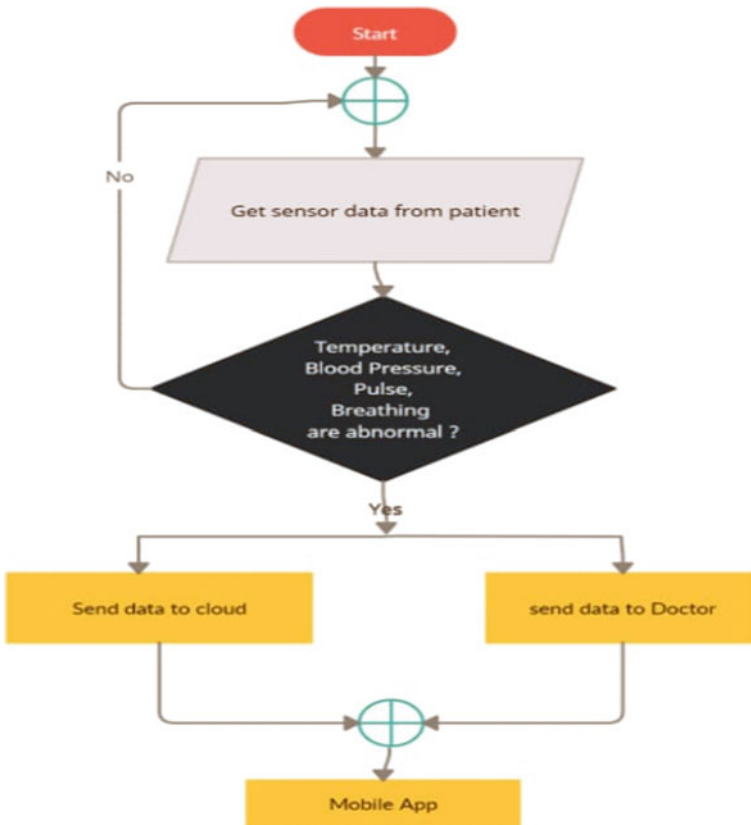


Fig. 3 Proposed idea flow chart

identified invalid worth from sensors and accelerometer sensors then, at that point settle on a decision to family specialist furthermore, rescue vehicle administration.

4 Experiment Result Analysis

4.1 Experiment Setup

One Bolt IoT module, two advanced mobile phones in-built with accelerometer module, two Arduino in-built with Wi-Fi module, one of Blood Pressure Sensor, Body Temperature sensor, Heart Rate Sensor, ECG Sensor were utilized to plan a Wearable IoT Devices for forecast of constant illnesses by Human Movement Monitoring framework. For giving association, JIO M2 Wireless Router was additionally utilized. For producing notice clamours, a functioning signal was picked as it needn't bother with timing circuits or outer, dislike an uninvolved bell. Twilio likewise used to send the SMS caution to the approved telephone numbers. Bolt IoT has ESP8266 module for network, 80 MHz clock frequency, 32-digit RISC CPU, 64 KB guidance RAM; 96 KB data RAM, 4 MB of Flash memory, 1pin 10-cycle ADC, PWM capable pins and auto associated Bolt Cloud arrangement moreover. Arduino Uno ATmega328P+ Wi-Fi R3 ESP8266 4 MB board additionally utilized in our proposed framework alongside beat sensor, temperature sensor and sound sensor for social occasion natural worth in our body, two cell phones a Xiaomi Redmi Note 9 Pro 4 GB RAM, Oppo A9 128 GB and 64 GB Storage were significantly tried for the proposed framework as customer gadgets. These Smartphones has Octa centre processor, installed Wi-Fi and it supports the 802.11 a/b/g/n locally available accelerometer. Python code utilized for gathering and investigating the detected information in customer and carrying out CNN calculation in worker for taking development forecast of illnesses (Fig. 4).

4.2 Result Analysis

The affectability and explicitness are two factual measures to evaluate the presentation of CNN classifiers. The other term used to gauge a classifier model's general presentation is exactness. Affectability is the extent of genuine positives that the classifier has appropriately characterized as sure. Particularity is the level of genuine negatives that the classifier has accurately obvious as negatives. The piece of pertinent events among the recovered events is characterized as exactness or positive prescient worth. Pertinence estimation is portrayed by both Precision and Recall. Exactness can be viewed as a precision or quality measure, though recognition is a culmination or amount measure. In mathematical examination of parallel classifiers, *F1* score's gauge of test exactness. This thinks about both significance of the test's

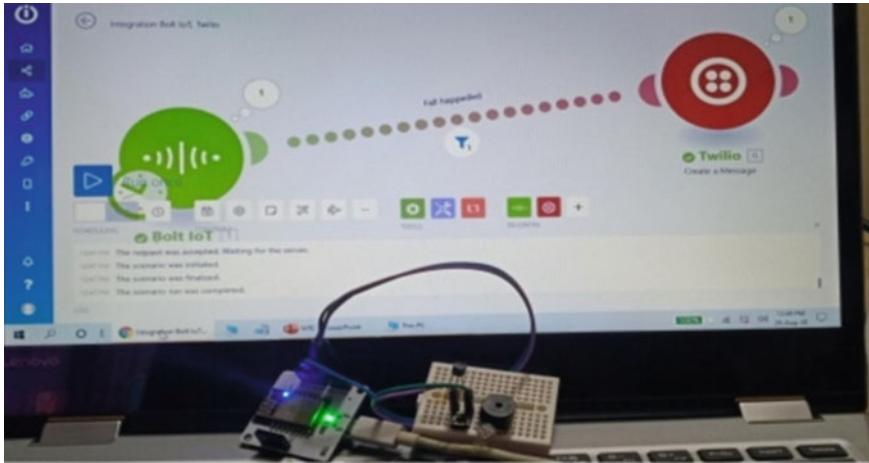


Fig. 4 Hardware and software setup

exactness and review to survey the positioning. The $F1$ score is the symphonies marker of precision and review. The greatest $F1$ rating is 1 and the base rating is 0. A disarray lattice is made to appraise the presentation of the prepared model by True Positives are the on-going illnesses information that the classifier has accurately named “strange conditions.” True Negatives are non-persistent infections information that the classifier has appropriately ordered as non-constant sicknesses. False Positives are non-persistent illnesses information wrongly arranged by the classifier as unusual conditions, while False Negatives are constant infections information erroneously grouped by the classifier as non-on-going sicknesses. A low False Positive and persistent sicknesses Negative rate ought to have a steady AI model. Precision is the measure of right ni results isolated by the quantity of all sure results are returned by classifier, exactness is the quantity of right sure results isolated by the quantity of every fitting example (examples that ought to have been set apart as sure) (Fig. 5).

4.3 Roll of the Client and Server

For customer’s situation, a cycle implies that an absolute time needed to send a message containing information get a worker reaction, and depict the reaction note. The first circle, generally 720 ms, required a lot of time. This second cycle and any remaining progressive cycles required a steady absolute chance to finish around 180 ms. This sum relies upon various factors, for example, network structure, speed of association, climate condition, and so forth considering a worker, a cycle is described as the all-out time taken for a worker string to get a customer’s parse the message, information message predicts the information, and return the information to the customer concerned. All things considered, circumstances, notwithstanding, the bell

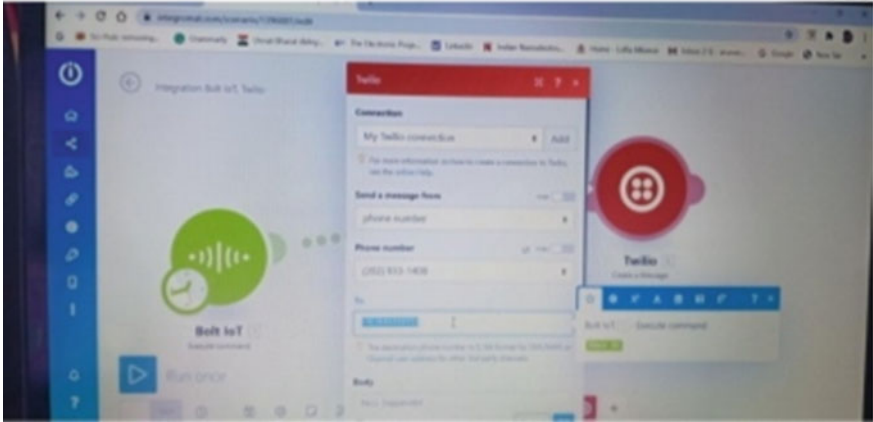


Fig. 5 Output of proposed system

just activates when it detects abnormal occasion. Test yield sent Alert SMS from BOLT IoT to approve telephone number.

CNN comparison with other techniques

The exactness obtained by CNN, ANN and SVM is 99%, 94% and 91%, respectively. Expansion in the preparation tests has improved the exhibition of SVM. All similar AI strategies give exceptionally high arrangement precision and CNN beat the near techniques (Fig. 6).

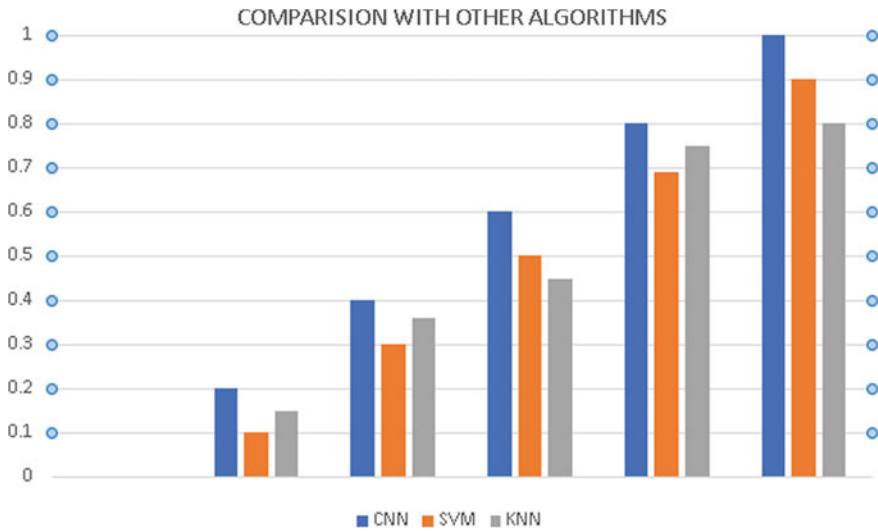


Fig. 6 Bar graph of other algorithms (CNN, SVM, KNN)

5 Conclusion

This paper proposes a Wearable IoT Devices Using Temperature Sensor, Pulse Rate sensor, ECG sensor, Blood Pressure sensor, for detecting chronic diseases in advance by Human Development Monitoring framework with straightforward equipment module like cell phones, Bolt IoT, tri-axial accelerometer, Arduino, GSM, Wi-Fi, and GPS. The Client-worker-based engineering; CNN classifier model-based pre-prepared multi-strung worker has are system. The gadgets interface with the worker's strings and send moving item information to the worker consistently from the accelerometer. The worker assessments whether there have been a normal or not and responds appropriately. The reaction message is gotten by the worker. The worker sends a ready warning containing the area of the customer and other related records to the middle person if an abnormal condition is noticed. The framework produces signal sound, crisis administrations by means of SMS. Accordingly, the controlled individual may get moment clinical guide. In enormous scope settings where on-going observing of a few people is required, for example, clinics or treatment spots and so forth, this framework can be carried out. 99.7% exactness, 99.6% explicitness and 96.3% affectability accomplished utilizing CNN straight classifier model. As the reaction time is extremely short, the created framework is very fast. This guarantees that the worker and customer's regulator know if an abnormal condition occurred inside about 180 ms of sending the detected information to the worker.

References

1. M. Kumar, Human health recognition system based on convolutional neural network algorithm and using wearable sensors. *Sens. Mater.* **31**(4) (2019)
2. Jasses, Human health detection using machine learning methods: a survey. *Int. J. Math. Eng. Manag. Sci.* (2020)
3. Dohr, Highly-efficient fog-based deep learning AAL human health detection system. *Internet Things* **11** (2020)
4. S. Roy, An IoT based device-type invariant human health detection system. *Internet Things* **09** (2020)
5. K. Malhi, A survey on recent advances in wearable human health detection systems. *BioMed Res. Int.* (2020)
6. Mansor, Sensor technologies for human health detection systems: a review. *IEEE Sens.* (2020)
7. J. Mohammed, Research article daily activity monitoring and human health detection based on surface electromyography and plantar pressure. *Complexity* (2020)
8. N.P. Jain, Wearable stretch sensors for human movement monitoring and human health detection in ergonomics. *Int. J. Environ. Res. Public Health* (2020)
9. R. Piyare, Implementation of a real-time human health detection system for elderly Korean farmers using an insole integrated sensing device. *Instrum. Sci. Technol.* (2019)
10. A. Ramachandran, A survey on recent advances in wearable human health detection systems. *BioMed Res. Int.* (2020)
11. A. Singh, Sensor technologies for human health detection systems: a review. *IEEE Sens.* (2020)
12. H. Chander, R.F. Burch, J.E. Ball, Wearable stretch sensors for human movement monitoring and human health detection in ergonomics. *Int. J. Environ. Res. Public Health* (2020)

13. L. Wang, M. Peng, Q. Zhou, Pre-impact human health detection based on multi-source CNN ensemble. *IEEE Sens. J.* **20**(10) (2020)
14. Z. Liu, Implementation of a real-time fall detection system for elderly Korean farmers using an insole integrated sensing device. *Instrum. Sci. Technol.* (2019)
15. H.A. Hashim, S.L. Mohammed, S.K. Gharghan, Accurate human health detection for patients with Parkinson's disease based on a data event algorithm and wireless sensor nodes. *Measurement* (2020)
16. W. Fan, K. Wang, F. Cayre, Median filtered image quality enhancement and anti-forensics via variational deconvolution. *IEEE Trans. Inf. Forensics Secur.* **10**(5), 1076–1091 (2015)
17. K.S. Balamurugan, A. Sivakami, Lora-IoT based self-powered multi-sensors wireless network for next generation integrated farming. *Int. J. Sci. Technol. Res.* **10**, 1528–1533 (2019)

Energy Efficient Advancement-Based Dive and Rise Localization for Underwater Acoustic Sensor Networks



R. Bhairavi and Gnanou Florence Sudha

Abstract Underwater Acoustic Sensor Network (UWASN) is a developing technology for exploring Sub-Sea environment and has innumerable applications like deep-sea data acquisition, overseeing the contamination level, calamity prohibition, aided navigation and diplomatic surveyance applications. Previous works on data transmission in UWASN show that the nodes exhaust energy during data transmission and thereby lifetime of the nodes is reduced. This paper integrates localization with Normalized Advancement Factor (NAF) and Packet Delivery Probability to augment the lifespan of network. The NAF is computed from residual energy, Expected Transmission Count and the link cost. Considerable simulations were carried on for analysing the proposed technique and for comparing its performance with the existing VBF, HH-VBF and TORA techniques. Considering the network with 750 nodes, the proposed technique demonstrates better performance with a Packet Delivery Ratio of 95.473%, energy consumption of 0.429 J and Average End to End Delay of 2.73 s.

Keywords Dive and rise localization · Normalized advancement factor · Residual energy · Euclidean distance · Link cost · Packet delivery probability

1 Introduction

Currently, Underwater Acoustic Sensor Network (UWASN) have acquired tremendous attention due to its distinct characteristics and immense applications. As radio waves are intensely absorbed by water and optical signals have severe scattering losses while travelling in water, transmission is limited to Line of site range of

R. Bhairavi (✉) · G. F. Sudha

Department of Electronics and Communication Engineering, Pondicherry Engineering College, Puducherry 605014, India

e-mail: bhairavi@pec.edu

G. F. Sudha

e-mail: gfsudha@pec.edu

communication which makes them infeasible for communication in subsea environment. Thus, acoustic communication is selected as an ideal choice for communication in subsea environment.

Underwater Acoustic Sensor Networks comprises of numerous acoustic sensor nodes capable of sensing, processing and transmitting it to the destination node on the sea surface which in turn aggregates the information gathered [1] and transmits it to onshore data centre. In an Aquatic environment, the primary challenges [2] encountered in communication are localization, limited bandwidth, higher propagation delay, path loss, high bit errors and energy expenditure. The vulnerabilities in UWASN and its distinct characteristics, facilitates the need for uniquely designed routing protocols to perform effectively in subsea environment.

The rest of the paper is systematized in the following manner: Several works in the subaqueous environment has been described in Sect. 2. Proposed Enhanced DNRL with Energy efficient Advancement Factor (EEAF)-based data transmission in Underwater Acoustic Sensor Network is described in Sect. 3. Simulation results of proposed algorithm and its comparison with the existing techniques are described in Sect. 4 and lastly the conclusion is made in Sect. 5.

2 Literature Survey

Several authors have proposed works related to localization, routing and acoustic communication between the sensor nodes in the subaqueous environment. Yan et al. in [3] proposed the foremost routing protocol in the subsea environment utilizing the depth information. Each sensor node upon the reception of data packets checks its depth level with its preceding sender. If it is lesser (nearer to sink), it is chosen as a current forwarder else its data transmission is suppressed. Thus, the routing is made greedily considering the depth levels of sensor nodes with respect to the aquatic surface.

Xie et al. designed Vector-based Routing Protocol in [4]. A virtual pipe is built considering the coordinates of sender and designation node. Each sensor before participating in routing process validates whether its interspace length to forwarding vector is lesser when compared with the preconceived brink else suppresses its data transmission. Each node computes its desirableness factor depending on its projection to the forwarding vector to decide the holding time of data packets. The main disadvantage of Vector-based Routing Protocol is that it has greater energy depletion and the radius of the virtual pipe have a notable effect performance of the network.

Ayaz et al. in [5] designed the Hop-by-Hop Dynamic Addressing-Based (H2-DAB) that eliminates the prerequisites of 3D positional information of acoustic nodes in subsea environment. Each acoustic node is designated with an exclusive address composed of node ID and hop ID. Dynamic addressing potentiality of H2DAB makes it unconfined to any static configuration. When source node broadcasts the query packet, neighbouring acoustic nodes acknowledges by mentioning its exclusive address, and the acoustic node with lower hop ID is selected as the next

forwarder. The periodic update of exclusive address of acoustic node influences the routing process thus higher delay has to be endured.

Han et al. in [6] made a comparative analysis about several issues associated in routing protocols by categorizing them based on Energy, Geographic information and Hybrid. Their performances were examined in terms of path delay, multipath capacity, packet transmission efficiency and fidelity measurements.

Guo and Liu proposed in [7] an Anchor Free Localization Scheme (AFLA). Many oceanographic networks are equipped with Beacon or Anchor nodes whose positional coordinates are computed beforehand or obtained from external GPS. AFLA eliminates the requirement of special Anchor nodes and utilizes the information obtained from the deployed adjacent acoustic nodes. In order to cope up with the dynamic environmental conditions and to prevent the acoustic node moving away from the observing region, they are attached to static Beacon node.

Using Directional Beacon Scheme (UDB Scheme) [8] proposed by Luo et al., uses directional beacons (UDB) instead of standard omnidirectional localization approach. Autonomous Underwater Vehicle broadcasts the acoustic signals and sensors deployed in the subsea environment have to listen without active response, known as Silent Localization. Though UDB technique provides energy efficient localization, reducing the number of beacons is the prime challenge that needs to be addressed.

Anil et al. in [9] made a contrastive analysis of numerous localization schemes in UWASN that are Beacon nodes-based Localization and beacon free nodes localization techniques. Localization for Double Head Maritime Sensor Networks (LDSN) [10] proposed by Luo et al. is a triple stage localization technique. Self-Moored Node Localization is the preliminary stage that calculates the geographic coordinates of moored sensors with the aid of anchor nodes. Second stage, Underwater Sensor Localization USL iteratively localizes the other moored nodes. In the concluding Floating Node localization Algorithm (FLA) the geographic coordinates of the freely floating acoustic nodes are computed.

Cheng et al. [11] designed Underwater Silent Positioning Scheme (UPS) based on the values computed from the Time Difference of Arrival (TDOA), thus eliminating requirement of time synchronization between the acoustic nodes. Trilateration-based coordinate estimation is carried out and a modified ultra-wideband Saleh-Valenzuela model is constructed for modelling the subsea channel. The disadvantage of this technique is that acoustic nodes are confined to a limited region by the four beacon nodes which is a crucial problem in dynamic Underwater Environment.

Han et al. proposed in [12] a collaborative localization-based Vector-based Routing protocol to efficiently carry out data transmission among acoustic nodes in the harsh subsea environmental conditions. The main disadvantage of this technique is its excessive energy consumption.

Scalable Localization with Mobility Prediction (SLMP) technique proposed in [13] is a two-step stratified localization process: anchored node localization and acoustic sensor node localization. Mobility prediction of each acoustic sensor and the estimation of the future geographic coordinates is done by utilizing its past positional information.

Hop-by-Hop Vector-Based Forwarding (HH-VBF) proposed in [14] is an enhancement of VBF where each data transmitting sensor starts a vector to destination node. This leads to numerous virtual pipes from sender to sink node. The computation of vector depending on hop increases its complexity. Even though HH-VBF surpasses VBF by enhancing network performances, the main drawback is its increased signalling overheads.

Han et al. in [15] studied the effects of three distinct deployment techniques for acoustic nodes: Random, Cube and tetrahedron deployment technique. Tetrahedral deployment technique shows better performance wrt other deployment techniques by minimizing the localization error and improved localization ratio.

Jin et al. in [16] proposed the Routing Void Prediction and Repairing technique. This technique works as follows, the restoration location is computed by utilizing Particle Swarm Optimization which is done by increasing the connectedness of void region and reducing the AUV mobility range. Markov Chain Model-based void prediction is proposed to guarantee that AUV approaches for restoration job prior to void formation.

Narmeen et al. in [17] proposed an adaptive control packet collision avoidance (ACP-CA) technique to prevent mishap of control packets in channel reservation phase. The optimal relay node is selected by shortest propagation delay-based relay selection (SPD-RS) technique to facilitate retransmission in Data Transmission Phase (DTP).

Rahman et al. in [18] proposed a Totally Opportunistic Routing Algorithm (TORA) where acoustic nodes are classified as Single Transmission Node (STN) and Double Transmission Node (DTN) and utilizes Time of Arrival (TOA) and Trilateration-based Localization Technique. In the initial phase of setup, STN acts as a reference node for iterative localization. In the data Transmission phase, STN and DTN participate in routing. Implementation of Random Walk Mobility Model, complex data forwarding technique and higher energy requirements are its drawbacks.

Kayalvizhi et al. in [19] estimated the geographic position of underwater nodes considering Ocean Current Mobility Model. Dive and Rise Localization scheme is achieved by incorporating with the Distance Vector Hop (DV Hop) Algorithm for estimating the position of unlocalized acoustic nodes. The unlocalized acoustic nodes make use of Average Hop Distance (AHD) computed by Beacon node instead of the genuine distance which instigates an error in localization. Thus, maintaining localization precision is a crucial problem which has to be considered in DV Hop.

The crucial concern in the previous works is that the nodes exhaust energy during data transmission and thereby lifetime of the nodes is reduced. This motivated to propose a novel data transmission technique that integrates the node localization with the residual energy of acoustic nodes while choosing potential transmitting acoustic sensor. Here, modification of Dive and Rise Localization (DNRL) method is carried out. To prevent horizontal data transmission between the nodes of same depth and to reduce the energy expenditure, a parameter called as Normalized Advancement Factor (NAF) is introduced. The NAF is computed from the residual energy, expected Transmission Count and the link cost. From a batch of nodes selected

based on the NAF and distance to destination node, the nodes with highest priority value are chosen as potential transmitting node and are used for the packet transmission. By this, the network efficiency is enhanced. The periodical computation of the geographic coordinates of the nodes by DNRL can efficiently cope up with the dynamic UWASN. Simulation results show that the proposed Enhanced DNRL with Energy Efficient Advancement Factor (EEAF)-based Data Transmission performs better than the existing techniques.

3 Proposed Enhanced Dive and Rise Localization (DNRL) with Energy Efficient Advancement Factor (EEAF)-Based Data Transmission Technique

The proposed method comprises of three stages, namely Dive and Rise technique with Enhanced Weighted Centroid localization, Normalized Advancement Factor computation, Potential Batch Selection and Transmitting acoustic node Selection.

3.1 Dive and Rise Technique with Enhanced Weighted Centroid Localization

Dive and Rise (DNR) Localization Technique consists of moveable anchor (DNR Beacon) nodes incorporated with GPS, to calculate its positional coordinates when it freely floats in the sea surface. DNR Beacon nodes move in a predetermined pattern and dive and rise vertically in the subsea environment. In the diving phase, DNR nodes regularly broadcasts its positional information to unlocalized acoustic nodes in its coverage. After rising (at the end of each cycle), DNR node recomputes its positional coordinates. Unlocalized nodes estimates its geographic information (say x and y coordinates) by Enhanced Weighted Centroid localization (EWCL) algorithm.

EWCL algorithm consists of the following steps:

Estimation of lowest hop count (hc).

Estimation of average hop distance $AVGhd(m)$.

Estimation of geographic coordinates of unlocalized acoustic node.

Step 1: Estimation of lowest hop count (hc):

Primarily, Each DNR node broadcasts $\langle xl, yl, hc \rangle$ information, where $\langle xl, yl \rangle$ denotes its geographic information in 2D and hc denotes its hop count value. The initial value of hc is designated as 0. Here, the unlocalized acoustic node (r) preserves a register with $\langle l, xl, yl, hrl \rangle$ for each beacon node ' l ' less than ' n ' hops and periodically receives beaonic messages from DNR nodes, in turn compares its hrl value with the received one and the value of n . If it is lesser, the received hrl value is ignored else increased by 1 and the updated value is stored in the table maintained by

unlocalized node. The table is broadcasted to the neighbouring unlocalized acoustic nodes.

Step 2: Calculate average hop distance $AVGhd(m)$:

Each DNR node calculates its average hop distance using Eq. (1)

$$AVG_{hd(m)} = \frac{\sum_{m=0; p=0}^u \sqrt{(x_m - x_p)^2 + (y_m - y_p)^2}}{h_m} \quad (1)$$

where u is the aggregate count of DNR sensor, P represents all other DNR nodes and hm denotes the hops in between two DNR nodes m and p , (x_m, y_m) and (x_p, y_p) denotes the positional coordinates of beaconic nodes m and p .

DNR nodes broadcast the computed value of average hop distance $AVGhd(m)$ to the unlocalized acoustic nodes less than n hops which in turn preserves closest beaconic node's information alone and discards the remaining packets.

The unlocalized acoustic node calculates the distance from DNRm utilizing Eq. (2),

$$dist_m = AVG_{hd(m)} * h_m \quad (2)$$

Step 3: Estimation of geographic coordinates:

Here, weight parameter utilized to calculate the 2 dimensional geographic coordinates (x, y) of the unlocalized acoustic nodes is generated from Eq. (3),

$$Weights = \left(\frac{\sum_{m=1}^u h_{am}}{u * h_{am}} \right)^{\frac{r}{AVG_{HD(p)}}} \quad (3)$$

where

h_{am} is lowest hop count of node 'a' from beaconic node 'm'.

u denotes the aggregate count of beaconic nodes in the network.

$AVG_{HD}(p)$ is the average hop distance of closest beaconic sensor p to unlocalized node a and

r represents the range of communication.

x and y coordinates of unlocalized acoustic nodes are computed from the following Eq. (4),

$$Xb = \frac{\sum_{m=1}^u w_m x_m}{\sum_{m=1}^u w_m}, \quad yb = \frac{\sum_{m=1}^u w_m y_m}{\sum_{m=1}^u w_m} \quad (4)$$

The third coordinate (z) depicts the depth which is calculated from Eq. (5),

$$Depth (Z) = \text{specific volume/GR} \quad (5)$$

Gravity variation and Specific volumes [20, 21] are computed by Eqs. (6) and (7),

$$GR = 9.780318 * (1.0 + (5.2788e^{-3} + 2.36e^{-5} * Q^2) * Q^2) + 1.092e^{-6} * k \tag{6}$$

where $Q = \text{Sin}(\text{Lat}/57.29578)$;

$$\text{specific volume} = (((-1.82e^{-15} * k + 2.279e^{-10}) * k - 2.2512e^{-5}) * k + 9.72659) * k \tag{7}$$

where $k = 10,000$ dbar.

3.2 Normalized Advancement (NADV) Computation

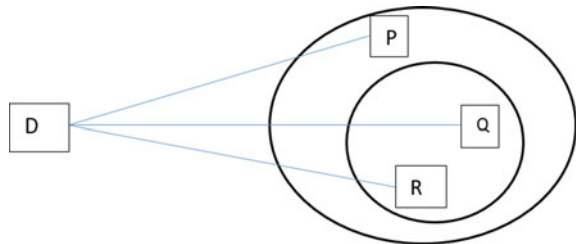
Once the nodes are localized, optimal neighbour nodes between the source and sink nodes must be selected to involve in the routing process. To illustrate this, consider a network with 3 acoustic sensor nodes P, Q, R deployed in various depths and a destination node D as depicted in Fig. 1. Sensor Q has a data to be transmitted to sensor D which is far away from its range of transmission or area of coverage. Thus, node Q depends on its nearby nodes to pass on the data packets. Both the nearby nodes P and R are in the region of coverage of node Q . Node Q selects node P to transmit data since its distance to destination node D is comparatively lesser when compared to node R . But node Q initiates its retransmission of lost data if node P has a greater error rate when compared with node R resulting in increased energy consumption. Thus, link cost is a prime metric that has to be considered while choosing the succeeding hop node for data transmission.

The length between sensor R and the nearby sensor Q relative to destination is given by Eq. (8),

$$AF(P) = \text{Dis}(Q) - \text{Dis}(P) \tag{8}$$

where, $\text{Dis}(Q)$ and $\text{Dis}(P)$ denotes the distance of nodes P and Q to the Destination node D .

Fig. 1 Normalized advancement factor computation model



$AF(P)$ represents Advancement Factor of node P towards the Destination node D . Greater $AF(P)$ denotes, greater probability of data transmission of node P to node D . Link cost is computed from Eq. (9)

$$C(p) = [1 - Eres(p)/Einitial(p)] + [1 - ExpTC(P, Q)/ExpTCmax] \quad (9)$$

Here, $Eres(p)$, $Einitial(p)$ denotes the residual energy and initial energy of node P . Consider, $ExpTC(P, Q)$ to denote the expected transmission count of link amidst sensor P and Q and $ExpTCmax$ to denote highest value of expected Transmission count.

The value $ExpTCmax$ is dynamic and varies with respect to network conditions.

$$AF(P) = \begin{cases} D - \frac{D(P)}{r} \\ D - \left\{ \frac{D(P)}{r} + (Eres(p)/Einitial(p)) \right\} \end{cases} \quad (10)$$

D denotes depth of the current sender, $deD(P)$ denotes the depth of nearby sensor P , and r is transmission range of node P , $Eres(p)$, $Einitial(p)$ denotes the residual energy and commencing energy of sensor P .

When two nearby sensors are in equivalent pressure level, acoustic node which has highest residual energy among them has more probability to be selected as intermediate node for data transmission to the destination node.

Normalized Advancement Factor (NAF) of node P is computed from the following equation,

$$NAF(P) = AF(P)/C(P) \quad (11)$$

3.3 Potential Batch Selection (PBS)

Potential batch selection is the process of selecting a subgroup of neighbouring nodes (depending on NAF) to involve in the routing process to perform data transmission. When a node has data to be transmitted, it begins to discover the nearby nodes that are appropriate for packet forwarding process.

In PBS, the initial step is to compute the length between source and sink. It is followed by computation of length amidst median sensor and destination. Finally, computed values in the previous steps are subtracted.

$$PB = Dis(q, dl) - Dis(p, dl) \quad (12)$$

Here,

$\text{Dis}(al, dl)$ represents the Euclidean length amidst acoustic sensor a and destination d ,

Dl —denotes the nearest destination of acoustic node (al) at the given time which can be depicted in Eq. (13),

$$Dl = \text{Arg min } \forall \in Dl(t). \text{Dis}(al, dl) \quad (13)$$

Thus, a set of efficient candidate forwarders are obtained to participate in data transmission.

3.4 Transmitting Acoustic Node Selection

The acoustic nodes obtained from PBS step are given priority at time instance t , which is computed from Eq. (14),

$$\text{PRIL}(t) = \text{NAF}.P(d, m) \quad (14)$$

Here,

$P(d, m)$ denotes delivery probability of m sized bits for length d .

Finally Packet delivery probability is computed from Eq. (15),

$$P(d, m) = (1 - \text{Pe}(d))m \quad (15)$$

The bit error probability over length d is,

$$\text{Pe}(d) = \frac{1}{2} \left(1 - \sqrt{\frac{\gamma(d)}{1 + \gamma(d)}} \right) \quad (16)$$

where

$\gamma(d)$ is the SNR over length d .

The nodes with highest $\text{PRIL}(t)$ value is chosen as potential transmitting node and involves in the packet transmission. If it fails, then the next node with highest $\text{PRIL}(t)$ value will be selected. It is iteratively carried on until all the packets are delivered without any failure.

In EEAF, when node k receives a data packet, it waits for node 1 to node $(k - 1)$ to complete its data transmission after the waiting time, upon the absence of acknowledgement, node k begins its propagation. Meanwhile, node K suppresses and drops its forwarding process if it overhears same data transmission by nodes having highest $\text{PRIL}(t)$ value.

Thus, the waiting time is computed as:

$$T_{\text{wait}} = T_{\text{propagating}} + \sum_{i=1}^k \frac{D(ni, ni + 1)}{v} + k * T_{\text{processing}} \quad (17)$$

where

$T_{\text{propagating}}$ denotes time taken for propagation,

$T_{\text{processing}}$ represents time taken for processing a packet,

$D(ni, ni + 1)$ denotes the length from sensor ni to sensor $ni + 1$,

V is the speed of sound in water.

Upon selection of the best forwarder, the sender sends packets to the chosen sensor and it is iteratively carried on until all the packets are transmitted to destination.

4 Simulation Results

Here, The performance of Enhanced DNRL with Energy Efficient Advancement Factor (EEAF)-based Data Transmission is evaluated and compared to the existing VBF [4], HH-VBF [14] and TORA [16] utilizing Aquasim (NS 2.30) which is highly authentic, adaptable to simulate subsea acoustic network. Here, we have deployed 750 acoustic nodes randomly in 3d space of 1800 m * 1800 m * 1800 m and 50 sink nodes floating above the aquatic surface. The movement of acoustic nodes are defined by Meandering Current Mobility (MCM) Model. It is defined by Eq. (18),

$$\varphi(x, y, t) = -\tanh \frac{[y - B(t) \sin(K(x - ct))]}{\sqrt{1 + K^2 B^2(t) \cos^2(K(x - ct))}}. \quad (18)$$

$$B(t) = A + \varepsilon \cos(\omega t) \quad (19)$$

where A is average meander width, c is phase speed, K is no. of meanders, ω is the frequency and ε is the amplitude. Velocity field in the kinematical model is estimated as shown in Eq. (20).

$$V_x = K_1 * \bar{\lambda} * v * \sin(K_2 * x) * \cos(K_3 * y) + K_1 * \bar{\lambda} * \cos(2K_1 t) + K_4 \quad (20)$$

$$V_y = -\bar{\lambda} * v * \cos(K_2 * x) \sin(K_3 * y) + K_5 \quad (21)$$

where V_x is the speed in x -axis, V_y is the speed in y -axis, K_1, K_2, K_3, K_4, v and $\bar{\lambda}$ are variables which are closely related to environmental factors. These parameters shows variation in diverse environmental conditions. MCM is executed from Eqs. (18)–(21).

Speed of acoustic node is 2 m/s and its transmission range is 400 m. In VBF and HHVBF, the ambit of the virtual pipe is 150 m. Initial energy of acoustic sensor is 1250 J. During the routing process, the energy depletion caused by various sensor activities like sending, receiving, packet header reading and idle listening are 3 W, 0.75 W, 0.2 W and 15 MW.

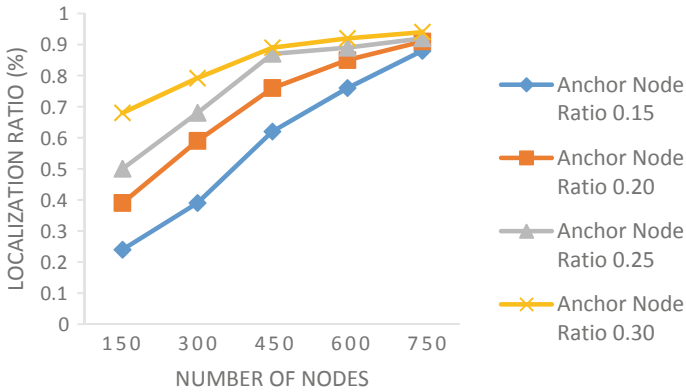


Fig. 2 Localization ratio by varying the percentage of anchor nodes

The Performance Metrics used for evaluation of the proposed technique are defined as follows:

4.1 Consequence Based on Beacon Node's Count

The performance of proposed technique while varying the percentage of beacon nodes is examined in terms of localization ratio. Beacon nodes whose geographic coordinates are known in advance are called as Anchor nodes. The beacon node's count has a significant part in data gathering and localization. The simulations are carried out by varying the anchor node ratio from 15 to 30% of the aggregate count of sensor for determining required beacon node's count to have an outstanding performance. Figure 2 depicts that, when number of beacon nodes are increased, the localization ratio is also increased. It is observed that, in low density networks, the percentage of beacon node has more effect in localization and it begins to decrease in dense networks.

4.2 Overall Energy Expenditure

Total Energy expenditure is the summation of energy expenditure caused due to localization and routing. In the existing VBF and HH-VBF schemes, radius of the virtual pipe is the main factor that influences the energy consumption of the network. From Fig. 3, it can be inferred that the proposed DNRL with EEAF-based data transmission has very less energy expenditure due to genuine usage of broadcasting characteristics and NAF-based data transmissions. From Fig. 3, it is observed that in low density networks, energy expenditure due to localization is high. But when

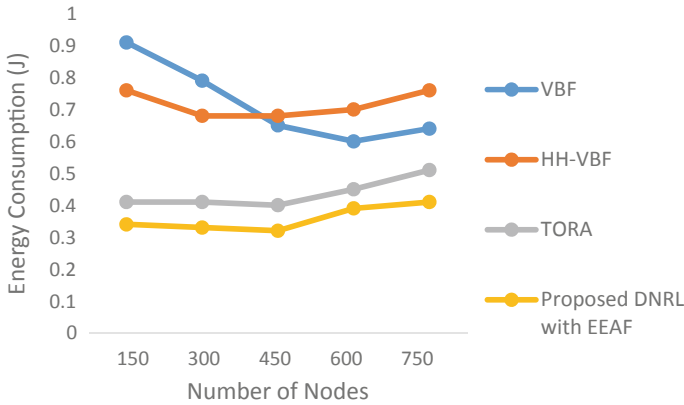


Fig. 3 Energy consumption for varying number of nodes

total number of sensor increments, energy expenditure gradually decreases up to certain network density, after which the average energy expenditure starts increasing as routing begins to influence the network.

4.3 End to End Delay

Performance of the proposed DNRL with EEAF-based data transmission is analysed in terms of Average End to End Delay while gradually increasing total number of nodes. When the number of nodes is increased from 150 to 750, the connectivity rate increases while decreasing the Average End to End Delay which is portrayed in Fig. 4. In existing data transmissions technique, time taken to define virtual pipe

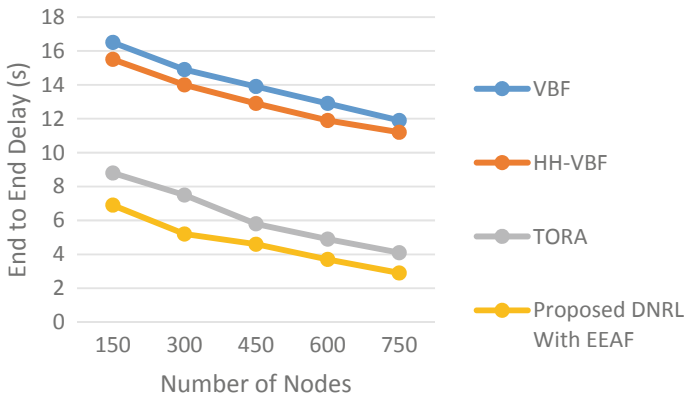


Fig. 4 End to end delay

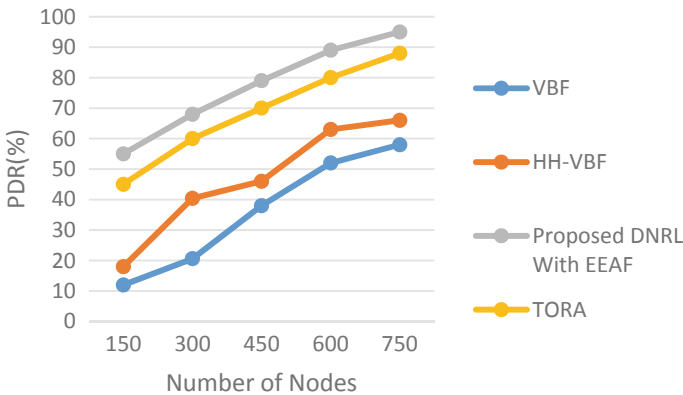


Fig. 5 PDR

between the source and sink and the holding time of data packets leads to higher End to End Delay. If an appropriate relay node is present outside the virtual pipe, it further increases End to End Delay. The proposed DNRL with EAAF-based data transmission reduces the holding time of the packet and uses best relay node which eventually minimizes the number of retransmissions. Thus, the proposed technique has minimal End to End Delay as shown in Fig. 4.

4.4 Packet Delivery Ratio

The proposed DNRL with EAAF-based data transmission is compared with the existing VBF and HH-VBF-based routing and TORA in terms of Packet Delivery Ratio while varying total number of sensors. We could infer from Fig. 5 that when node density increases, inter nodal distance decreases, which eventually increases the connectivity and results in better PDR. In the existing algorithms, residual energy of nodes is not considered in data transmission, which causes failure of data transmission. Proposed DNRL with EAAF-based data transmission considers initial and the residual energy of sensor in the computation of Advancement Factor of a node, and thus achieves higher PDR compared to the existing works.

5 Conclusion

This work presents Enhanced DNRL with Energy Efficient Advancement Factor (EAAF)-based Data Transmission which integrates Localization of nodes with Normalized Advancement Factor and Packet Delivery Probability, to greedily transmit the data packets to destination node is proposed. DNRL with EWCL is

used to localize the nodes. Packet Delivery Probability and NAF computed from the residual energy and link cost are used in calculation of priority of nodes to participate in the data transmission. The experimental results manifest that EEAF technique shows improvement in when compared with existing VBF, HH-VBF and TORA techniques.

References

1. H. Maqsood, N. Javaid, A. Yahya, B. Ali, Z.A. Khan, U. Qasim, *MobiL-AUV: AUV-aided localization scheme for underwater wireless sensor networks*, in *10th International Conference on Innovative Mobile Internet Services Ubiquitous Computing (IMIS)*, Japan (2016), pp. 170–175
2. Z. Wadud, K. Ullah, A.B. Qazi, S. Jan, F.A. Khan, N. Minallah, *An efficient routing protocol based on stretched holding time difference for underwater wireless sensor networks* (2019)
3. H. Yan, Z.J. Shi, J.H. Cui, *DBR: depth-based routing for underwater sensor networks*, in *7th International IFIP-TC6 Networking Conference on Ad Hoc Sensor Networks, Wireless Networks, Next Generation Internet* (Springer, Heidelberg, 2008), pp. 72–86
4. P. Xie, J.H. Cui, L. Lao, *VBF: vector-based forwarding protocol for underwater sensor networks*, in *5th International IFIP-TC6 Networking Conference on Networking Technologies, Services, and Protocols, Performance of Computer and Communication Networks; Mobile and Wireless Communications Networks*, Portugal (2006), pp. 1216–1221
5. M. Ayaz, A. Abdullah, I. Faye, Y. Batira, *An efficient dynamic addressing based routing protocol for underwater wireless sensor networks*. *Comput. Commun.* **35**, 475–86 (2012)
6. G. Han, J. Jiang, N. Bao, L. Wan, M. Guizani, *Routing protocols for underwater wireless sensor networks*. *IEEE Commun. Mag.* 0163-6804 (2015)
7. Y. Guo, Y. Liu, *Localization for anchor-free underwater sensor networks*. *Comput. Electr. Eng.* **39**, 1812–1821 (2013)
8. H. Luo, Y. Zhao, Z. Guo, S. Liu, P. Chen, L.M. Ni, *UDB: using directional beacons for localization in underwater sensor networks*, in *14th IEEE International Conference on Parallel and Distributed Systems. ICPADS*, Australia (2008), pp. 551–558
9. C.B. Anil, S. Mathew, *A survey and comparison study of localization in underwater sensor networks*. *Int. J. Comput. Sci. Mob. Comput.* **3**, 23–29 (2014)
10. H. Luo, K. Wu, Y.-J. Gong, L.M. Ni, *Localization for drifting restricted floating ocean sensor networks*. *IEEE Trans. Veh. Technol.* **65**, 9968–9981 (2016)
11. X. Cheng, H. Shu, Q. Liang, D.H.C. Du, *Silent positioning in underwater acoustic sensor networks*. *IEEE Trans. Veh. Technol.* **57**, 1756–1766 (2008)
12. S. Han, J. Yue, W.X. Meng, X. Wu, *A Localization Based Routing Protocol for Dynamic Underwater Sensor Networks* (IEEE Global Communications, Washington, 2016), pp. 1–6
13. Z. Zhou, Z. Peng, J.H. Cui, Z. Shi, A. Bagtzoglou, *Scalable localization with mobility prediction for underwater sensor networks*. *IEEE Trans. Mob. Comput.* **10**, 335–348 (2011)
14. N. Nicolaou, A. See, P. Xie, J.H. Cui, D. Maggiorini, *Improving the robustness of location-based routing for underwater sensor networks*, in *OCEANS* (IEEE Press, UK, 2007), pp. 1–6
15. G. Han, C. Zhang, L. Shu, J.J.P.C. Rodrigues, *Impacts of deployment strategies on localization performance in underwater acoustic sensor networks*. *IEEE Trans. Ind. Electron.* **62**, 1725–1733 (2015)
16. Z. Jin, Q. Zhao, Y. Luo, *Routing void prediction and repairing in AUV-assisted underwater acoustic sensor networks*. *IEEE Access* **8**, 54200–54212 (2020)
17. R. Narmeen, I. Ahmad, Z. Kaleem, U.A. Mughal, D.B. Da Costa, S. Muhaidat, *Shortest propagation delay-based relay selection for underwater acoustic sensor networks*. *IEEE Access* **9**, 37923–37935 (2021)

18. Z. Rahman, F. Hashim, M.F.A. Rasid, M. Othman, Totally opportunistic routing algorithm (TORA) for underwater wireless sensor network. *PLoS ONE* **13** (2018)
19. C. Kayalvizhi, R. Bhairavi, G.F. Sudha, Localization of nodes with ocean current mobility model in underwater acoustic sensor networks, in *International Conference on Computer Networks and Inventive Communication Technologies* (Springer, India, 2018), pp. 99–109
20. X. Che, I. Wells, G. Dickers, P. Kear, X. Gong, Re-evaluation of RF electromagnetic communication in underwater sensor networks. *IEEE Commun. Mag.* **48**, 143–151 (2010)
21. V.A. Del Grosso, New equations for the speed of sound in natural waters (with comparison to other equations). *J. Acoust. Soc. Am.* **56**, 1084 (1974)

Performance Comparison of Machine Learning Algorithms in Identifying Dry and Wet Spells of Indian Monsoon



Harikumar Rajaguru and S. R. Sannasi Chakravarthy

Abstract For water-related industries, the characteristics of wet spells and intervening dry spells are highly useful. In the face of global climate change and climate-change scenario forecasts, the facts become even more important. The goal of this study is to determine the wet and dry spells that occur throughout the monsoon season in peninsular India. The India Meteorological Department (IMD) observations were made over the course of a hundred days, from October 23 to January 30, 2019, with 334 rainy days and 60 dry days. The IMD data provides ten observational characteristics in peninsular India, including maximum, minimum, and average temperatures, rainfall wind speed, atmospheric pressure, illumination, visibility, relative cloud density, and relative humidity. Four statistical factors, such as mean, variance, skewness, and kurtosis, further decrease these characteristics. The observed characteristics and their statistical parameters follow a nonlinear trend, as seen by histogram plots. For assessing the classification performance, a collection of four algorithms is used: Logistic regression, gradient boosting, Gaussian mixture model, and firefly with Gaussian mixture model. During both the dry and rainy spells of monsoon observation, all of the classifiers achieve greater than 85% classification accuracy (average).

Keywords Dry spell · Wet spell · Monsoon · Water scarcity · Machine learning · Firefly

1 Introduction

Rainfall occurs in spells in tropical monsoonal regions and is a seasonal phenomena [1]. Classifiers and other criteria are used to describe the start and conclusion of the rainy season, as well as the frequency, quantity, and intensity of rainfall, the duration of wet spells (WSs), and the duration of intervening (between two rain spells) dry spells (DSs). Weather forecasting is a work that uses science and technology to

H. Rajaguru (✉) · S. R. Sannasi Chakravarthy
Department of Electronics and Communication Engineering, Bannari Amman Institute of Technology, Sathyamangalam 638401, India

anticipate the state of the atmosphere for a certain time and location in the future [1]. Since ancient times, humans have sought to forecast the weather. Rainfall prediction is one of the most essential aspects of weather forecasting, since it is crucial for food production, water resource management, and many other outdoor activities. The problem's most important difficulty is determining the rainy season's annual beginning and ending dates, as well as identifying wet and dry periods in the rainfall time distribution [2]. Using classifiers and parameters, we attempt to determine the wet and dry dates in a given monsoon season in this work. The India Meteorological Department (IMD) observations in this study were obtained during a period of one hundred days, from October 23 to January 30, 2019, with 334 rainy days and 60 dry days. From the IMD data for peninsular India, ten observational characteristics such as maximum, minimum, and average temperatures, rain fall wind speed, atmospheric pressure, illumination, visibility, relative cloud density, and relative humidity are obtained with the label of wet and dry spell. Using classifiers and input parameters, the number of wet and dry spells is calculated. The outcomes are then compared with the findings of IMD labels.

The workflow proposed for the research is depicted in Fig. 1. From this figure, the database is visually analyzed using graphs and pre-processed for their better results and the data classification is then implemented through the four distinct ML algorithms, namely logistic regression, Gradient boosting, Gaussian mixture model, and firefly with Gaussian mixture model classifiers. As a final point, the comparison of results is done for the performance of classifying dry and wet spells.

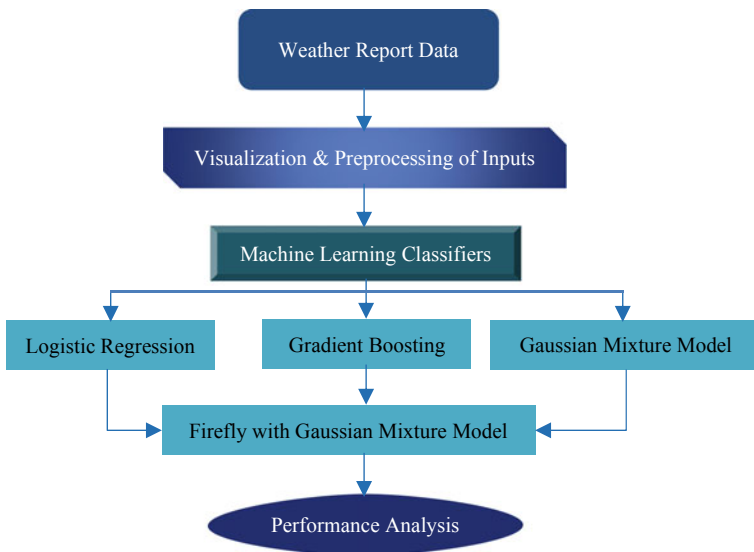


Fig. 1 Workflow proposed

2 Materials and Methods

2.1 Input Dataset

From the India Meteorological Department (IMD) data that acquired for peninsular India, includes the output classes of dry and wet spells during the season of monsoon. This has the ten observational attributes such as max, min, and average temperature values, speed of rainfall wind, pressure of atmosphere, relative cloud-density and humidity, visibility values, and illumination values. In the input dataset, the total number of rainy days is 334, while the total number of dry days is 60. The number of rainy days outnumbers the number of sunny days. As a result, it is collected during the rainy season.

2.2 Data Visualization of Input Data

Irrespective of classification problem solving through ML algorithms, the analysis of data input is very crucial for further research phases. The univariate input data analysis through a distribution plot has carried out and is given in Fig. 2.

From Fig. 2, the average temperature and humidity attribute values are inferred as a much right-skewed one in the input dataset. In addition, it reveals that the

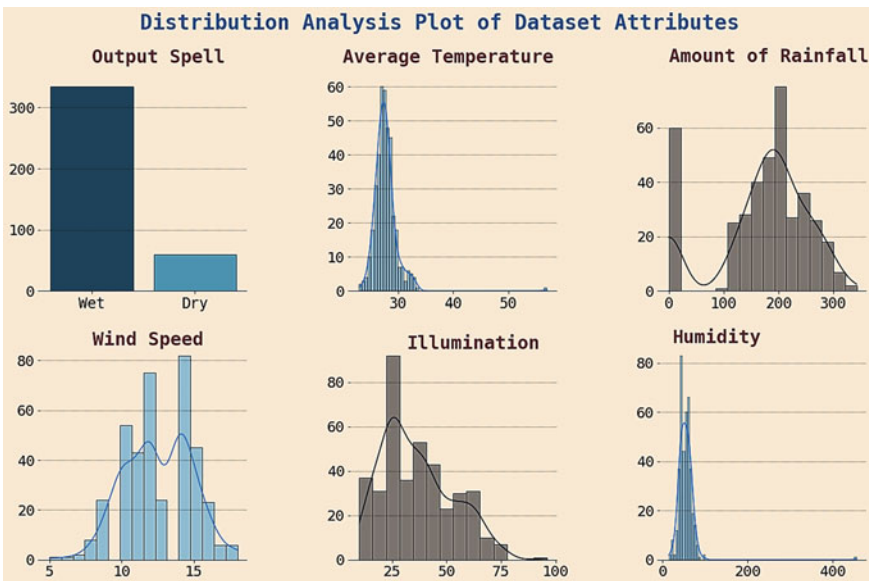


Fig. 2 Univariate input analysis through distribution plot

dataset includes more rainfall intensity values for rainy spell in the range of 100–350%, and the wind speed is equally distributed throughout the season. Moreover, the illumination values of input data seem to be substantially skewed. The above discussion implies that the input data needs to be normalized before classifying the data.

2.3 Data Preprocessing

As from the input dataset, the distribution plot of Fig. 2 provides that the input measurements are being highly nonlinear and overlapped. Also, while inspecting each attributes of the input data, it is found that the data needs to be normalized before data classification. And so, the input measurements are normalized then though the use of Standard Scalar technique [3]. As a result, the data input is now ready for next succeeding step of data classification as depicted in Fig. 1.

3 Classification Algorithms

The paper employs a hybridized algorithm that includes the advantage of Gaussian mixture model concept with the nature inspired firefly algorithm. Also, the base classifiers, namely logistic regression, Gradient boosting, and Gaussian mixture model algorithms are employed. The algorithms of the above-said classifiers are detailed in this section.

3.1 Classification Using Logistic Regression (LR) Algorithm

In this type of logistic regression means of classification, statistical methods are adopted for predicting the binary targets that includes rainy wet and non-rainy dry spells [4]. The LR algorithm being a linear learning technique, it generally make use of the odds of an event for performing predictions with logistic regression concept. For this action, the LR approach employs a simple sigmoidal mathematical function for mapping of all input data points to their binary targets [5]. As a result, an S-shaped curve can be represented as the note of traditional logistic function. This could be depicted mathematically using a simple sigmoidal equation as shown in below equation of [5],

$$\text{Sigmoidal Function} = \frac{1}{1 + e^{-x}} \quad (1)$$

3.2 Gradient Boosting (GB) Classifier

The Gradient boosting algorithm is a simple collection of ML models that includes several weak-learning algorithms for building a powerful prediction classifier [6]. While implementing Gradient boosting, decision trees are commonly utilized. The Gradient boosting models are gaining popularity as a result of their ability to categorize complicated information of input dataset [6]. The decision tree (DT)-based GB is employed in this paper, where the implementation steps are summarized below [7],

Step 1: Computing the average value of the output binary targets.

Step 2: Computing the residual values computed as a difference of actual and prediction.

Step 3: Construction of DT is done.

Step 4: Predicting the output binary target by the use of every trees created in the ensemble.

Step 5: Repeat the computation of new residual values.

Step 6: Repeating the steps of 3 to 5 with the condition of matching the number of iterations with the amount of estimators used.

Step 7: Once completion of training, make use of all the trees in the ensemble for making a conclusion on final prediction as one of the output targets.

3.3 Gaussian Mixture Model (GMM) Classifier

Gaussian mixture model (GMM), probability-based algorithm used more common for depicting normally distributed sub population over overall populations [8]. The GMM algorithm is actually utilized for unsupervised learning problems for learning the sub-population and the automatic assignment of sub populations. However, in this paper, the GMM algorithm is employed for classification or supervised learning problems for learning the boundaries of sub population. After the training phase, that is, once fitting the data with GMM, it can classify which of the cluster a newer data point belongs to. But, this is possible only if the GMM is provided with the target labels. Here, it is very important that the clusters are chosen arbitrarily, and its probability density function can be defined as [9],

$$y = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (2)$$

where μ and σ represent the mean and standard deviation of the input data. Here, the probability of an input data point can be calculated as [9],

$$p(x) = \sum_{i=1}^k \varphi_i \mathfrak{N}(x|\mu_i, \sigma_i) \quad (3)$$

3.4 Firefly with GMM as a Classifier

Unlike using GMM as an unsupervised learning approach, the paper utilizes the GMM for solving supervised learning problem. However, the performance of the GMM would not be a satisfied one while comparing with other conventional ML algorithms. Thus, in order to improve their prediction ability in supervised learning problems, the paper hybridized the metaheuristic firefly algorithm with the Gaussian mixture model algorithm. This type of hybrid implementation involves in helping the prediction by removing the insignificant data points and outliers and so making the GMM model to provide better accuracy in supervised approaches. The parameters of firefly algorithm are selected as experimented in our previous work [10].

4 Results and Discussion

The research work implemented as depicted in Fig. 1 of this paper is done using Google Colab which is an online IDE research base provided by Google though a personal Gmail account used on the web browser, Google Chrome. The data inputs after preprocessing as illustrated in Sect. 2 have accordingly splitted for the phase of classification with 70:30 standard with 70% of training inputs and 30% of testing inputs. As depicted in Fig. 2 (first column plot), the input data comprises more wet sample class targets than dry class target, so there might be a problem of class imbalance. For overcoming this class imbalance problem, SMOTE type [11] of splitting data is used. The number of total input data taken and its split up for training and testing phase are portrayed in Fig. 3. In addition to this, prior to classification part of implementation, the input data processed can be normalized through min–max standardization [3] technique as per the equation shown below,

$$x_{\text{norm}} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \quad (4)$$

where x_{\min} , x_{\max} denote the minimum and maximum of data point values and x refers to the input vectors.

As discussed above, after normalization and splitting of preprocessed data, the considered ML algorithms are then fitted (trained) and tested to check the efficacy in predicting dry and wet spell. That is, the LR, GB, and GMM classifiers together with their hybridized classifier model, i.e., firefly with GMM algorithms are employed for

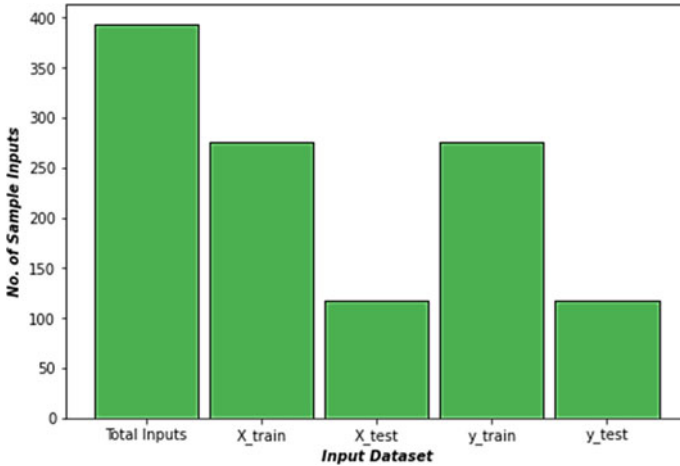


Fig. 3 Splitting of data inputs for classification

this prediction. In addition, the performance is fivefold cross-validated to provide better results. The results obtained in classifying spells can be assessed using the benchmark measures [12] which are generally a standard one in the problems of binary prediction. The metrics adopted in the paper are sensitivity, accuracy, specificity, F1 score, and precision. Here, the above-said performance measures are derived or taken through the confusion matrix (CM) which comprises the particulars of true and false negatives and positives. These results are then validated through a standard measure, Matthews Correlation coefficient (MCC).

The obtained elements of CM regarding each classification algorithms are graphically plotted in Fig. 4. It is noted from this graph that the amount of true negative

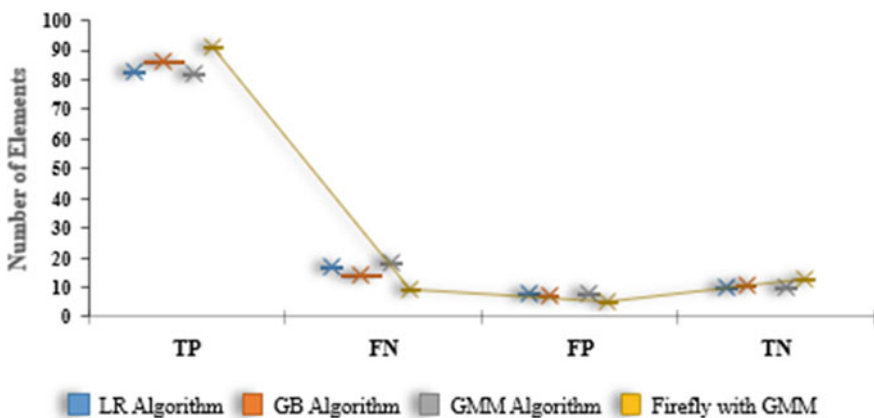


Fig. 4 Plot of confusion matrix of ML algorithms

Table 1 Performance of algorithms used for classification

Classifiers	Performance comparison (%)					
	Sensitivity	Specificity	Accuracy	Precision	F1 score	MCC
LR classifier	83	55.56	78.81	91.21	86.91	33
GB classifier	86	61.11	82.2	92.47	89.12	41.45
GMM model	82	55.56	77.97	91.11	86.32	31.74
Firefly with GMM classifier	91	72.22	88.14	94.79	92.86	58.37

and positive elements (TN and TP) is high for the Gradient Boosting classification algorithm as compared with the LR classifier. And while including the performance of the GMM classifier, still the true negative and positive elements of CM is high for the Gradient Boosting algorithm. As implemented with the hybrid algorithm for classification that includes the firefly algorithm together with the GMM model, the true prediction elements of CM significantly improved as illustrated in the plot of Fig. 4. In a similar way, while considering the false misclassification, the GMM model as a classifier provides more FN and FP elements as compared with other base classifiers. However, the same GMM classifier together with the Firefly algorithm provides very less misclassification in this prediction problem. Moreover, the prediction, i.e., the amount of false classification gets depreciated and so the amount of correct predictions gets improved for the Firefly with GMM model as depicted in Fig. 4. The discussion on the results obtained using performance metrics will be further discussed in detail.

The performance obtained using the above-said classification algorithms are listed in Table 1. In this table, the logistic regression algorithm's performance as compared with the GMM algorithm is higher. But the Gradient boosting algorithm provides a better performance than this logistic regression classifier. This implies that the Gradient boosting classifier provides a better accuracy of 82.2% accuracy, 92.47% of precision, and 89.12% of F1 score. Here, it is noted that while comparing the individual base algorithms, the gradient boosting algorithm yields the high classification, and so, the value of MCC for GB classifier is attained as 41.45 which is supreme over other base classifiers. For further improving its performance, the hybridization technique is used by making use of the metaheuristic approach.

The classification performance of this hybrid firefly with GMM algorithm while comparing with other algorithms is plotted graphically in Fig. 5. In this graph, the hybridized firefly with GMM algorithm yield 88.14% of accuracy with a precision of 94.79%, and F1 score of 92.86%. These obtained performances are validated by the MCC attainment value of 58.37, and it is obviously higher than other employed classification algorithms. Hence, the hybridized firefly together with the GMM algorithm attains a maximum performance than the LR, GB, and GMM algorithms as depicted in Table 1 and Fig. 5.

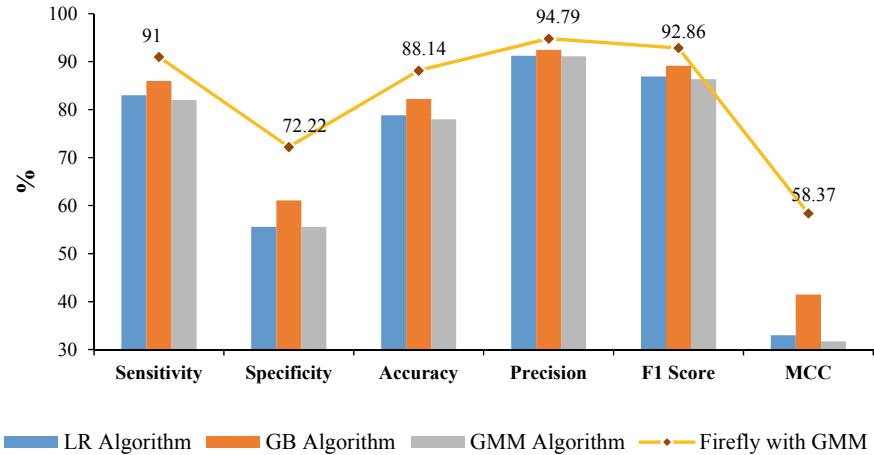


Fig. 5 Graphical comparison plot used for classifiers' performance

5 Conclusion and Future Scope

The work proposed a hybridized approach for developing a weather forecasting predictor used in the prediction of rainfall through several measurements. The algorithms used for classification are logistic regression, gradient boosting, and Gaussian mixture model. Here, the prediction performance has been improved further by incorporating the firefly algorithm together with the above-said Gaussian mixture model algorithm. For this evaluation, the dataset considered has ten different measurements with the inputs taken from 334 rainy days and 60 dry days. That is, the dataset comprises of 694 input samples taken from 694 different climatic days. And this input data is analyzed graphically using the distribution plot which revealed the nonlinearity nature of the inputs. Then, the input data is normalized and fed for different classifiers for prediction. For this, the input data had been splitted using a standard ratio of 70:30 by means SMOTE technique. As the aim of the research, the work attains a supreme performance of 88.14% accuracy with the improved value of MCC as 58.37. Several algorithms and approaches are proposed for efficient rainfall prediction are now available in literature, but there is still a need for a comprehensive literature review and systematic mapping research that can represent proposed solutions, current challenges, and current developments in this sector. The outcome of this research add to the existing body of knowledge in numerous ways. For engineers, scientists, managers, and planners working in water-related industries, the climatology and variability of the rainy season's characteristics, as well as wet and dry periods, are invaluable information. The focus of future study will be on using other metaheuristic algorithms with different preprocessing techniques.

References

1. B.T. Pham, L.M. Le, T.T. Le, K.T.T. Bui, V.M. Le, H.B. Ly, I. Prakash, Development of advanced artificial intelligence models for daily rainfall prediction. *Atmos. Res.* **237**, 104845 (2020)
2. N. Mishra, H.K. Soni, S. Sharma, A.K. Upadhyay, Development and analysis of artificial neural network models for rainfall prediction by using time-series data. *Int. J. Intell. Syst. Appl.* **10**(1) (2018)
3. C. Abirami, R. Harikumar, S.S. Chakravarthy, in *Performance Analysis and Detection of Micro Calcification in Digital Mammograms Using Wavelet Features*. 2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET) (IEEE, 2016), pp. 2327–2331
4. A. De Caigny, K. Coussement, K.W. De Bock, A new hybrid classification algorithm for customer churn prediction based on logistic regression and decision trees. *Eur. J. Oper. Res.* **269**(2), 760–772 (2018)
5. T. Pranckevičius, V. Marcinkevičius, Comparison of naive bayes, random forest, decision tree, support vector machines, and logistic regression classifiers for text reviews classification. *Baltic J. Mod. Comput.* **5**(2), 221 (2017)
6. S. Georganos, T. Grippa, S. Vanhuyse, M. Lennert, M. Shimoni, E. Wolff, Very high resolution object-based land use–land cover urban classification using extreme gradient boosting. *IEEE Geosci. Remote Sens. Lett.* **15**(4), 607–611 (2018)
7. F. Climent, A. Momparler, P. Carmona, Anticipating bank distress in the Eurozone: an extreme gradient boosting approach. *J. Bus. Res.* **101**, 885–896 (2019)
8. A. Das, U.R. Acharya, S.S. Panda, S. Sabut, Deep learning based liver cancer detection using watershed transform and Gaussian mixture model techniques. *Cogn. Syst. Res.* **54**, 165–175 (2019)
9. Y. Li, W. Cui, M. Luo, K. Li, L. Wang, Epileptic seizure detection based on time-frequency images of EEG signals using Gaussian mixture model and gray level co-occurrence matrix features. *Int. J. Neural Syst.* **28**(07), 1850003 (2018)
10. S.R. Sannasi Chakravarthy, H. Rajaguru, Detection and classification of microcalcification from digital mammograms with firefly algorithm, extreme learning machine and non-linear regression models: a comparison. *Int. J. Imaging Syst. Technol.* **30**(1), 126–146 (2020)
11. S.R. Sannasi Chakravarthy, H. Rajaguru, A novel improved crow-search algorithm to classify the severity in digital mammograms. *Int. J. Imaging Syst. Technol.* **31**, 921–954 (2021). <https://doi.org/10.1002/ima.22493>
12. S.R. Sannasi Chakravarthy, H. Rajaguru, Lung cancer detection using probabilistic neural network with modified crow-search algorithm. *Asian Pac. J. Cancer Prev. APJCP* **20**(7), 2159 (2019)

Automated Hardware Recon—A Novel Approach to Hardware Reconnaissance Process



Kalpesh Gupta, Aathira Dineshan, Amrita Nair, Jishnu Ganesh, T. Anjali, Padmamala Sriram, and J. Harikrishnan

Abstract Technology is growing at an exponential rate, and everything around us from shopping to instant messaging and emails, studies, etc. is getting connected to the Internet and becoming smarter. This enabled reconnaissance (or recon) is a collection of procedures and methods, including enumeration, foot-printing, that are used to covertly discover and acquire information about a target system. The protracted recon process requires utmost attention and precision when it comes to handling the device for inspection. There exists a high risk of tampering with the device while inspecting the interior, requiring the replacement of the device. With FCC ID or chip number extraction using optical character recognition, followed by double-checking with the dataset, the specifically designed web scrapers will help to scrape all the information required, including the vulnerabilities from the web, after which a brief report will be generated. Hence, our proposed system automates the process of reconnaissance, saving time and helps in avoiding risks to an extent.

Keywords Hardware recon · Reconnaissance · OCR · Dataset · Web scraper · BeautifulSoup · Selenium · Hardware security · Datasheet

1 Introduction

The Internet is a crucial part of today's era, and from shopping to instant messaging and emails, academics, everything surrounding us is getting connected to the Internet and becoming smarter. This enables instant communication and interaction and provides simple access to information and services. To make these things smart, smart sensors, micro-controllers, and microprocessors are used in the devices. And

K. Gupta (✉) · A. Dineshan · A. Nair · J. Ganesh · T. Anjali · P. Sriram
Department of Computer Science and Engineering,
Amrita Vishwa Vidyapeetham, Amritapuri, India
e-mail: anjalit@am.amrita.edu

P. Sriram
e-mail: padmamala@am.amrita.edu

J. Harikrishnan
Cisco Systems, Bangalore, India

with the growing use of such devices, protection of our information from getting misused is must. The software needs to be protected from malware, viruses, etc., and this is achieved by using anti-virus software. And with the increase in usage of smart devices, protecting the hardware from getting compromised becomes necessary too.

The hardware consists of various microprocessors and integrated circuits, and due to its complexity, it is hard to detect vulnerabilities and fix them as replacing hardware components can be a tedious task. And, it is difficult to find the source of the vulnerability, as there is a possibility of it being a manufacturing defect.

In the Cyber Security domain, pen-testing is a process of analyzing and assessing the “secureness” of a device by doing a series of simulated attacks on the device and looking for vulnerabilities. This tells the pen-testers the flaws of the device and gives them an idea of the possible attacks. In the hardware domain, pen-testers use a process called reconnaissance.

Reconnaissance (or recon in short) is a collection of methods and processes, including scanning and enumeration, that are used to secretly uncover and collect knowledge about the target systems. There are two kinds of recon processes—active and passive reconnaissance [1]. Active reconnaissance is a type of computer intrusion in which an attacker interfaces with the targeted device to collect information about vulnerabilities [2]. Passive reconnaissance is the method of gathering information about the intended victim of a malicious hack without the target knowing what is going on [3].

Hardware reconnaissance is a process composed of various steps like device disassembly, looking at various components, port debugging like JTAG/SWD, chip identification, and information extraction from its datasheet. Before invading the target, a full understanding of the device is required, even though it is a black box during the process of pen-testing or security analysis. The recon phase helps to identify multiple components of the system so that one’s attacks can be targeted toward what one knows, including the vulnerabilities found when learning about the system.

Every electrical and electronic product with a working frequency of more than 9KHz needs to be FCC certified. The FCC stands for Federal Communications Commission. FCC regulations are designed to reduce electromagnetic interference, manage and control a range of radio frequencies to protect the normal work of telecommunications networks and electrical products [4]. Wireless devices or products with wireless transmission frequency are assigned an FCC ID. It is a unique identifier assigned to a device registered with the United States Federal Communications Commission [5]. Using the FCC ID, one can obtain photographs of the device, user manuals for the device, etc. [6].

Another source of information for getting details about the chips and micro-controller is the manufacturer’s Web sites where the information such as the datasheets, operating information is available, but it is in a very scattered manner and requires a lot of manual intervention to search the required information.

Web scraping is the practice of gathering organized web data in an automated manner. It is also called extracting web info. Any of the key applications of web scraping includes, among many others, pricing tracking, price intelligence, news monitoring, market survey, and sentiment analysis. It is also called extracting data [7–9].

A Python library to pull data from HTML and XML files is Beautiful Soup. It functions to include idiomatic ways to browse, find, and alter the parser tree using the parser. It usually saves hours or days of work for programmers. It also automatically converts incoming docs to unicode and outgoing documents to UTF-8 [10].

Selenium is a cross-platform framework based in JavaScript, Python, C#, Ruby whose development was first started by ThoughtWorks, by a person named Jason Huggins when he was building a testing application for an internal expenses and time application [11]. Selenium is predominantly used for extracting data from dynamic web pages and building automated functional tests for testing web applications.

This work proposes an easy-to-use web application-based tool which automates the hardware recon process by automatically extracting the FCC ID or the chip number of the chip from the device image and generates a brief report containing the details of the past exploits, vulnerabilities in the device, operating temperature, datasheet links, etc.

The main contribution of our work is that we are proposing a novel approach for the recon process and an extensive pre-compiled dataset. The proposed approach combines the optical character recognition with the recon process and lookup in this extensive dataset to automate the recon process, saving time and avoiding risks to an extent.

The rest of this paper is structured as follows. Some of the related existing works and approaches toward web scraping and text extraction from images are in the second section. The third section describes our proposed approach including the details of the dataset that is compiled from a variety of web resources to fasten up the recon process. The further sections describe the report generation process, benefits of the proposed system, conclusion and future works.

2 Related Works

Ray Smith describes the steps involved in the text extraction using Tesseract, including processing, recognition, and classification of the image, to extract the text character by character [12]. Nguyen et al. perform a statistical analysis the possible errors caused by the optical character reader using four different datasets [13]. Payel Roy et al. compare different algorithms by comparing the adaptive threshold values using Correlation and Structural Similarity Index (SSIM) calculations [14].

The algorithm proposed by S. Chaudhari et al. uses the web scraper tool Scrapy and MongoDB in the application. The application stores the recipe name, ingredients, and the URL of the recipe in the database, collected through web scraping beforehand [15]. Shinde Santaji Krishna and Joshi Shashank Dattatraya presented a page-level

data extraction system that extracts web page schema from template generated web pages automatically [16]. Ahmad Pouramini and Shahram Nasiri proposed a tool that generates web scrapers to extract data items from the web page. They have tried to stimulate the way humans look at web pages and have used textual anchors to create patterns for the target data regions [17]. Sanya Goel et al. propose a PHP-based web application which is able to crawl through useful information from the schools' Web site and provide aid to parents in the Delhi NCR region [18]. S. Thivaharan. et al. compared the popular web scraping libraries such as BeautifulSoup, LXML, and RegEx in terms of response time (best, average, and worst cases) and accuracy [19].

3 Proposed Solution

Our suggested approach starts with getting an input of the image of the device from the user of the device or chip under study for the vulnerabilities and exploits. Upon receiving the input, the extraction phase starts to extract the text from the input image and process it to get the FCC ID or the chip number using the OCR engine. The extracted chip number is searched in the extensive dataset to get the resource links and other details. Then, the required information is web scrapped from those resource links, while for the FCC ID, the information is directly web scrapped from the order to generate a brief report for the user containing the details of the past exploits, vulnerabilities, and other important device information. The proposed approach is summarized in Algorithms 1 and 2.

Algorithm 1: Hardware recon using the FCC ID

Input : An image of the device containing FCC ID

Output : Report containing datasheets, internal, and external images of the device

1. Preprocessing of the input image;
 2. Passing the processed image to the OCR engine to extract the FCC ID ;
 3. Extracted FCC ID is appended to the URL "www.fccid.io\" ;
 4. Web scrape all the datasheets, internal, and external images of the device;
 5. Generate a combined report;
-

The diagram displayed in Fig. 1 represents the proposed solution using FCC ID and chip number.

3.1 OCR Engine

OCR or optical character recognition is a text recognition system that recognizes and extracts text from digital documents and images. It is widely used in AI, machine

Algorithm 2: Hardware recon using chip number

Input : An image of the chip/device containing chip number
Output : Summarized report of the chip along with CVE links

1. Preprocessing of the input image;
2. Passing the processed image to the OCR engine to extract the chip number;
3. Extracted chip number is searched in the managed dataset;
4. **if** *chip number is found* **then**
 - i. Retrieve the resource links and other information from the database ;
 - ii. Web scrape device information from the Web site ;
 - iii. Generate a report from the retrieved data ;
- else**
 - Generate error message for the chip number not found.
- end**

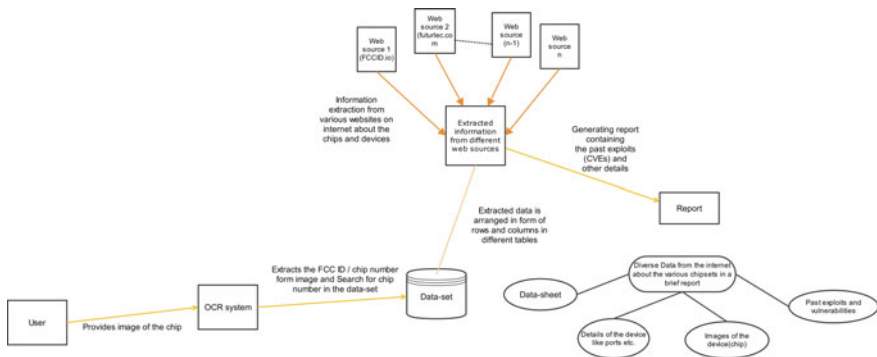


Fig. 1 Block diagram of the proposed solution for automated hardware recon

learning, robotics, IoT, banking, and health care [20]. One of the most popular and commonly used OCR engines is Tesseract. It is open sourced and identifies a wide range of languages. PyTesseract, which is the OCR tool for Python, is used for our system.

In the proposed system, the OCR engine is used for FCC ID extraction or the micro-controller chip number extraction of the scanned images, based on the user's need.

3.1.1 FCC ID Extraction

Typically, the device details are found in the form of printed text. Thus, in this case, the preprocessing involves converting the image into grayscale and increasing the contrast so as to make the text to appear more prominent [21].

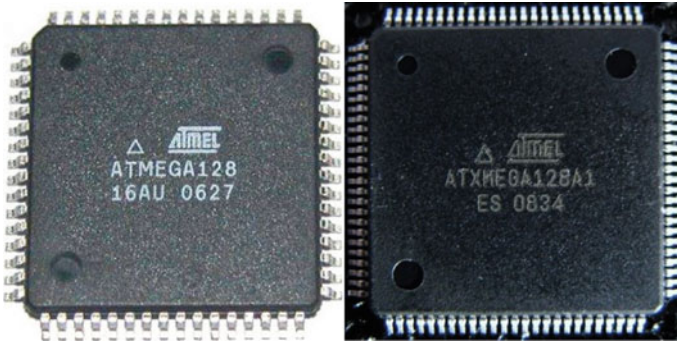


Fig. 2 Comparison between new and old micro-controller texts

3.1.2 Chip Number Extraction

In the case of micro-controllers, the text is not printed, rather etched on the mounting package of the integrated circuit. As a consequence over time, the chip number may lose its clarity (Fig. 2).

As a result, the OCR may or may not recognize the letters accurately. After several trials and errors, two different preprocessing methods which yielded better results concerning the system were proposed.

1. **Edge Detection:** In this case, the noise is removed, and edges of the letters are detected via Canny edge detection. This method works well in cases where the text engravings are not distinguished and the micro-controller surface is more smooth.
2. **Thresholding :** In cases where the letters are not defined or the edges are not very clear, the image is dilated to accentuate the letters, and this is followed by thresholding.

So, the image is passed through the above two methods, and the processed image is then fed to Tesseract to obtain the chip number. The output from the methods is then cross-checked with a predefined array which consists of common prefixes of the different micro-controller families. If a match exists, a lookup of the particular chip number is done against the dataset to obtain the corresponding resource and CVE links.

3.2 Dataset Construction

Although there are thousands of micro-controller or communication modules used today worldwide, the most commonly used can be found in various applications around us like that of Arduino (Atmega series), Microchip's PIC series (PIC 32,

Table 1 Sample of records in the dataset

Chip number	Resource link	CVE1	CVE2	CVE3
ATSAMA5D21	⟨Manufacturer site⟩	⟨CVE link 1⟩	⟨CVE link 2⟩	⟨CVE link 3⟩
ATSAM3U2C	⟨Manufacturer site⟩	⟨CVE link 1⟩	⟨CVE link 2⟩	⟨CVE link 3⟩
:	:	:	:	:
:	:	:	:	:

18, 16, 12), Microchip’s ATSAM series, ESP32 series, TI’s C2000 series, TI’s SimpleLink Series, TI’s MSP430 series, Arm’s Cortex series, Arm’s Mali GPUs.

Arduino micro-controller is used in variety of smart IoT-based systems today due to its cheaper cost and high usability. It is also used in the domestic air quality monitoring and gas leak detection systems [22]. It is also used in the IoT-based smart bins [23].

Hence, we have compiled an extensive dataset consisting of the various resource links of these commonly used micro-controllers and chip set from various sources on Internet. These resource links point to the Web sites from where we can get much information about the chip sets like the types of interface supported, chip family, DRAM interface types, micro-controller images, number of pins, whether supports GPU or not, and other chip specific information like I2C, SSC, datasheets, and other important resources about the chip set.

Besides resource links, the common vulnerabilities and exposures (CVE) links for the micro-controller chips present in the dataset (wherever available) were also included, from where one can get to know about the vulnerabilities in the smart devices and take appropriate actions to resolve them. If there are multiple CVE links available for some chips, they were included so that most appropriate information can be obtained in the brief report generated by the proposed system.

This compiled dataset is composed of the details of over 1000 most commonly used chip sets which will aid us in our proposed automated hardware recon process.

The different family of micro-controllers is arranged across different sheets to enable faster search operation. A sample of the type of record in the dataset in a sheet can be seen in Table 1.

3.3 Resource Link Lookup

Once the chip number from the OCR engine component of the proposed approach is obtained, one can lookup for it in our extensive dataset (which we mentioned above) among the families in which the chip belongs which can found using the initial part of the chip number. Once the chip number is found in the dataset, one can retrieve

the resource link from there to start the information extraction process. Also, one can retrieve the CVE links for that particular chip from the dataset.

If FCC ID is obtained from the chip, one can directly move to the data extraction part (FCC ID part) of the proposed approach.

3.4 Data Extraction

The resource links that are retrieved in the previous stage (in the case of chip number) are used to scrape the important information from the Internet. Python modules such as BeautifulSoup and Selenium framework are used in our proposed system to web scrape important information about the micro-controller chip from the different Web sites on the Internet as denoted by the resource links.

The information from the Web sites is extracted using the X-paths of the sections where the information is located on the web page using the Selenium framework. The extracted information includes the port types, number of pins, supported interfaces that can be useful for pen-testers and hardware researchers. .

For FCC ID, the information about the chip can be web scraped from fccid.io, including the internal and external images of the device under study.

4 Report Generation

The extracted information from the previous stages is compiled in a brief report in PDF format. Besides the information about the micro-controller chip and device under study, it will also include the CVE links that are retrieved from the dataset which can help the pen-testers and hardware researchers to know about the past vulnerabilities and exploits found in the micro-controller chip.

This entire process can be easily completed by the user by utilizing our web application by simply providing the image of the device, and our proposed system will perform the recon process and generate a brief report for the user.

5 Experimental Results

Hence, the proposed automated system can generate a brief report consisting of micro-controller chip features, para-metrics, CVE, and datasheet links scraped from the Internet which can aid the pen-testers and hardware researchers in reconnaissance process. This will help them to save time, focusing on other important work to enhance hardware security.

A sample snip of the report is shown in Figs. 3, 4 and 5.

ATSAM3U2C

Features:

ARM Cortex-M3 revision 2.0 running at up to 96 MHz

Memory Protection Unit (MPU)

Thumb®-2 instruction set

128 Kbytes Dual Plane embedded Flash, 128-bit wide access, memory accelerator, dual bank

36 Kbytes embedded SRAM

16 Kbytes ROM with embedded bootloader routines (UART, USB) and IAP routines

Parametrics

Name : Value

Part Family : ATSAM3U2C

Max CPU Speed MHz : 96

Program Memory Size (KB) : 128

SRAM (KB) : 36

SDIO/SD-CARD/eMMC : 1

Temperature Range (C) : -40 to 85

Operating Voltage Range (V) : 1.62 to 3.6

Direct Memory Access Channels : 21

SPI : 4

I2C : 1

Peripheral Pin Select / Pin Muxing : Yes

Number of USB Modules : 1

USB Interface : High Speed

ADC Input : 8

Max ADC Resolution (Bits) : 12

Max ADC Sampling Rate (ksps) : 1000

Input Capture : 6

Max 16-bit Digital Timers : 3

Parallel Port : EBI

Fig. 3 Sample snip from the report (Part 1)

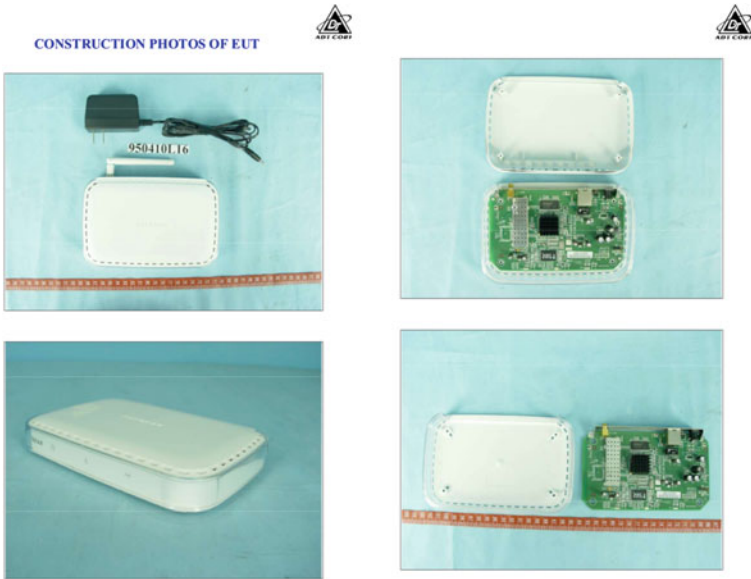


Fig. 4 Sample snip from the report (Part 2)

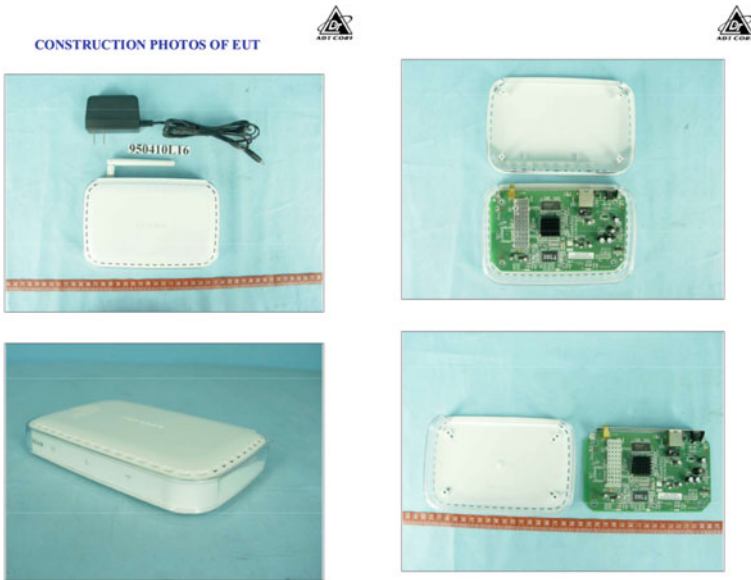


Fig. 5 Sample snip from the FCC ID report

6 Advantages

Pursuing the vision of our university, we directed our research efforts toward compassion-driven research. This proposed novel solution will be able to reduce the man-hours and efforts put into the pen-testing and reconnaissance processes. It will also help in eliminating human errors that might slip in during the manual process, in turn helping to reduce the amount of e-wastes produced due to discarding of the tampered devices. Toxic substances present in these e-wastes are lethal both to the environment and the beings; thus, this proposed automated approach will contribute to lowering it to a certain extent.

7 Conclusion and Future Work

Hence, a novel approach is proposed that automates the traditional hardware recon process using the combined vigor of OCR, our compiled dataset, and web scrapping to generate a brief report containing important information about the micro-controller chip which may act as an aid for pen-testers and hardware researchers. And, all of these can be controlled using an easy-to-use web application.

The work can be further enhanced by developing a better and efficient OCR for refining the process of FCC ID and chip number extraction. Dataset expansion by including new micro-controllers and the development of an algorithm to collect recent CVEs can contribute to make the application versatile. Research on adaptable web scrapers can help in replacing the customized web scrapers, thus saving time in developing those for newly included micro-controllers.

Acknowledgements We would like to utilize this section for expressing our gratitude toward our source of inspiration, Mata Amritanandamayi Devi, affectionately known as Amma, Chancellor of Amrita Vishwa Vidyapeetham who has devoted her life to selfless service to the society. We would also like to extend our sincere gratitude to our mentors and teachers who have always helped us whenever we faced any difficulties and motivated us to continue this work. Also, not to skip, thanks to all the people who have directly and indirectly aided us in this work.

References

1. Ethical Hacking—Reconnaissance—Tutorialspoint, in Tutorialspoint.com. https://www.tutorialspoint.com/ethical_hacking/ethical_hacking_reconnaissance.htm. Accessed 8 May 2021
2. Active Versus Passive Reconnaissance—ASM, Rockville, Maryland, in ASM, Rockville, Maryland. <https://asmed.com/active-vs-passive-reconnaissance/>. Accessed 8 May 2021
3. Passive Versus Active Reconnaissance, in *Medium*. <https://medium.com/@jharve08/passive-vs-active-reconnaissance-c2974913237f>. Accessed 8 May 2021

4. What is the FCC-ID authentication? Huawei Enterprise Support Community, in *Huawei Enterprise Support Community*. <https://forum.huawei.com/enterprise/en/what-is-the-fcc-id-authentication/thread/588944-883>. Accessed 8 May 2021
5. Fccid, in Fccid.io. <https://fccid.io/>. Accessed 8 May 2021
6. A. Dineshan, G. Gokul Krishna, J.L.A. varshini, J. Ganesh, T. Anjali, J. Harikrishnan, Hardware security reconnaissance application using FCC ID lookup and computer vision, in *2020 International Conference on Communication and Signal Processing (ICCSP)*, 2020, pp. 526–529. <https://doi.org/10.1109/ICCSP48568.2020.9182318>
7. What is web scraping and how does it work? Zyte.com, in Zyte (formerly Scrapinghub) #1 Web Scraping Service. <https://www.scrapinghub.com/what-is-web-scraping/>. Accessed 8 May 2021
8. T. Anjali, T.R. Krishnaprasad, P. Jayakumar, A novel sentiment classification of product reviews using Levenshtein distance. *Int. Conf. Commun. Signal Process. (ICCSP)* **2020**, 0507–0511 (2020). <https://doi.org/10.1109/ICCSP48568.2020.9182198>
9. A.C. Jishag, et al., Automated review analyzing system using sentiment analysis, in *Ambient Communications and Computer Systems. Advances in Intelligent Systems and Computing*, vol. 904, eds. by Y.C. Hu, S. Tiwari, K. Mishra, M. Trivedi (Springer, Singapore, 2019). https://doi.org/10.1007/978-981-13-5934-7_30
10. L. Richardson, Beautiful Soup: We called him Tortoise because he taught us, in *Crummy.com*. <https://www.crummy.com/software/BeautifulSoup/>. Accessed 8 May 2021
11. Selenium history, in Selenium.dev. <https://www.selenium.dev/history/> Accessed 8 May 2021
12. R. Smith, An overview of the Tesseract OCR engine, in *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, Parana, 2007, pp. 629–633. <https://doi.org/10.1109/ICDAR.2007.4376991>
13. T. Nguyen, A. Jatowt, M. Coustaty, N. Nguyen, A. Doucet, Deep statistical analysis of OCR errors for effective post-OCR processing, in *2019 ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, Champaign, IL, USA, 2019, pp. 29–38. <https://doi.org/10.1109/JCDL.2019.00015>
14. P. Roy, S. Dutta, N. Dey, G. Dey, S. Chakraborty and R. Ray, Adaptive thresholding: a comparative study, in *2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*, Kanyakumari, 2014, pp. 1182–1186. <https://doi.org/10.1109/ICCICCT.2014.6993140>
15. S. Chaudhari, R. Aparna, V.G. Tekkur, G.L. Pavan, S.R. Karki, Ingredient/recipe algorithm using web mining and web scraping for smart chef, in *2020 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*, Bangalore, India, 2020, pp. 1–4. <https://doi.org/10.1109/CONECCT50063.2020.9198450>
16. S.S. Krishna, J.S. Dattatraya, Schema inference and data extraction from templated Web pages, in *2015 International Conference on Pervasive Computing (ICPC)*, Pune, 2015, pp. 1–6. <https://doi.org/10.1109/PERVASIVE.2015.7087084>
17. A. Pouramini, S. Nasiri, Web data extraction using textual anchors, in *2015 2nd International Conference on Knowledge-Based Engineering and Innovation (KBEL)*, Tehran, 2015, pp. 1124–1129. <https://doi.org/10.1109/KBEL.2015.7436204>
18. S. Goel, M. Bansal, A.K. Srivastava, N. Arora, Web crawling-based search engine using python, in *3rd International Conference on Electronics, Communication and Aerospace Technology (ICECA)*. Coimbatore, India, 2019, pp. 436–438 (2019). <https://doi.org/10.1109/ICECA.2019.8821866>
19. S. Thivaharan, G. Srivatsun, S. Sarathambekai, A survey on python libraries used for social media content scraping, in *2020 International Conference on Smart Electronics and Communication (ICOSEC)*, Trichy, India, 2020, pp. 361–366. <https://doi.org/10.1109/ICOSEC49089.2020.9215357>
20. D. Kanteti, D.V.S. Srikar, T.K. Ramesh, Intelligent smart parking algorithm. *International Conference On Smart Technologies For Smart Nation (SmartTechCon)* **2017**, 1018–1022 (2017). <https://doi.org/10.1109/SmartTechCon.2017.8358524>

21. V. Bharath, N.S. Rani, A font style classification system for English OCR, in *2017 International Conference on Intelligent Computing and Control (I2C2)*, 2017, pp. 1–5. <https://doi.org/10.1109/I2C2.2017.8321962>
22. K. Gupta, G. Gokul Krishna, T. Anjali, An IoT based system for domestic air quality monitoring and cooking gas leak detection for a safer home, in *2020 International Conference on Communication and Signal Processing (ICCSP)*, 2020, pp. 0705–0709. <https://doi.org/10.1109/ICCSP48568.2020.9182051>
23. A. Praveen, R. Radhika, M.U. Rammohan, D. Sidharth, S. Ambat, T. Anjali, IoT based Smart Bin: a Swachh-Bharat initiative. *International Conference on Electronics and Sustainable Communication Systems (ICESC)* **2020**, 783–786 (2020). <https://doi.org/10.1109/ICESC48915.2020.9155626>

Randomised Analysis of Backtracking-based Search Algorithms in Elucidating Sudoku Puzzles Using a Dual Serial/Parallel Approach



Pramika Garg , Avish Jha , and Kumar A. Shukla 

Abstract Sudoku is a 9×9 grid-based puzzle. It is a game where each row, column, and 3×3 box must have one instance of a number from 1 to 9. In present paper, we shall evaluate three different algorithmic approaches both in serial and parallel configurations that can be utilised to solve a puzzle of Sudoku to assess their comparative performance metrics for differential randomly generated Sudoku datasets. We shall utilise Breadth-first search, Depth-first search, Depth-first search with Breadth-first search parallelisation for sub-tress, for evaluating a large number of randomly generated Sudoku puzzles with a varying number of clues to find the best algorithm based on time and space complexity as well as clue complexity. With this, we shall analyse and develop a best practice algorithm that can be ideally used to solve a large number of puzzles in any given situation in the most time-efficient manner. Our analysis has found that there was a significant improvement in utilising the parallel algorithm over both the Breadth-first and Depth-first search approaches from 28% to over 56%. Even moving from Breadth-first to Depth-first search, we have gauged quite a moderate improvement in performance from 15 to 21%.

Keywords Sudoku · Algorithms · Breadth-first search · BFS · Backtracking · Depth-first search · DFS · Efficient sudoku solving · Parallel sudoku · Randomly generated dataset · Randomised analysis

1 Introduction

Sudoku puzzle is a logic-based game that was first mentioned in the number place [1]. It was adopted by the Japanese as “Suuji Wa Dokushin Ni Kagiru”, and later on, the name was shortened down to Sudoku in 2005, leading to widespread popularity. It consists of a grid that is divided into squares. It consists of varying sizes ranging from 4×4 grids to 16×16 grids (Fig. 1).

P. Garg (✉) · A. Jha · K. A. Shukla
SCOPE, Vellore Institute of Technology, Vellore, TN 632014, India
e-mail: anik912345699@gmail.com

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
S. Smys et al. (eds.), *Inventive Computation and Information Technologies*, Lecture Notes in Networks and Systems 336, https://doi.org/10.1007/978-981-16-6723-7_21

281

1			2
		1	3
		2	

(a) A 4x4 Sudoku.

		4	7	3	6	8	5	
			2		4	1	6	7
		7	1				4	2
3	4	9	8		1	5		
		6	5		3	2	9	
	2	5	9		7			
7			6				2	
5	9	1	4	7	2	6	8	3
4	6		3	8	9		1	5

(b) A 9x9 Sudoku.

Fig. 1 Unsolved Sudoku's of varying sizes

It can be generalised onto an $N \times N$ grid where the square root of N is a positive integer. However, the Sudoku containing 9×9 grids is considered to be conventional due to the large prevalence in its usage amongst people around the world. Unsystematic numbers are assigned to the board in prior, where the ultimate goal is to fill off the remaining empty spaces. There must be a one-of-a-kind solution, with no repetitions permitted. Certain sets of rules and regulations are required to be followed by every player which includes the following (for a 9×9 grid) (Fig. 2):

- Only numbers containing integer values from 1 to 9 can be used.
- All the nine squares must contain different numbers each within a larger 3×3 box.
- Each column or row should also contain all the numbers exactly once.
- All the squares must be filled.
- All the numbers should only be from 1 to 9.

The boxes are a set of nine 3×3 smaller grids that form the bigger grid. The main principle of solving the game involves solving it by unravelling the squares with values without breaking any of the rules of the game. The initial clues given in the puzzle allow the game to have a start point. A Sudoku puzzle with 17 or more clues (clues imply filled in squares) will have a unique solution [2].

2 Related Works

The Sudoku is an NP complete problem which allows for quite a diverse variety of solutions. We have analysed multiple approaches used previously in the upcoming text.

	3	7			9	2		
	8	1		4			5	
5	9				8		7	3
		6		9		7		4
	2		7				1	5
	7					8	2	
7	4		9				8	
9		3	8	2	1		6	
	6		4					2

(a) Unsolved Sudoku

6	3	7	1	5	9	2	4	8
2	8	1	3	4	7	9	5	6
5	9	4	2	6	8	1	7	3
8	1	6	5	9	2	7	3	4
4	2	9	7	8	3	6	1	5
3	7	5	6	1	4	8	2	9
7	4	2	9	3	6	5	8	1
9	5	3	8	2	1	4	6	7
1	6	8	4	7	5	3	9	2

(b) Solved Sudoku

Fig. 2 9 × 9 Sudoku problem and solution

In [3], the authors use a stochastic search-based algorithm that works with underlying meta-heuristic techniques. They also utilise an simulated annealing-based algorithm which uses a set of Markov chains. They have also proved the existence of easy-hard-easy phase transitions in Sudoku puzzles of larger grid sizes such as 16 × 16 and 25 × 25.

The authors in [4] propose a solution based on constraint programming, treating the Sudoku as a constraint satisfaction problem. They focus on both the enumeration strategies of variable selection heuristics as well as value selection heuristics, recognising their impact on performance. In variable selection heuristics, they cover both the static and dynamic selection heuristics parameters, and in value selection heuristics, they cover smaller, greater, average, and just greater than average value of domain to gauge the best performing parameters. Even in [5], the author Simonis H. uses a constraint programming-based approach implements versions of “Swordfish” and “X-Wing” which use complex propagation schemes in a more redundant nature with well-defined constraints such as coloured matrix and defined cardinality. This leads to a stronger model which they improve on further by using flow algorithms with bipartite matching.

Using State Space Search is another method of implementing a Sudoku solver as [6] explores using both Breadth-first search and Depth-first search. It finds that Depth-first search is more optimal for blind searching, and it can be implemented using a backtracking-based approach. They compared a brute force approach using both Breadth-first search and Depth-first search. They also implemented both algorithms using a tree pruning technique which essentially is backtracking.

Backtracking is an algorithmic approach to solve a constraint satisfaction problem which was approached as a Depth-first traversal of a search tree using varied branching techniques as described in detail in [7]. In [8], both the authors have explored on the different forms of backtracking-based search algorithm and compared it against other approaches such as rule-set or Boltzmann machine. Hence, there has been

research in the utilisation of these search algorithms, but all of these have been serial approaches, and there has been none in a parallel methodology. This is what our research aims to fill the gap in since parallel processing is a major optimization as thread counts on central processing units keep on increasing.

3 Methodology

3.1 Challenges Involving Bias

The challenges will involve minimising the time as well as space complexity, due to the nature of the Sudoku as well as the algorithms in question [9]. Sudoku is well-known to be an NP complete problem and backtracking-based heuristics algorithms such as backtracking-based Depth-first search (DFS) and Breadth-first search (BFS) provide an excellent methodology to prepare an efficient solution [10].

In such a problem where we have a given Sudoku problem and we need to assess and determine a unique solution (we will analyse Sudoku puzzles with 17 or more given clues in present paper), we have to use search algorithms to perform a potential solution on the puzzle set. A random set of 1000 Sudoku puzzles shall be generated via a process that first involves generating a full Sudoku solution and then removing numbers as necessary to create a problem with a prior specified number of clues. This process will be repeated 1000 times to generate a random dataset of Sudoku puzzles with X number of clues, where X may be an integer from 25 to 30.

3.2 Solution to the Challenges

This will ensure that we have a large set ensuring any bias if at all can be eliminated via the large size of the trial and will allow the proper comparative analysis of the four algorithms in both their serial and parallel implementations. For each X from 25 to 30, the dataset shall be tested to all the algorithms, and the analysis will be done in such a way as to see if any algorithm outpaces another based on the number of clues given to the algorithm at the start. This will allow in future an algorithm to be developed that utilises the best algorithm amongst these for each specific situation based on the input size, ensuring an efficient solution can be employed. This may lead one to believe that the more the number of clues, the easier it is to solve, but it is far from the truth, as it is highly situational. A puzzle with 25 clues may indeed be harder to solve than one with 17 [11].

3.3 Implementation

In proposed algorithm, we shall employ the backtracking-based Breadth-first search algorithm in a serial configuration, and we will analyse the performance of the algorithm over a random dataset of size 1000 with varying X (25–30). In a similar manner, the same process will be repeated for the backtracking-based Depth-first search to determine the best possible configuration that works for each test case (where a test case is defined is when X has a specific value from 25 to 30). Then, the Depth-first search with a parallel sub-tree Breadth-first search algorithm shall be used to determine and analyse the improvement that is brought by utilising a parallel core architecture compared to a serialised solution. We have utilised a graph data structure based on constraint graph structure for the final code.

3.4 Specification of Test Platform

For conducting a test with no other factors playing a role in the results, we decided to approach this with a brand new fresh copy of Windows 10 LTSC, Python 3.8. The code was entirely written in Python, and it was set to utilise a single virtual thread for both the serial Breadth-first search and Depth-first search implementation, whereas the parallel algorithm utilised six virtual threads. Each test ran within its virtual environment, with identical system utilisations. Each test was run at an interval of 1 h before the previous to allow for heat dispensation on the Intel Eighth-generation CPU. Each test ran including the generation of 1000 different random Sudoku puzzles of the given clue number.

4 Serial Breadth-first Search

4.1 What Is It?

Breadth-first search is a crucial search algorithm consisting of data structures for graphical representation. It is heavily utilised for traversing graph or tree-based structures. It is an uninformed graph search strategy that moves from level to level ensuring that each level is fully searched before moving on to the next deeper level [12]. It traverses a graph structure in a breath-ward direction starting from the first vertex and follows the below-mentioned rule-set strictly.

In current situation where we implement Breadth-first search in Sudoku, we will take the first position on the top-left as the initial or first vertex. Each square is represented as a vertex with two possible states, visited and unvisited. A queue is then used by the algorithm as it executes, as mentioned below.

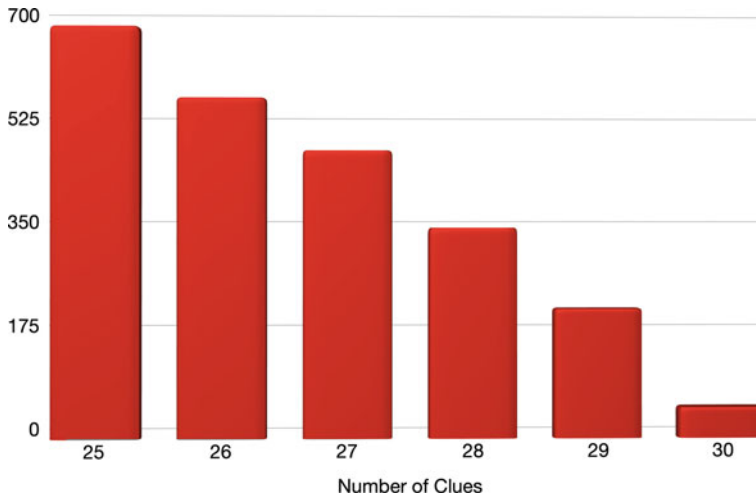


Fig. 3 BFS—average mean time of 100 random puzzles (ms) in Y—number of clues in X

4.2 Algorithm

1. From the current vertex, find all the adjacent unvisited vertexes.
2. Visit each vertex as found from Step 1 and add it to a queue.
3. If the current vertex has no adjacent vertex, then dequeue the first vertex from the queue.
4. Repeat the first three steps till all the vertices have been visited.

4.3 Results

The algorithm is implemented serially utilising the above rule-set, and the following are the results after 1000 randomly generated Sudoku are processed for each of the initial numbers of clues from 25 to 30 (Fig. 3).

5 Serial Depth-first Search

5.1 What Is It?

Depth-First Search (also known as Depth-First Traversal) is a graph-cum-tree traversal and search algorithm that works on the Artificial Intelligence algorithmic approach of Backtracking [13]. The algorithm starts from a root node and moves

down one of the branches fully till it reaches the end, if the answer is found along the way, then it stops, else it utilises the backtracking methodology. It continues this till the answer is found, with the eventual goal of visiting all the unvisited vertices. The backtracking methodology can be implemented via a recursive approach to allow for better time and space complexity [14].

In the current implementation of Depth-First Search to port it to Sudoku, we will take the first position on the Top-Left as the initial or first vertex. Each square is represented as a vertex in the algorithm with 2 possible states, visited and unvisited. A queue is then utilised by the DFS algorithm whilst it executes, as mentioned below.

5.2 Algorithm

1. Take the first position in the Sudoku and put it as the starting vertex and place it onto a stack.
2. Mark the vertex on the top of the stack as visited.
3. Find all the adjacent vertices from the current vertex and store them in a list.
4. Add the unvisited ones from the list to the stack.
5. Repeat Steps 2 to 4 till all vertices are visited.

5.3 Results

The algorithm is created in a serial configuration utilising the above rule-set, and below-mentioned are the results after 1000 randomly generated Sudoku are processed for each of the initial numbers of clues from 25 to 30 (Fig. 4).

6 Parallel Dual DFS and BFS-based Algorithm

6.1 What Is It?

Parallel Depth-first search with sub-tree Breadth-first search is an approach that involves setting a search depth till which a serial Depth-first search will be run and after which a number of Breadth-first search threads shall be spawned which will each have a sub-queue as well as access to a global queue to prevent multiple copies of the Sudoku needing to be created which would require much more memory leading to higher space complexity. The Breadth-first search threads run in parallel till the unique solution for the Sudoku is found.

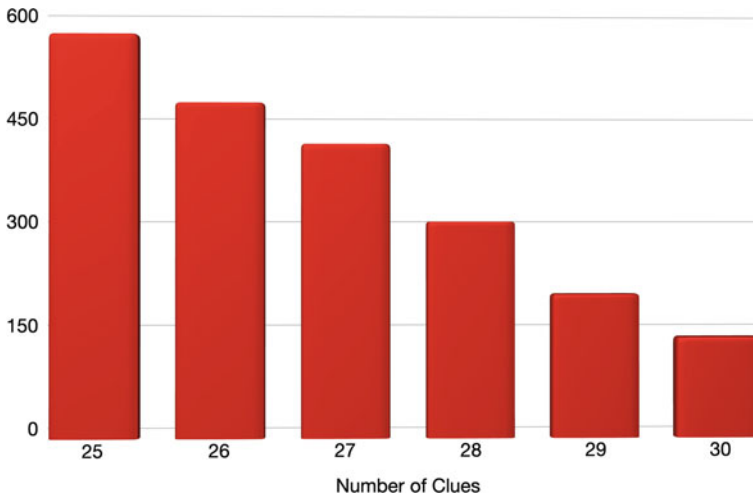


Fig. 4 DFS—average mean time of 100 random puzzles (ms) in Y—number of clues in X

6.2 Algorithm

1. A search depth is chosen.
2. Initially, Depth-first search is run till the chosen depth serially with a similar rule-set as formerly given.
3. From the search depth, a set of parallel threads is spawned which run at the same time with a global queue to solve till the end of the Sudoku is reached.
4. The Breadth-first search nodes run in parallel utilising sub-queues and the global queue.

6.3 Results

The algorithm is created in a parallel configuration utilising the above rule-set so that till the pre-defined search depth, Depth-first search is run serially after which it spawns multiple Breadth-first search nodes running in parallel for the rest of the depth of the Sudoku, and below-mentioned are the results after 1000 randomly generated Sudoku are processed for each of the initial numbers of clues from 25 to 30 (Fig. 5).

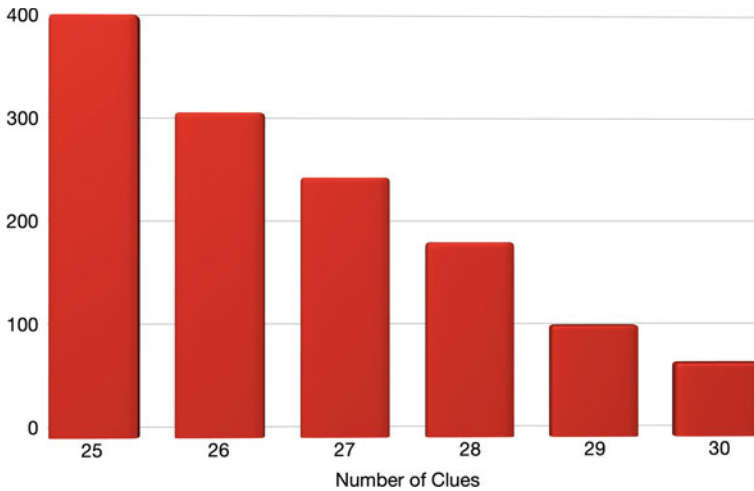


Fig. 5 Parallel mixed DFS and BFS—average mean time of 100 random puzzles (ms) in Y—number of clues in X

7 Visualisations and Analysis

Utilising the large mirage of data we collected from our testing, we can have two different outlooks, one based on the individual clues for each algorithm and an overall analysis where we average out the performance across different numbers of clues.

7.1 *Twenty-five Clues*

When we take the initial number of clues to be 25, it can be analysed that moving from serial Breadth-first search to serial Depth-first search, we see an improvement of over 15.86826%. Now, if we compare the parallel dual Depth-first search with Breadth-first search with the serial Breadth-first search algorithm, we mark an improvement on 41.16766% which is quite significant. Moving from serial Depth-first search to the parallel algorithm, we see an improvement of about 28.38926% (Fig. 6).

7.2 *Twenty-six Clues*

When the number of clues is taken to be 26, observations state an increase of 15.12915%, whilst moving from serial Breadth-first search to serial Depth-first search. In addition to that, the parallel version of Depth-first search and Breadth-first search shows a notable increase of 44.83394% over serial Breadth-first search,

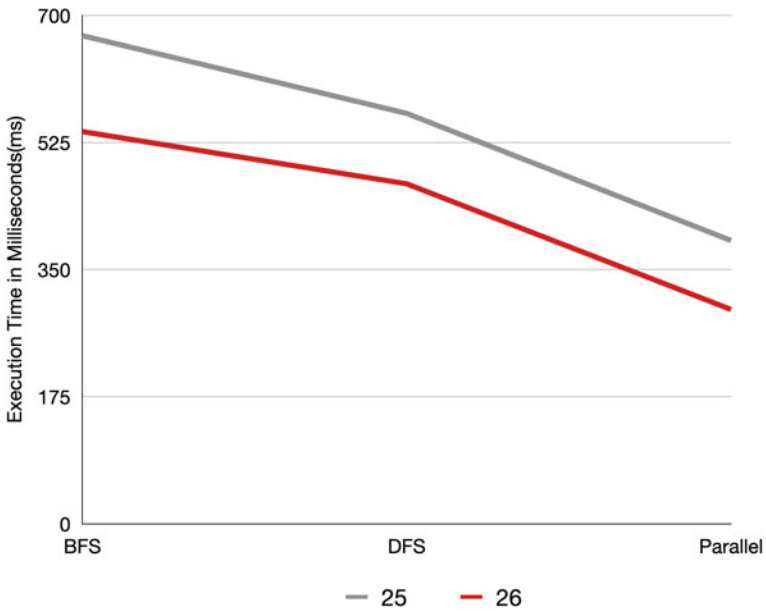


Fig. 6 Comparing the performance of 25 versus 26 initial clues

which concludes that the performance of the paralleled implementation of Depth-first search and Breadth-first search is higher than the serial implementation of Breadth-first search. When comparing the parallel dual Depth-first search with Breadth-first search method to the serial Depth-first search algorithm, enhancement of 35.15724% is beheld.

7.3 *Twenty-seven Clues*

Analysing the bar plot for 27 clues, we see an improvement of about 18.73922% continuing the pattern of serial Depth-first search being more efficient than serial Breadth-first search. In comparison to the previous analysis, the drop from serial Depth-first search to the parallel algorithm is on the higher end at 51.34924%, whilst the gain in performance moving from the serial Depth-first search to the parallel algorithm stays around 38.47207% (Fig. 7).

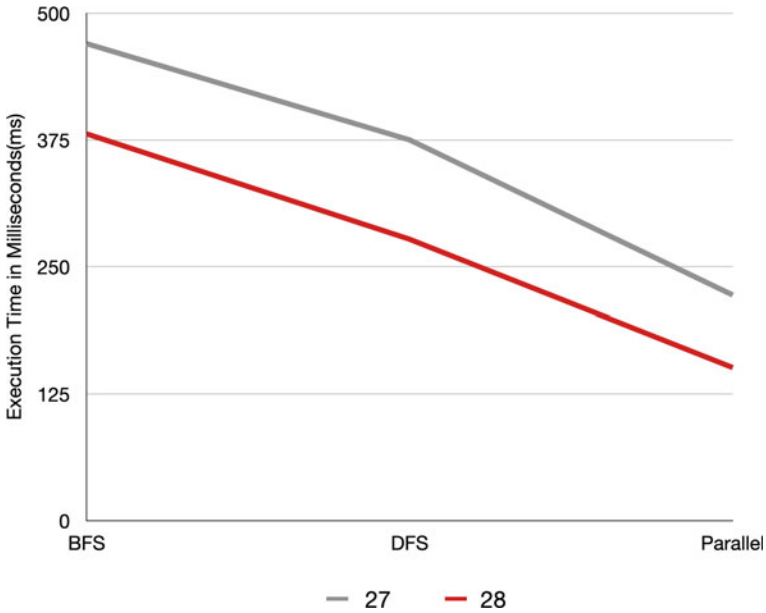


Fig. 7 Comparing the performance of 27 versus 27 initial clues

7.4 Twenty-eight Clues

In case of 28 clues, we see a gain in performance of approximately 21.47925% moving from serial Breadth-first search to serial Depth-first search. If we switch from the serial Breadth-first search algorithm to the dual parallel algorithm, we see a gain of over 52.64389% in performance. Whereas moving from serial Depth-first search to the dual parallel algorithm, we see an improvement of 39.82891%.

7.5 Twenty-nine Clues

Starting off with 29 clues, we are able to analyse that moving from serial Breadth-first search to serial Depth-first search, we get an improvement of about 17.57393%, but there is a major improvement with over the doubling of speed moving from serial Breadth-first search to the dual parallel algorithm with over 53.73929% improvement. The movement from serial Depth-first search to the dual parallel algorithm is again a pretty sharp increase in performance by about 41.32907% which is anomalously higher than the previous cases (Fig. 8).

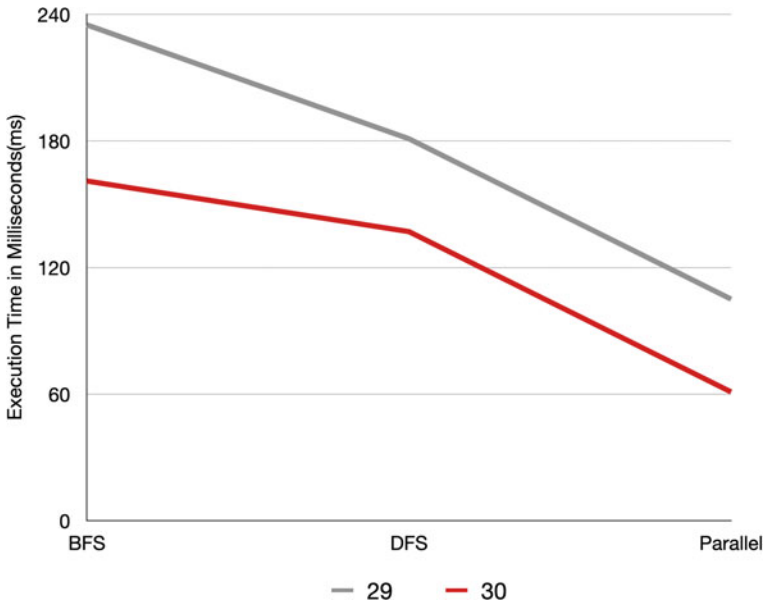


Fig. 8 Comparing the performance of 29 versus 30 initial clues

7.6 Thirty Clues

In this section, we analyse the performance where 30 clues are given initially, and we see an improvement of 19.28571% jumping from serial Breadth-first search to serial Depth-first search. Moving from the serial Breadth-first search to the parallel algorithm, we see a large improvement of over 56.38127% which confirms that this parallel algorithm continues to become more and more efficient as the number of clues is increased. The jump from serial Depth-first search to the parallel algorithm is quite on the higher end at 35.82937% suggesting that the parallel algorithm starts to gain an advantage over the serial Depth-first search algorithm with an increasing number of clues (Figs. 9 and 10).

8 Conclusion

With the averaging of the values as shown in Figs. 3, 4 and 5, we can see that there is an improvement of approximately 20.79027% as we utilise backtracking-based Depth-first search over Breadth-first search which can be attributed to the benefits that backtracking brings to Depth-first search, and hence, it can be determined that a serial implementation of Depth-first search will uniformly perform better than a serialised implementation of Breadth-first search (Fig. 11; Table 1).

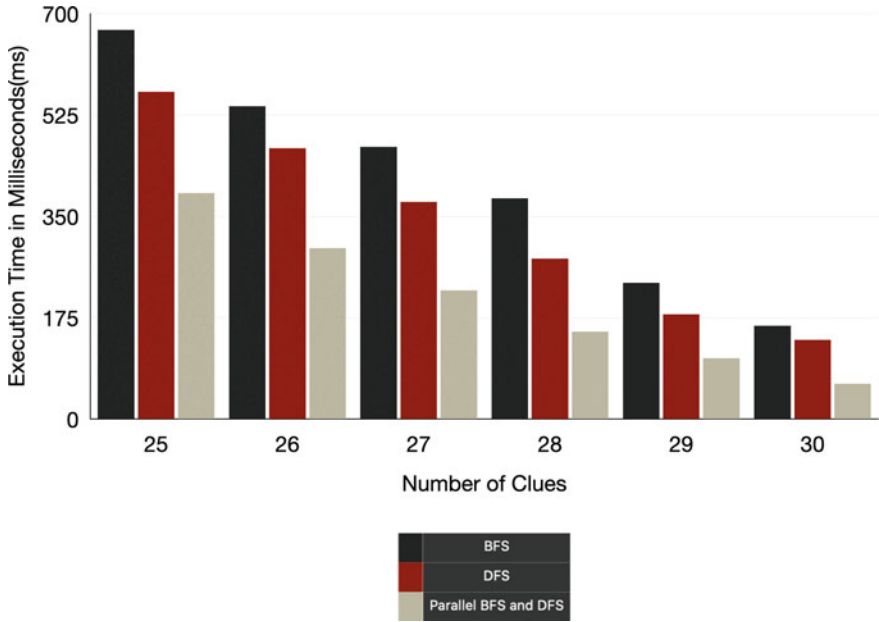


Fig. 9 Overall summary of execution time

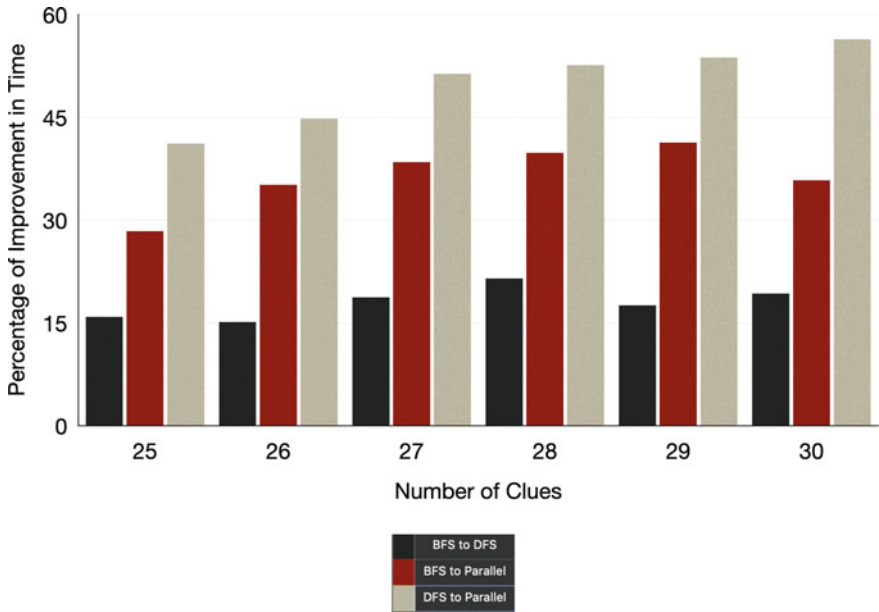


Fig. 10 Comparison of percentage of improvement

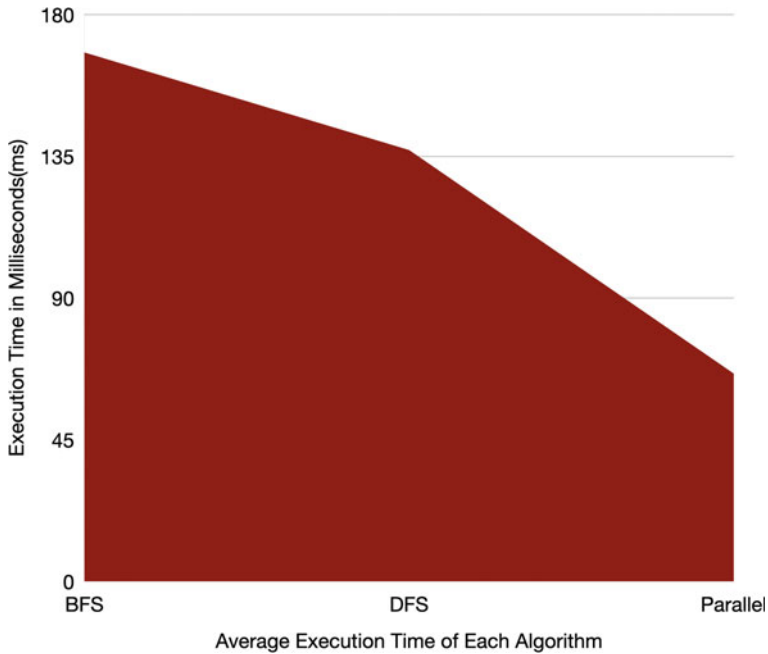


Fig. 11 Average execution time of the three algorithms

Table 1 Percentage gain as we move from Algorithm A to Algorithm B for all three possible permutations

Percentage improvement			
Number of clues	BFS to DFS (%)	BFS to parallel (%)	DFS to parallel (%)
25	15.86826	41.16766	28.38926
26	15.12915	44.83394	35.15724
27	18.73922	51.34924	38.47207
28	21.47925	52.64389	39.82891
29	17.57393	53.73929	41.32907
30	19.28571	56.38127	35.82937

From the previous set of analysis at individual clue levels, we can notice that the performance gap between serial Breadth-first search and serial Depth-first search is approximately 20% constantly irrespective of the change in the initial number of clues. The major improvement occurs in the switch from serial Depth-first search to the dual parallel Depth-first search with Breadth-first search algorithm which ensues a constant performance gain with the increase in the number of clues, from about 28% to over 35%.

The parallel version of Depth-first search utilising Breadth-first search for its subtrees is a major improvement over serial Breadth-first search and even over serial

Depth-first search. It manages to uphold a humongous 39.35643% over the serial Breadth-first search implementation. Though, it is to be noted that this was with six parallel Breadth-first search threads that were launched after the pre-defined search depth has been reached. Even though utilising Breadth-first search partially in this algorithm would be expected to offer a slowdown compared to the backtracking-based highly performant serial Depth-first search, the threaded Breadth-first search absorbs the complexity increase due to parallelisation, and it still comes out ahead majorly of the serial Depth-first search with a 23.43783% improvement.

Overall, we can see there is an immense improvement in the real-world performance of our algorithm even though the time complexity is rather unchanged from that of serial Depth-first search and serial Breadth-first search. The novelty of our approach is adding the concept of parallelisation on not just a single algorithm rather combining two algorithms in such a way that not only do, we reap the benefits of parallelisation, and we also get the best from both the algorithms. Hence, we can conclude that such a parallel approach is the best algorithm amongst the studied to solve a given Sudoku puzzle.

References

1. H. Garns, *Number Place* (Dell Pencil Puzzles & Word Games, 1979)
2. J.P. Delahaye, The science behind Sudoku. *Sci. Am.* **294**(6), 80–87 (2006)
3. R. Lewis, Metaheuristics can solve sudoku puzzles. *J. Heuristics* **13**(4), 387–401 (2007)
4. B. Crawford, M. Aranda, C. Castro, E. Monfroy, Using constraint programming to solve sudoku puzzles, in *2008 Third International Conference on Convergence and Hybrid Information Technology*, vol. 2 (IEEE, 2008), pp. 926–931
5. H. Simonis, Sudoku as a constraint problem, in *CP Workshop on Modeling and Reformulating Constraint Satisfaction Problems*, vol. 12 (Citeseer, 2005)
6. M. Eremic, R. Adamov, N. Bilinac, V. Ognjenovic, V. Brtko, I. Berkovic, Comparison of state space search algorithms-sudoku puzzle game. *Chief Responsible Editor* **154** (2013)
7. P. Van Beek, Backtracking search algorithms. *Found. Artif. Intell.* **2**, 85–134 (2006)
8. P. Berggren, D. Nilsson, *A Study of Sudoku Solving Algorithms* (Royal Institute of Technology, Stockholm, 2012)
9. M.J. Pathak, R.L. Patel, S.P. Rami, Comparative analysis of search algorithms. *Int. J. Comput. Appl.* **179**(50), 40–43 (2018)
10. T. Yato, T. Seta, Complexity and completeness of finding another solution and its application to puzzles. *IEICE Trans. Fund. Electron. Commun. Comput. Sci.* **86**(5), 1052–1060 (2003)
11. A study of Sudoku solving algorithms. <https://cutt.ly/zb2GfyN>. Last accessed 18 May 2021
12. A. Bundy, L. Wallen, Breadth-first search, in *Catalogue of Artificial Intelligence Tools* (Springer, Berlin, Heidelberg, 1984), p. 13
13. L. Chen, Y. Guo, S. Wang, S. Xiao, K. Kask, *Sudoku Solver* (2016)
14. H.S. Stone, P. Sipala, The average complexity of depth-first search with backtracking and cutoff. *IBM J. Res. Dev.* **30**(3), 242–258 (1986)
15. Multi-threaded algorithm for solving Sudoku. <https://stackoverflow.com/a/850892>. Last accessed 18 May 2021

High Speed VLSI Architecture Design Using FFT for 5G Communications



P. Lakshmi Devi, Somashekhar Malipatil, and P. S. Surekha

Abstract A high speed FFT processor is designed supporting 16- to 4096-point FFTs and 12- to 2400-point DFTs for 5G, WLAN. The processor is designed for high speed applications and source code is written in Verilog. Synthesis and simulation is done in Xilinx ISE 14.7. The power dissipation is minimized (20.3 mW) and delay is 9.539 ns and further extension is done using CORDIC processor delay is 7.55 ns. In this paper, high speed VLSI Architecture designed using FFT for 5G Communications. The proposed results are compared with the existed work.

Keywords Coordinate rotation digital computer (CORDIC) · FFT · DFT · Xilinx ISE 14.7 · Verilog · 5G · VLSI

1 Introduction

FFT is a compute-intensive algorithm in the physical layer of an OFDM system to convert data between time domain and frequency domain. Many OFDM systems such as 4G LTE/LTE-A and WLAN require power of two FFTs. LTE uplink preceding requires non power of two DFTs from. In the upcoming 5G, FFT is still an essential algorithm for all of the waveform candidates, and the FFT computation speed should be high enough to support the high data rate of 5G. Furthermore, the decomposition algorithm employs a technique known as high-radix–small-butterfly to reduce computation cycles and simplify the processing engine [1]. To build a low-power FFT processor, the Radix-2m-bits encoding scheme was used to minimize the partial product lines in the multiplication [2].

P. Lakshmi Devi
Department of Electronics and Communication Engineering, St. Peeter's Engineering College (A), Misammaguda, Kompally, Hyderabad, Telangana, India

S. Malipatil (✉) · P. S. Surekha
Department of Electronics and Communication Engineering, Malla Reddy Engineering College and Management Sciences, Medchal, Telangana, India

2 Proposed Work

The structure of proposed FFT processor as shown in Fig. 1 the proposed FFT processor consists of butterfly unit, scaling unit, CORDIC unit, aligning unit and data memory with two 16 bank 28 bit single port memory. The 16 × 28 bit input and output data ports. The TF multiplications takes place in CORDIC unit.

Butterfly unit in processing element as shown in Fig. 2. It consists of processing element PE-A, PE-B and 2 PE-C along with switch networks. To increase reuse of the hardware resources, the radix-16 DFT is divided into 2.

$$X_{16}(k_1, k_2) = \sum_{n_2=0}^3 \left[\underbrace{W_{16}^{n_2 k_1}}_B \sum_{n_1=0}^3 \underbrace{\left(x_{16}(n_1, n_2) W_4^{n_1 k_1} \right)}_A W_4^{n_2 k_2} \right]$$

where A and B are implemented in the PE-A, PE-B hardware units in Fig. 3 (Fig. 4).

$$X_8(k_1, k_2) = \sum_{n_2=0}^1 \left[\underbrace{W_8^{n_2 k_1}}_B \sum_{n_1=0}^3 \underbrace{\left(x_8(n_1, n_2) W_4^{n_1 k_1} \right)}_A W_4^{n_2 k_2} \right].$$

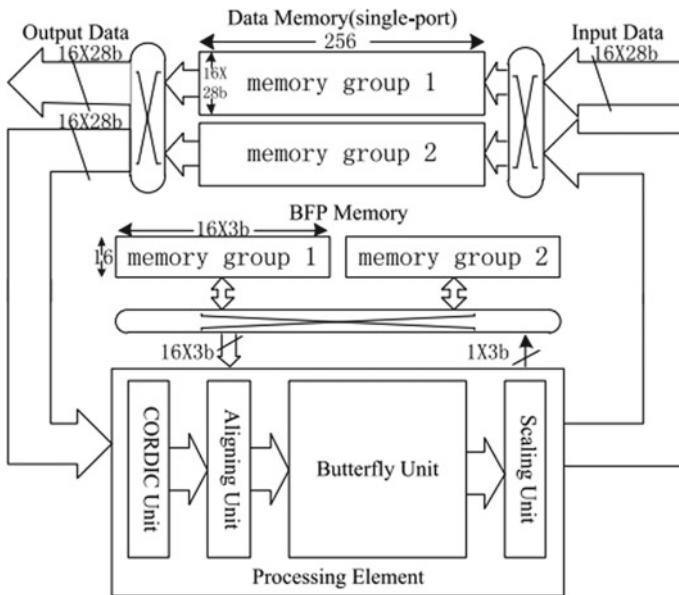


Fig. 1 Proposed FFT processor

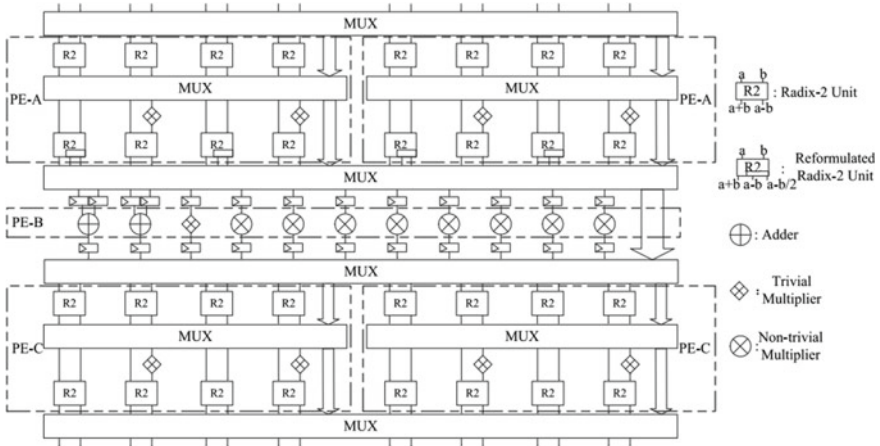


Fig. 2 Butterfly unit in processing element

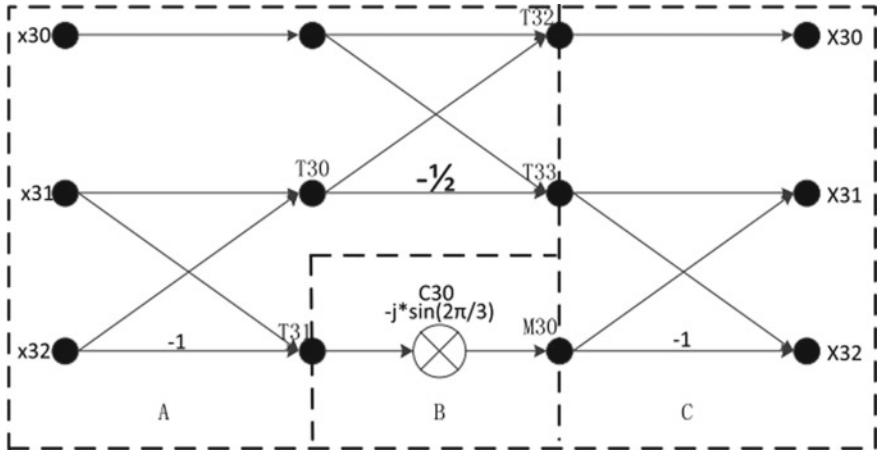


Fig. 3 Signal flow 3point DFT algorithm

3 Simulation and Synthesis Results

See Table 1.

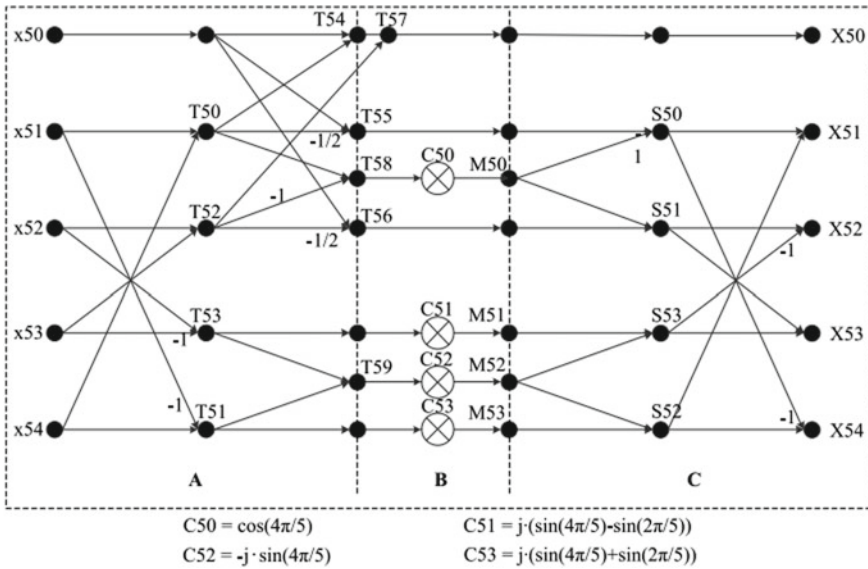


Fig. 4 Signal flow graph 5point DFT algorithm

Table 1 Power analysis

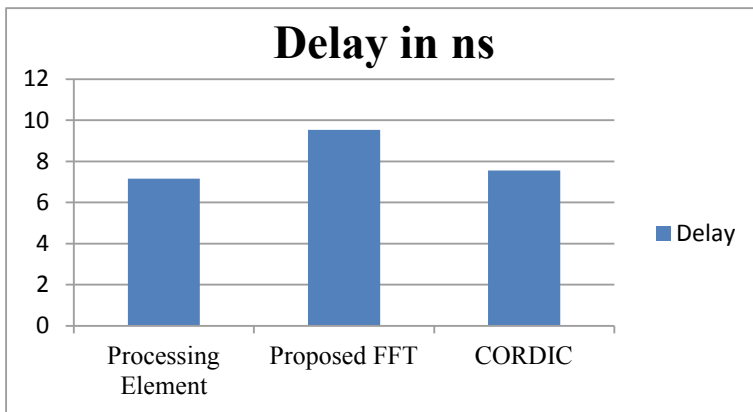
On-Chip	Power (W)	Used	Available	Utilization (%)
Clocks	0.000	2	---	---
Logic	0.000	668	29504	2
Signals	0.000	1209	---	---
MULTs	0.000	8	36	22
I/Os	0.000	156	376	41
Leakage	0.203			
Total	0.203			

Supply Summary		Total	Dynamic	Quiescent
Source	Voltage	Current (A)	Current (A)	Current (A)
Vccint	1.200	0.069	0.000	0.069
Vccaux	2.500	0.045	0.000	0.045
Vcco25	2.500	0.003	0.000	0.003

Supply Power (W)		Total	Dynamic	Quiescent
		0.203	0.000	0.203

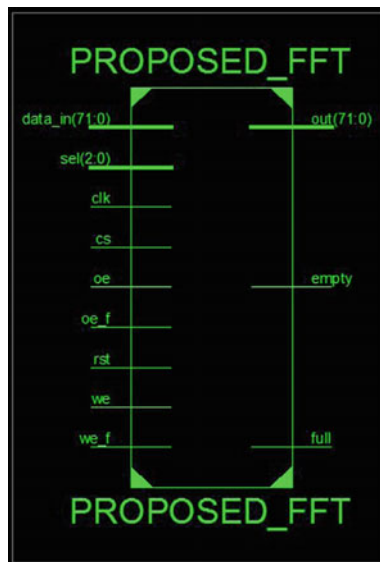
Thermal Properties		Effective TJA	Max Ambient	Junction Temp
		(C/W)	(C)	(C)
		18.6	81.2	28.8

Delay Analysis



See Figs. 5, 6, 7, 8, 9, 10 and Tables 2 and 3.

Fig. 5 Top module of proposed FFT



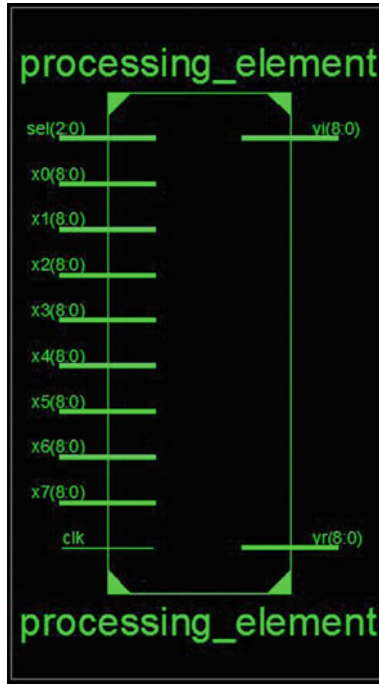


Fig. 6 Top module of processing element

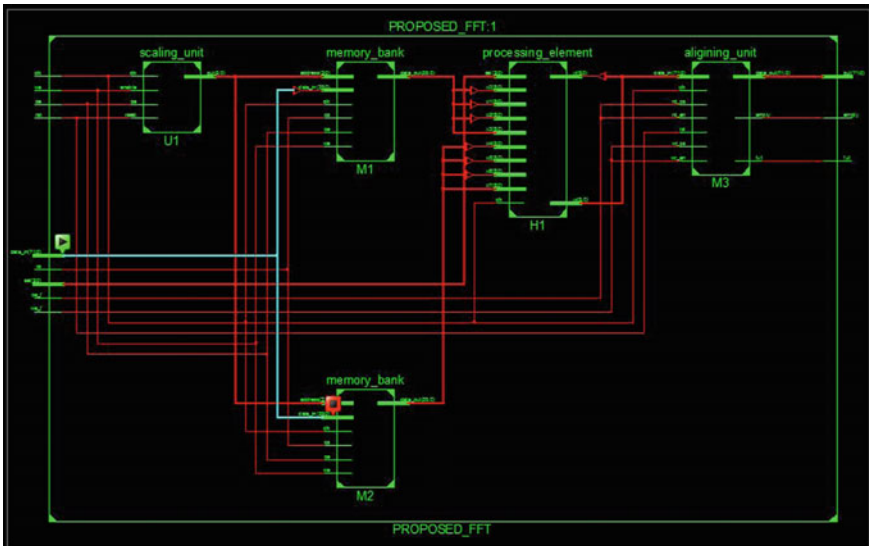


Fig. 7 RTL schematic of proposed FFT

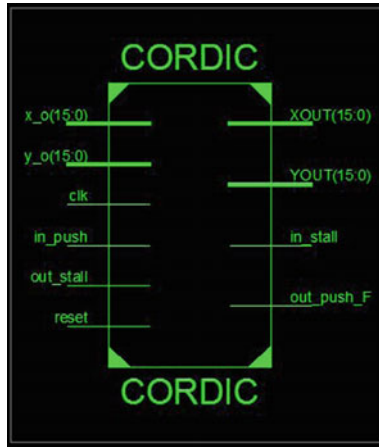


Fig. 8 Top module of CORDIC

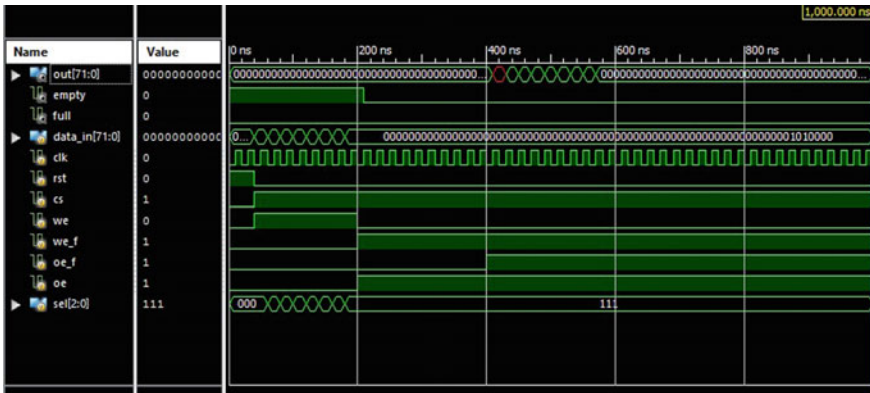


Fig. 9 Simulation results of proposed FFT processor

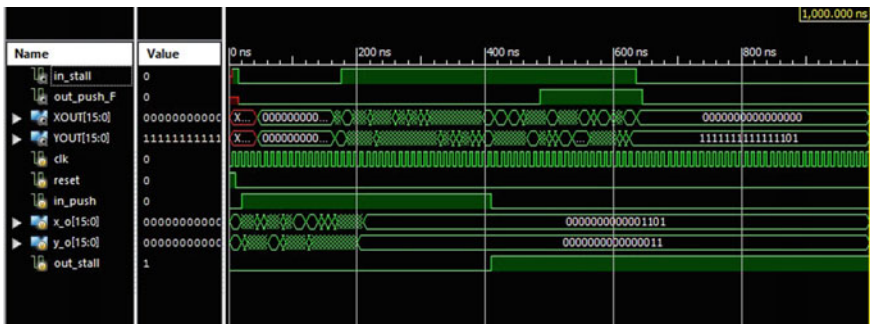


Fig. 10 Simulation results of CORDIC

Table 2 Device utilization summary of proposed processor

Logic utilization	Used	Available	In %
Total number of slice registers	412	29,504	1
Number of 4 input LUT	664	29,504	2
Number of bonded IOBs156	156	376	41

Table 3 Comparison results

Parameter	Proposed work	Xia [1]	Chen [2]
Power dissipation	20.3 mW	26.3 mW	320 mW

4 Conclusion

In this paper, the processor is designed for high speed applications and source code is written in Verilog. Synthesis and simulation are done in Xilinx ISE 14.7. The power dissipation is minimized (20.3 mW), and delay is 9.539 ns and with CORDIC extended delay is 7.55 ns.

References

1. K.-F. Xia, A memory-based FFT processor design with generalized efficient conflict-free address schemes. *IEEE Trans. VLSI* **25**(6) (2017)
2. J. Chen, J. Hu, S. Lee, G.E. Sobelman, Hardware efficient mixed radix-25/16/9 FFT for LTE systems. *IEEE Trans. VLSI Syst.* **23**(2), 221–229 (2015)
3. S.-N. Tang, C.-H. Liao, T.-Y. Chang, An area- and energy-efficient multimode FFT processor for WPAN/WLAN/WMAN systems. *IEEE J. Solid-State Circuits* **47**(6), 1419–1435 (2012)
4. S.-N. Tang, F.-C. Jan, H.-W. Cheng, C.-K. Lin, G.-Z. Wu, Multimode memory-based FFT processor for wireless display FD-OCT medical system. *IEEE Trans. Circuits Syst. I Regul. Pap.* **61**(12), 3394–3406 (2014)
5. P. Wang, J. McAllister, Y. Wu, in *Software Defined FFT Architecture for IEEE 802.11ac*. Proceedings IEEE Global Conference Signal Information Processing (2013), pp. 1246–1249
6. H. Abdoli, H. Nikmehr, N. Movahedinia, F. de Dinechin, Improving energy efficiency of OFDM using adaptive precision reconfigurable FFT. *Circuits Syst. Signal Process.* **36**(7), 2742–2766 (2017)
7. V.M. Somashekhar, R.P. Singh, FPGA implementation of fault tolerant adder using verilog for high speed VLSI architectures. *Int. J. Eng. Adv. Technol. (IJEAT)* **9**(4) (2020). ISSN: 2249–8958
8. S. Malipatil, R. Basavaraju, P. Kumar Nartam, Low power & high speed carry select adder design using verilog. *IOSR J. VLSI Sig. Process. (IOSR-JVSP)* **6**, 77–81 (2016)
9. T.B. Nguyen, H. Lee, High-throughput low-complexity mixed-radix FFT processor using a dual-path shared complex constant multiplier. *J. Semicond. Technol. Sci.* **17**(1), 101–109 (2017)
10. S. Malipatil, A. Gour, V. Maheshwari, Design & implementation of reconfigurable adaptive fault tolerant system for ALU. *Int. J. Electr. Eng. Technol.* **11**(9), 01–07 (2020)

A Literature Review on Bidirectional Encoder Representations from Transformers



S. Shreyashree, Pramod Sunagar, S. Rajarajeswari, and Anita Kanavalli

Abstract Transfer learning is a technique of training a model for a specific problem and using it as a base for training another related problem. It has been proved to be very effective and has two phases: the pre-training phase (generation of pre-trained models) and the adaptive phase (reuse of pre-trained models). Auto-encoding pre-trained learning model is one type of pre-trained model, which uses the transformer model's encoder component to perform natural language understanding. This work discusses the bidirectional encoder representations from transformers (BERT) and its variants and relative performances. BERTs are transformer-based models developed for pre-training unlabeled texts, bidirectional, by considering the semantics of texts from both sides of the word being processed. The model implements the above function using two specific functions: masked language modeling (MLM) and next sequence prediction (NSP). The robustly optimized BERT (RoBERTa) variant of BERT with few modifications has significant improvements in removing NSP loss function due to its inefficiency. SpanBERT is another variant that modifies MLM tasks by masking contiguous random spans and also uses the span-boundary objective (SBO) loss function. A lite BERT (ALBERT) is another variant with two-parameter reduction techniques: factorized embedding parameterization and cross-layer parameter sharing. It also uses inter-sentence coherence loss instead of NSP. The performance of the BERT's variants is found to be better than BERT, with few modifications as per the available literature.

Keywords Bidirectional encoder representations from transformers (BERT) · Robustly optimized BERT (RoBERTa) · A lite BERT (ALBERT) · Span-boundary

S. Shreyashree (✉) · P. Sunagar · S. Rajarajeswari · A. Kanavalli
Department of Computer Science and Engineering, M S Ramaiah Institute of Technology
(Affiliated To VTU), Bengaluru, India

P. Sunagar
e-mail: pramods@msrit.edu

S. Rajarajeswari
e-mail: raji@msrit.edu

A. Kanavalli
e-mail: anithak@msrit.edu

objective (SBO) · Masked language modeling (MLM) · Next sequence prediction (NSP) · Sequence-to-sequence (Seq2Seq)

1 Introduction

The sequence-to-sequence modeling (Seq2Seq) in NLP is a method of building models that are used to transform sequences of text from one kind of representation (e.g., a phrase in Kannada) to another (e.g., phrases in Hindi). This technique has a wide range of applications, namely question answering, machine translation. This modeling includes two tasks, namely natural language understanding and natural language generation. The focus of this paper is on natural language understanding. Machines can quickly analyze the data in moments and save the company's innumerable hours and resources, while analyzing client feedbacks uses natural language understanding (NLU) and machine learning. The capacity of a system to comprehend natural language is the main deliberation of NLU. It primarily focuses on reorganizing unstructured input data that aids the machine in analyzing and understand the data. The machines should identify the characteristics of the natural language before processing the data. Applications of NLU include machine translation, question answering, text summarization. Instead of using traditional machine learning modeling approaches, the viable option is to use transfer learning. It is a method that incorporates the re-usability of models with slight modifications or fine-tuning applied [1, 2]. It is measured very efficiently compared to the traditional machine learning approach, as it takes less time to build models. One more critical aspect is that this approach does not need a large dataset to train the model. In this type of learning, the initial layers of the neural network learn more general features and make a shift toward specificity when moving toward the final layers. This kind of initialization to the model increases the performance as compared to initialization using random features [3].

There are two phases for transfer learning implementation, i.e.,

1. Pre-training phase where the initial, re-usable pre-trained models are built.
2. The adaptive phase where the fine-tuning for the pre-trained model is performed.

The pre-trained models can be classified into two types:

1. **Auto-encoding models:** These models implement the encoder component of the transformer model and perform natural language understanding.
2. **Auto-regressive models:** These models implement the decoder component of the transformer model and perform natural language generation.

To enhance the natural language processing functions, it is vital to perform the pre-training of the language model [4]. Such NLP functions include phrase-level tasks, such as paraphrasing. The relationships between the phrases are determined, analyzed, and token-level tasks such as named entity recognition (NER), which focuses on generating token-level output. The standard language models are unidirectional, which leave few choices for architectures being used while pre-training.

This is a significant disadvantage that had to be overcome. The famous BERT model is known to be an auto-encoding pre-trained model. The BERT model works by pre-training the natural language representations bidirectionally unlike the unified LM pre-training [5], from the unstructured text, considering semantics from both the sides of the processing word in all the layers of its neural network [6]. BERT uses two main loss functions:

1. Masked language modeling (MLM)
2. Next sequence prediction (NSP).

1.1 Input Representation in BERT

The input to a BERT model follows a specific format, as shown in Fig. 1. It requires three embeddings, i.e.,

1. **Token embedding:** Represents a vector for the corresponding input tokens.
2. **Segment embedding:** If two sentences are given to the model as input from two different segments, then this embedding represents the corresponding segment that the token belongs to.
3. **Position embedding:** It represents the position vectors of corresponding tokens.

The combination of all the above three embeddings represents the final input that is given to the model. Input to the BERT model can either be a pair of sentences or a single sentence. In addition, there are two unique tokens used in the input representations, namely:

1. **CLS token:** This is a classifier token primarily used in the NSP loss function. It is added at the front of every input symbol.
2. **SEP token:** It is a separator token used to identify the ending of one sentence from one segment and the beginning of another sentence from another segment.

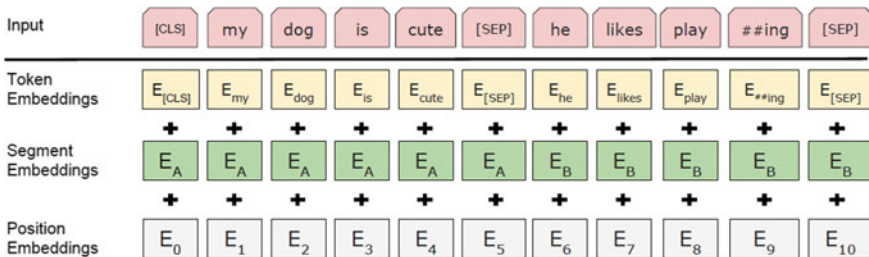


Fig. 1 Input representation for BERT model

1.2 BERT Architecture

Notation: The number of layers (transformer blocks) is denoted as L , the hidden size as H , and the number of self-attention heads as A . There are two models of BERT based on the number of parameters or model sizes: BERTBASE ($L = 12, H = 768, A = 12$, total parameters = 110 M), and BERTLARGE ($L = 24, H = 1024, A = 16$, total parameters = 340 M) [6] (Fig. 2).

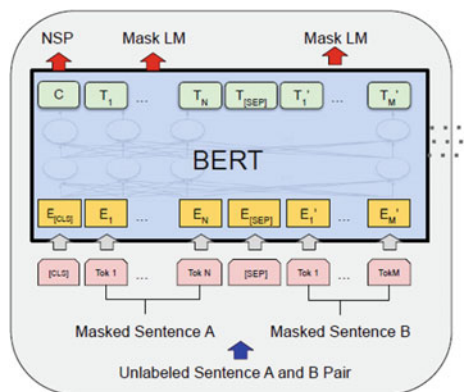
1.2.1 Masked Language Modeling

This function of BERT, inspired by Cloze-driven pre-training [7], eliminates the problem of unidirectionality of standard architectures. The working of MLM involves masking few percentages of the input tokens randomly and predicting the masked input word considering only its context. Instead of pre-training the model in only one direction, this function enables left-to-right pre-training, which results in a strong understanding of the input by the machine. Practically, 15% of the input tokens are randomly selected and masked, where, e.g., for the sentence “My dog is hairy” with the last word being masked as follows.

- About 80% of the time, the tokens are replaced by the token [MASK], i.e., “My dog is [MASK].”
- About 10% of the time, the tokens are replaced by another word, i.e., “My dog is apple.”
- Another 10% of the time, the tokens are kept as is, i.e., “My dog is hairy.”

The above examples show that not all the tokens are masked in the same manner all the time.

Fig. 2 BERT architecture



1.2.2 Next Sequence Prediction

In many NLP tasks, such as question answering, the model must understand the relationship between sentences present in the corpus. For this reason, the NSP function is introduced, which is a simple binary classification task. For example, given two sentences, A and B, this task determines whether sentence B is an immediate successor of A in the original document. It will place its output in the [CLS] token with *IsNext* or *NotNext*, where *IsNext* represents a positive input, and *NotNext* represents a negative input. Such examples are done equiprobably, i.e., 50% of the positive information and 50% of the negative samples are used to pre-train. To summarize, BERT performs natural language understanding with the input being sentenced to mask some of the tokens. The output is the prediction of masked tickets by considering the tokens' left and proper context. Thus, introducing bidirectionality in the process provides accurate results, and the machine will have more understanding of the input sentences provided.

2 Types of BERT Models

2.1 *RoBERTa*

The *RoBERTa* stands for robustly optimized BERT, a replication study of BERT with consideration toward training data size. The modifications in this model, after observing that the BERT model was undertrained, include:

- Training the model with the large dataset
- Removing the NSP loss function
- Training with lengthy sequences
- Changing the masking patterns dynamically

To check the significance of NSP loss function, the researchers experimented on BERT with four training arrangements:

1. **SEGMENT PAIR + NSP:** The input arrangement used in BERT, where pair of segments was provided, chosen from one or different documents, along with the NSP loss function. The total length of the input should be less than 512 tokens.
2. **SENTENCE PAIR + NSP:** In this arrangement, the input is formed by combining two sentences rather than segments chosen from the same or different documents. The NSP loss function is retained in this case.
3. **FULL SENTENCES:** The input contains complete sentences taken from one or more documents continuously, with the maximum number of tokens being 512. When the sentences from one document come to an end, and the 512-threshold is not reached yet, then a separator token is introduced with the sentences starting from the following document. The NSP loss function is removed.

Table 1 Comparison of F1 scores of training formats in RoBERTa

Models	SQuAD 1.1	SQuAD 2.0
<i>Using NSP loss</i>		
SEGMENT PAIR	90.4	78.7
SENTENCE PAIR	88.7	76.2
<i>Without using NSP loss</i>		
FULL SENTENCES	90.4	79.1
DOC SENTENCES	90.6	79.7

4. **DOC SENTENCES:** The input format is similar to FULL SENTENCES, but if the sentences of a document are finished inputting, then the sentence from the following document is not sampled. There might be cases where the 512 threshold would not be satisfied. In those cases, the batch size is increased dynamically to have the total length. The NSP loss function is removed.

After implementing these training formats on SQuAD 1.1 and SQuAD 2.0 datasets, below is the observation that Liu et al. (2019) found.

From Table 1, it can be concluded that.

- The original BERT model will not perform well when using sentences individually because the model cannot capture long-range dependencies.
- The model shows an increase in performance when trained without NSP loss.
- The model performs better when texts are provided as inputs from a single document (DOC SENTENCES) instead of delivering texts from multiple documents (FULL SENTENCES) [8].

2.2 SpanBERT

The SpanBERT is another variant of BERT that mainly focuses on representing and predicting spans of text, and it comes with two modifications:

1. Instead of random tokens, random contiguous spans of text are masked.
2. The masked span of text is predicted with the help of span-boundary tokens, i.e., span-boundary objective, without depending on individual tokens for prediction [9].

SpanBERT is trained by removing the NSP loss function and providing individual segments of text rather than two text segments having half-length.

Span Boundary Objective (SBO): In this approach, the span of text selected to mask will be predicted by the model as a final output using representations of the tokens at the beginning and end of the span only. If we denote the model's output as x_1, x_2, \dots, x_n , and (b, t) represents the starting and ending positions of the masked span, and then the span is denoted by (x_b, \dots, x_t) . The output y_j , which is the predicted

token x_j in the span, is a function of the encoding representations of the boundary variables (x_{b-1} and x_{t+1}) and the positional encoding vector (p_{j-b+1}).

$$y_j = f(x_{b-1}, x_{t+1}, p_{j-b+1}) \tag{1}$$

where the positional encoding vectors p_1, p_2, \dots are the position vector relative to the starting vector of the masked span, x_{b-1} . The function is implemented as a feedforward neural network consisting of two layers with GeLU as the activation function [10] and layer normalization. The two hidden layers and the output are given as,

$$h_0 = [x_{b-1}; x_{t+1}; p_{j-b+1}] \tag{2}$$

$$h_1 = \text{LayerNorm}(\text{GeLU}(W_1 h_0)) \tag{3}$$

$$y_j = \text{LayerNorm}(\text{GeLU}(W_2 h_1)) \tag{4}$$

SpanBERT combines the two loss functions, i.e., MLM and SBO to get the final loss,

$$\mathfrak{L}(x_j) = \mathfrak{L}_{\text{MLM}}(x_j) + \mathfrak{L}_{\text{SBO}}(x_j) = -\log P(x_j|x_j) - \log P(x_j|y_j) \tag{5}$$

Here, the summation is done on negative log likelihood losses.

Figure 3 shows an example of working of SpanBERT.

In Fig. 3, the “transformer encoder” represents the SpanBERT model. Four contiguous texts (span) are masked in the given sentence, i.e., “an American football game.” The boundary tokens are “was” and “to,” which are embedded as x_4 and x_9 . Using these tokens, the span of masked text is predicted. If we consider the word “football” as an example, its position vector is p_3 , i.e., the relative position with respect to x_4 . The loss function of the word “football,” using Eq. 6, is given as,

$$\mathfrak{L}(x_i) = \mathfrak{L}_{\text{MLM}}(\text{football}) + \mathfrak{L}_{\text{SBO}}(\text{football})$$

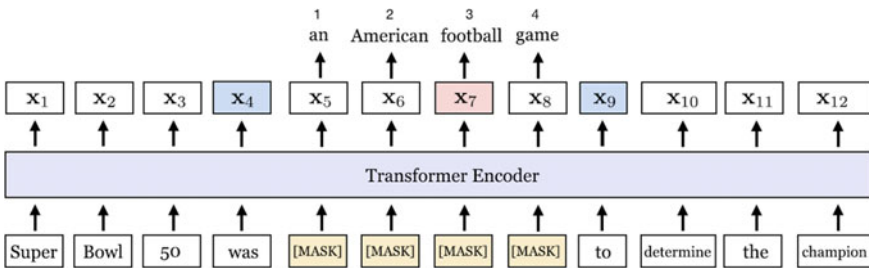


Fig. 3 Example working of SpanBERT

$$= -\log P(\text{football}|x_7) - \log P(\text{football}|x_4; x_9; p_3) \quad (6)$$

The loss function is nothing but a log likelihood function describing the probability of the word given other tokens.

2.3 ALBERT

There is an increase in the model's performance while pre-training if the model's size is large enough. But, this increase in size impacts the performance because of memory limitations (graphics/tensor processing units) and long training periods [11]. To avoid these problems, ALBERT, stands for A lite BERT, was introduced with few modifications. Furthermore, these modifications lead to better scaling because ALBERT uses fewer parameters than BERT by using two-parameter reduction techniques. The two-parameter reduction techniques used to improve scaling of pre-trained models are:

- **Factorized embedding parameterization:** Usually, in BERT, the hidden layer size H is tied to the embedding size Em , i.e., $Em = H$. But, this is not an optimal approach to pre-train. Because as and when the size of the hidden layer increases, the embedding matrix ($V_0 * Em$) size also increases, with V_0 being the vocabulary size. This can result in billions of parameters. For this reason, the embedding matrices are fragmented into two smaller matrices by ALBERT. Rather than providing one-hot vectors to the hidden space H directly, they are provided to the lower-dimensional embedding space Em and H . Thus, there is a reduction in parameter size from $O(V_0 * H)$ to $O(V_0 * Em + Em * H)$. This reduction seems efficient only when $H \gg Em$.
- **Cross-layer parameter sharing:** In most cases, either the feedforward network's parameters are shared, or the attention parameters are shared among all the layers. But in ALBERT, all the parameters are shared among all the layers.

Along with these modifications, ALBERT uses inter-sentence coherence prediction. The main reason for the inefficiency of NSP loss in BERT is that it combines topic prediction and coherence prediction. Topic prediction sometimes overlaps with what is learned during the MLM task. This technique only focuses on coherence prediction by introducing sentence-order prediction (SOP) loss. This follows the same method of NSP while training positive examples (training with two consecutive segments of text taken from the same document). Still, while teaching negative samples, the same set of successive segments is used by exchanging their order. From this process, the model understands the details between sentences and their coherences.

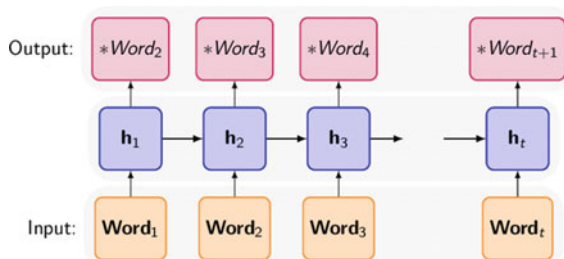
3 Related Works Models

There are many applications of Seq2Seq modeling, where the main applications include question answering, image captioning, text summarization, language translation, and many more. They can be implemented using various models, with the earliest model being recurrent neural network (RNN). Deep neural networks consist of a large category of neural networks called recursive neural networks. The weights applied in these neural networks are the same overall as the inputs, i.e., they are applied recursively to generate an output vector representation. The most well-known recursive neural networks for text classification are RNNs as they are easy to use and practical, with the reason being that they do not require any structural tags such as parse trees while processing the input text. RNNs are very popular in working with sequences. The below diagram depicts the general working of RNN [12]. Like any other neural network, RNN consists of the input layer, hidden layer, and output layer as the three essential layers in its architecture, as shown in Fig. 4. A hidden layer takes input from the corresponding input layer unit and the previously hidden layer unit to predict the following sequence at any given time, creating a dependency on the previously hidden layer unit. This leads to a problem of “long-range dependency”; i.e., the precision of the output decreases as the length of the input increases. Because, the output layer, at any given point of time, has knowledge only about the current input vector and the previously hidden state vector. There can be a possibility that there is an essential word at the beginning of the sentence and the output layer, while processing the last word, does not consider that important word. This disadvantage led to the introduction of the encoder-decoder model using RNNs.

The encoder-decoder models use the attention mechanism to improve their accuracy in predicting the following sequence. Below is the architecture of this model (Fig. 5).

Attention is driven by how we attend to several parts of an image or associate different words in a given text. In building deep learning models, the concept of attention is widely used. In terms of NLP and language modeling, attention is just a vector that contains important weights. The attention vector for a word that is being predicted is used to determine how strongly it is associated with other words by taking the weighted sum of the attention vectors to get an approximated vector. The encoder-decoder model has encoders and decoders that are a combination of RNNs.

Fig. 4 General working of RNNs



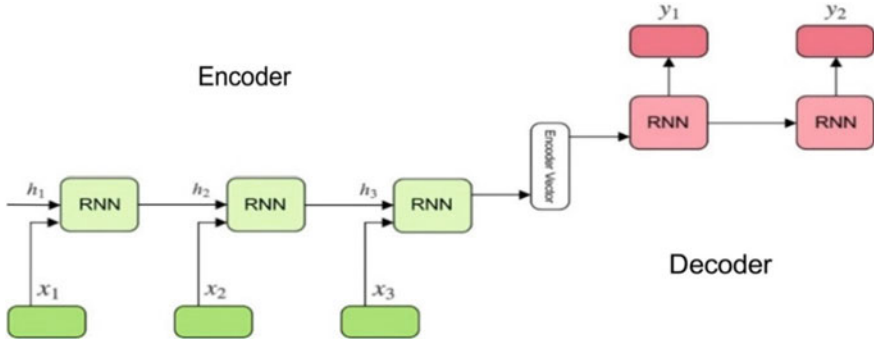


Fig. 5 Encoder-decoder model

Every hidden state unit forwards all the hidden state vectors from previously hidden state units in this model. All hidden state vectors, h_1 , h_2 , and h_3 , are provided at the end of the encoding stage. The hidden layer vector calculation at the encoder side is performed as,

$$h_i = f(W^{(hd)}h_{i-1} + W^{(hx)}x_i) \tag{7}$$

where

- h_i = i th hidden layer vector,
- h_{i-1} = $i - 1$ th hidden layer vector, previous state vector,
- x_i = input layer vector,
- $W^{(hd)}$ = weights between hidden layer,
- $W^{(hx)}$ = weights between hidden and input layer,
- f = activation function.

The hidden layer vector at the decoder side is as calculated below,

$$h_i = f(W^{(hd)}h_{i-1}) \tag{8}$$

where

- h_i = i th hidden layer vector,
- h_{i-1} = $i - 1$ th hidden layer vector, previous state vector,
- $W^{(hd)}$ = weights between hidden layers,
- f = activation function.

The output vector calculation at the decoder is calculated as below,

$$y_t = \text{softmax}(W^S h_t) \tag{9}$$

where

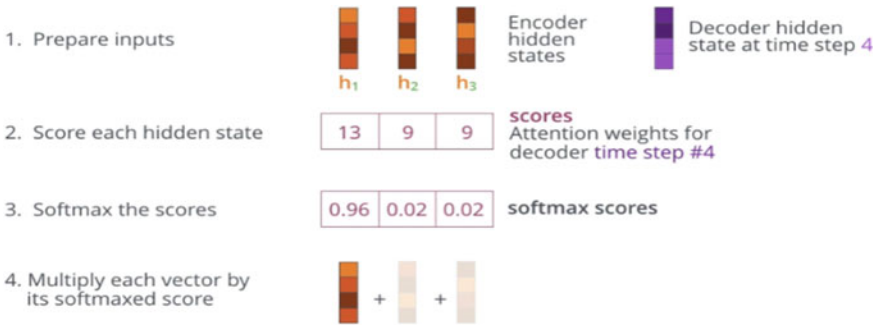


Fig. 6 Calculation of attention scores

y_i = output vector,
 W^S = weights between hidden and output layer,
 h_i = i th hidden layer vector,
 softmax() = softmax function which provides the probability vector.

Out of these hidden layer vectors h_1 , h_2 , and h_3 , representing the corresponding encoded input vectors, the most important sequences are chosen by applying attention mechanism, which provides attention scores. The calculation of attention scores as shown in Fig. 6.

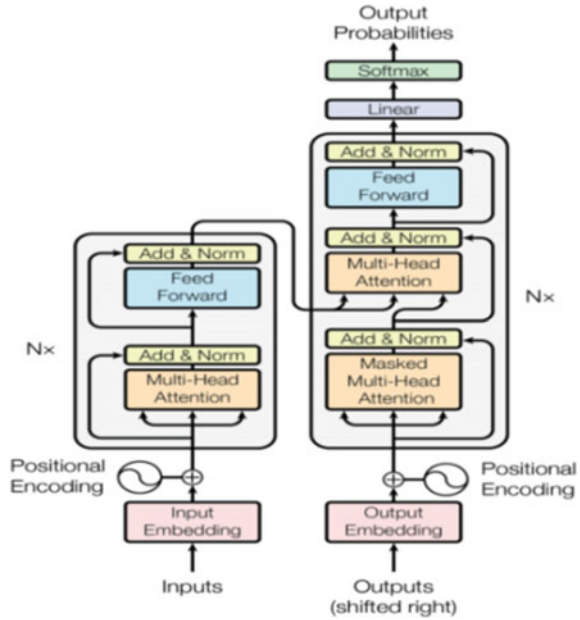
The most probable softmax score is treated as an essential sequence, and it is multiplied with the hidden layer vector before getting the weighted sum. This model also has all the disadvantages of RNN, i.e., sequential processing, leading to a lack of parallelism. To overcome this problem, transformer-based models are used. Typically, recurrent models consider the input token's positions while computing the output. They produce the hidden layer state's output as a function of position i and the previously hidden layer state h_{i-1} . This is the very reason which prevents parallelism while training, which degrades the performance when the input length is increased as there might occur limitations on memory.

Some of the works done recently show some improvements in the performance of the model. But, the core problem of serial computation persists. Self-attention, commonly referred to as intra-attention, is a process of attention that relates different tokens in different positions to represent a particular token. There are several tasks where self-attention finds its applications in. They are text summarization, reading comprehension, text-based inferences, and learning task-independent text representations. And, transformer models depend entirely on self-attention to generate the input and output representations without involving serial computing RNNs [13]. The architecture of the transformer model is shown in Fig. 7.

Out of all the components, the essential components are,

- Multi-head attention:** The attention function is expressed as a mapping between a query, some set of key-value pairs, and an output, where all of them are vectors. The outcome is calculated as a weighted sum of value vectors. A compatibility

Fig. 7 Transformer model



function of the query and corresponding key vector is used to assign a weight for the value vector [14].

- Scaled-dot product attention:** The input comprises query vector, d_k dimensional key vector, and d_v dimensional value vector. To compute the weights of the value vector, dot products of query and key vectors are calculated, followed by dividing each intermediate result by the square root of the dimension of essential vector d_k , and finally applying softmax function.

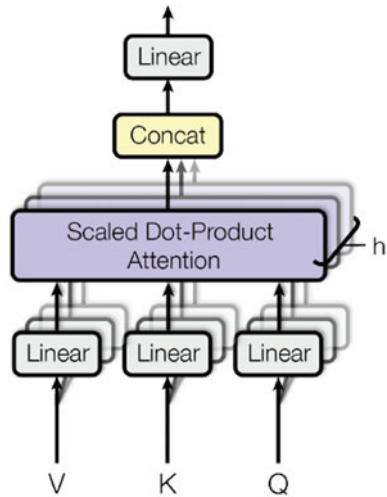
The formula for calculating self-attention is given below,

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{10}$$

Q = query vector, K = key vector, V = value vector, d_k = dimension of key vector, multiplying input vector with weights W_q, W_k, W_v . Below is the figure for the multi-head attention layer (Fig. 8).

Rather than calculating the attention function once, with d_{model} dimensions of query, key, and value vectors, all these vectors are linearly projected h times, with distinct learned projections with respective dimensions. Then, the attention function is applied to each of these projected vectors parallelly generating d_v dimensional output vectors. Finally, these are combined and projected again to produce the final results.

Fig. 8 Multi-head attention layer



- **Masked multi-head attention:** This layer of the decoder, while processing a word, masks all the subsequent sequences and gets the context only based on previous sequences, retaining auto-regressive property.
- **Positional encoding:** The position of a word has a significant impact on its meaning. This information is also encoded in the input embedding.

Parallelism is introduced only after calculating attention scores, i.e., at the feed-forward neural network layer. Even though it provides parallelism, it understands the given input sequence only from one direction, which does not offer accurate results. This led to auto-encoding pre-trained models.

Title, Author, and Year	Model name	Key highlights
Deep learning-based text classification: a comprehensive review Author: Minaee S Year: 2021	RNN	<ul style="list-style-type: none"> • Same weights are applied recursively all along the neural network • Unidirectional processing • Leads to “long-range dependency” problem • No parallelization
Deep learning-based text classification: a comprehensive review Author: Minaee S Year: 2021	RNN with encoder-decoder	<ul style="list-style-type: none"> • Uses multiple RNNs for constructing encoders and decoders • Captures important words using “attention” mechanism • Unidirectional processing • No parallelization

(continued)

(continued)

Title, Author, and Year	Model name	Key highlights
Attention is all you need Author: Vaswani A Year: 2017	Transformers	<ul style="list-style-type: none"> • Uses “self-attention” mechanism along with encoder-decoder model to understand the context better • Unidirectional processing • Uses positional encoding • Introduces parallelization
Transfer learning in natural language processing Author: Ruder S Year: 2019	Auto-encoding pre-trained models (BERT and its variants)	<ul style="list-style-type: none"> • Uses encoder segment of the transformer model • Introduces bidirectionality • Faster training due to transfer learning

4 Existing Works and Results

All the variants have been implemented on SQuAD 1.1 and SQuAD 2.0 datasets and are compared with the basic BERT model using their EM and F1 scores. SQuAD stands for Stanford question answering dataset, which is a dataset used for reading comprehension. It contains question and answer pairs, where the questions are posed by crowd workers on a set of Wikipedia articles and the answers are texts taken from those articles. SQuAD 1.1 is the previous version, containing 100,000 + question–answer pairs. SQuAD 2.0 is the latest version which contains unanswerable questions along with the question–answer pairs of SQuAD 1.1 dataset. EM score is called exact match score that indicates the number of answers matching exactly with the ground truth. F1 score is a measure of accuracy of the trained model.

BERT Versus RoBERTa

As observed by Liu et al. (2019), by removing the NSP loss function from the original BERT model, the accuracy increases (Table 2).

BERT Versus ALBERT

As observed by Lan et al. (2019), by using parameter reduction techniques and coherence loss function, BERT can be scaled efficiently (Table 3).

Table 2 Comparison of F1 and EM score between BERT and RoBERTa

Models	SQuAD 1.1		SQuAD 2.0	
	EM	F1	EM	F1
BERT	84.1	90.9	79.0	81.8
RoBERTa	88.9	94.6	86.5	89.4

Table 3 Comparison of F1 and EM score between BERT and ALBERT

Models	SQuAD 1.1		SQuAD 2.0	
	EM	F1	EM	F1
BERT	85.5	92.2	82.2	85.0
ALBERT	88.3	94.1	85.1	88.1

Table 4 Comparison of F1 and EM score between BERT and SpanBERT

Models	SQuAD 1.1		SQuAD 2.0	
	EM	F1	EM	F1
BERT	84.3	91.3	80.0	83.3
SpanBERT	88.8	94.6	85.7	88.7

BERT Versus SpanBERT

As observed by Joshi et al. (2020), masking spans of contiguous texts yields better performance than the original architecture (Table 4).

5 Applications

BERT and its variants can be used in several different NLP tasks, such as text classification, semantic similarity between pairs of sentences, question-answering task with the paragraph, text summarization, and machine translation.

All the applications mentioned above give more accurate results due to the bidirectionality information retrieval property of BERT.

6 Conclusion

Subsequent scientific developments related to language models using transfer learning have shown that sophisticated, unsupervised pre-training is a crucial feature of many systems of natural language comprehension. Even, low-resource tasks can be benefitted from deep unidirectional systems with this type of training. This concept of transfer learning helps in tackling a wide variety of NLP tasks by using the same pre-trained model. Once various BERT models have been evaluated, it has been shown that the efficiency may be significantly improved by training the model longer with more data, eliminating the NSP function, training for lengthy sequences, and dynamically modifying the masking patterns that are applied to the training data. The enhanced pre-training process, known as RoBERTa, attains cutting-edge results on SQuAD datasets. These findings show the significance of BERT's design

choices that were overlooked. Pre-training SpanBERT shows an increase in its performance compared to BERT in many tasks, with the change being that a span of tokens is masked contiguously instead of a single token. With the introduction of two-parameter reduction strategies, the pre-training procedure of ALBERT model improves the scalability.

Acknowledgements This work was supported by M S Ramaiah Institute of Technology, Bangalore-560054 and Visvesvaraya Technological University, Jnana Sangama, Belagavi-590018.

References

1. S. Ruder, M.E. Peters, S. Swayamdipta, T. Wolf, in *Transfer Learning in Natural Language Processing*. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials, pp. 15–18 (2019). <https://doi.org/10.1017/9781139061773.020>
2. A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, *Improving Language Understanding by Generative Pretraining* (2018)
3. J. Yosinski, J. Clune, Y. Bengio, H. Lipson, in *How Transferable are Features in Deep Neural Networks?* Advances in Neural Information Processing Systems 27, NIPS Foundation (2014)
4. X. Qiu, T. Sun, Y. Xu et al., Pre-trained models for natural language processing: a survey. *Sci. China Technol. Sci.* **63**, 1872–1897 (2020). <https://doi.org/10.1007/s11431-020-1647-3>
5. L. Dong, N. Yang, W. Wang, F. Wei, X. Liu, Y. Wang, J. Gao, M. Zhou, H.W. Hon, in *Unified Language Model Pretraining for Natural Language Understanding and Generation*. Advances in Neural Information Processing Systems (NIPS) (2019)
6. J. Devlin, M.W. Chang, K. Lee, K. Toutanova, Bert: *Pre-training of Deep Bidirectional Transformers for Language Understanding* (2018). arXiv preprint arXiv: 1810.04805
7. A. Baevski, S. Edunov, Y. Liu, L. Zettlemoyer, M. Auli, *Cloze-Driven Pretraining of Self-Attention Networks* (2019). arXiv preprint arXiv: 1903.07785
8. Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, V. Stoyanov, *Roberta: A Robustly Optimized Bert Pretraining Approach* (2019). arXiv preprint arXiv: 1907.11692
9. M. Joshi, D. Chen, Y. Liu, D.S. Weld, L. Zettlemoyer, O. Levy, Spanbert: improving pretraining by representing and predicting spans. *Trans. Assoc. Comput. Ling.* **8**, 64–77 (9) (2020)
10. D. Hendrycks, K. Gimpel, *Gaussian Error Linear Units (Gelus)* (2016). arXiv preprint arXiv: 1606.08415
11. Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, R. Soricut, *Albert: A Lite Bert for Self-Supervised Learning of Language Representations*. arXiv
12. S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, J. Gao, Deep learning-based text classification: a comprehensive review. *ACM Comput. Surv. (CSUR)* **54**(3), 1–40 (2021). <https://doi.org/10.1145/3439726>
13. S. Singla, Comparative analysis of transformer based pre-trained NLP Models. *Int. J. Comput. Sci. Eng.* **8**, 40–44 (2020). <https://doi.org/10.26438/ijcse/v8i11.4044>; <https://doi.org/10.5121/csit.2021.110111>
14. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin, in *Attention is All You Need*. 31st Conference on Neural Information Processing Systems (Long Beach, CA, USA, 2017)
15. J. Andreas, M. Rohrbach, T. Darrell, D. Klein, *Learning to Compose Neural Networks for Question Answering* (2016). arXiv preprint arXiv: 1601.01705

Localization and Multi-label Classification of Thoracic Diseases Using Deep Learning



Atique Siddiqui, Sudhanshu Chavan, Sana Fatima Ansari,
and Prasenjit Bhavathankar

Abstract The chest X-ray image has been one of the most commonly accessible, which is used for radiological examination whether it be for screening or diagnosis of chest diseases. An enormous amount of study has been done in the field of medical imaging accompanied by different radiologists, but sometimes even for the radiologists it becomes difficult and challenging to examine and review chest radiographs. Our paper aims to provide a new approach for diagnosis of chest diseases into different categories having 15 different labels with the help of transfer learning using pre-trained VGG-16 neural network and localization of the chest images using class activation mapping (CAM). For training the whole model and performing the task, we used a chest X-ray presented by NIH. This has 14 different labels like pneumonia, nodule, and lastly no finding.

Keywords Visual geometry group (VGG) · Class activation mapping (CAM) · Convolutional neural network (CNN) · Chest X-ray (CXR) · National Institution of Health (NIH) · Natural language processing (NLP) · Generative adversarial network (GAN)

1 Introduction

The total number of X-ray images globally is increasing each year steadily over the decades, but from the X-ray images chest X-ray images are being used to diagnose a large number of chest diseases which include mass, nodule, infiltration, cardiomegaly, and many more. These tasks of examining and reviewing the X-ray reports are still being done by the radiologists in a traditional way that is going to each and every scan without any automated system. In this present paper, we are trying to localize and predict the different thoracic disease categories with the help of transfer learning using the VGG-16 model and class activation mapping. The whole task would be

A. Siddiqui (✉) · S. Chavan · S. Fatima Ansari · P. Bhavathankar
Bharatiya Vidya Bhavan's Sardar Patel Institute of Technology, Mumbai, India
e-mail: atique.siddiqui@spit.ac.in

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
S. Smys et al. (eds.), *Inventive Computation and Information Technologies*, Lecture Notes
in Networks and Systems 336, https://doi.org/10.1007/978-981-16-6723-7_24

321

done on the dataset released by NIH [1] containing 14 distinct thorax diseases having around 1,10,000 grayscale chest images collected over 30,000 unique patients.

Generally, doctors are quite satisfactory in their diagnosis of the diseases but mistakes can happen where some minor details that are to be examined would be left out. one of the studies [2] shows that when the doctors take a second opinion 66 percent of the time it is refined which is given to the second doctor in which 21 percent of the time diagnosis is changed from the original one. Considering India which has a population of nearly around 1.3 billion but only very few radiologists that are 10,000 in number but to overcome and solve this our model which would be trained on various categories would help to predict the thorax diseases and would give a check in order to get better diagnostics. Our paper aims to bridge the uncertainty and misdiagnosis of diseases by making a model predict the diseases that could serve as a helping tool for radiologists.

The rest of the paper is structured as follows: A literature survey on previous which is done to date. A brief to the various architectures we would be using (VGG-16 and CAM). Followed by this an introduction to the dataset would be training our model on the details of the proposed solution and methodology and finally the experimentations and our results.

2 Literature Review

Performing diagnoses using chest X-ray has become very fast and easy nowadays. Performing diagnoses using X-ray may include any part such as heart, lung, chest, and any other organ so this leads to pile up and deposit large numbers of reports and X-rays in the hospital. On the flip side, these hospital's databases include the crucial information processing images so their question arises how they use this information and smoothly construct the computer-aided diagnosis system, presenting a database namely "ChestX-ray8" which includes more than 1 lakh X-ray images with a frontal view of more than 32 thousand patients which consists of multi-labels using NLP related to the radiology reports. Also, they indicate that these diseases that take place frequently observe and structure singly. Weakly classification of numerous label images does the localization using this dataset [1]. Giving rise to the performance, deep learning capitulates the images of medical examinations, segmentation, and recognition tasks. Recently there are many works and overviews related to deep learning, and the key drawback is that the evaluation that is done on these different problems on working-class techniques of many patients. It cannot be said how well the deep learning approaches can be enlarged to many studies of patients. There are basically three factors describing the diagnosis based on large-scale images of medical, firstly the pathology labels and anatomy of image-level open-ended are not acquired by the crowd-sourcing, therefore they use natural language processing technique to possibly mine per multiple typical thoracic pathology label related to the radiological studies of chest X-rays, secondly, the spatial measurements of typically 2000X3000 pixels. Compared to the full image size, the pathological image

regions may display incredibly varying sizes or extensions. In order to overcome this difficulty, they formulate and validate the localization structure and classification of these multi-label images. Thirdly, while developing a database for medical images they need to learn the localization of models and deep recognition of images. So basically they illustrate that frequently taking place diseases can be localized and detected structurally with the formulation of disease localization and the classification of the weakly multi-label images [3].

As it is said deep learning promises and aims to boost the quality of medical treatment, nevertheless due to confusion about the little experience in artificial intelligence and objective data science, and lack of intelligence or thoughts about artificial intelligence (AI) in hospitals, hospital laziness, and strategic thinking; managerial acceptance barrier in this deep learning technology is still in experimentation process and not yet implemented quickly enough. The paper approaches to construct a very open prototype structure that grasps the reality of deep learning instruments for diagnosis of chest X-ray which can be pre-owned by the medical experts. To make the conformation or aid to the diagnosis, the system is built in such a way that it can provide different opinions to the users; this paper gives the superiority to the users firstly in order to protect anonymity, and patients' data is stored on the user computer, secondly, scaling the measurement at the minimal cost so the process takes place locally and the other distribution techniques such as the creation of the higher overhead and for the free prototype would be unsustainable [2]. The goal of this paper is with deep learning techniques enabling the medical section to judge where they are cooperative and where they can go wrong, and coming up with the response the problem to be worked on can be found by the machine learning section. The paper focuses on the three parts that is a description of prediction, projection, simplification, and the out of distribution diagnostic projection. In order to set the experts' professional abilities, these components came from the full instruments, so while developing this project they came across many obstacles for which they investigated the solution in this work. When the prediction is to be done, this is the main difficulty they came across, and the system architecture is the concept that can be applied to the solutions for medical globally without substantial server cost [4].

In the expectation that will accelerate the improved patient's effect, this paper gives the approach for the state-of-the-art machine learning techniques to solve the dilemma of the medical diagnosis; they proposed orthogonal directions: firstly after experimenting many times it verifies that the label dependencies that are ignored by the crafted baseline model will surpass the prior training state of the art by a wide margin without pre-training and secondly for better diagnostic outcomes between abnormality labels they suggested specifically leveraging the conditional dependencies, for the achievement; for such tasks, the RNN is intentionally updated, demonstrating the advantages over techniques that do not take into account interdependencies. The paper does the comparison with the results on ImageNet. In addition, to tackle the problem of clinical illustrable, the recording in the benchmark is made about the number of different metrics, and those commonly used in ml are compared [5].

One of the virtuosi in the medical profession is early prognosis, particularly in the cardiovascular sector. A diet chart developed by the concerned physician following

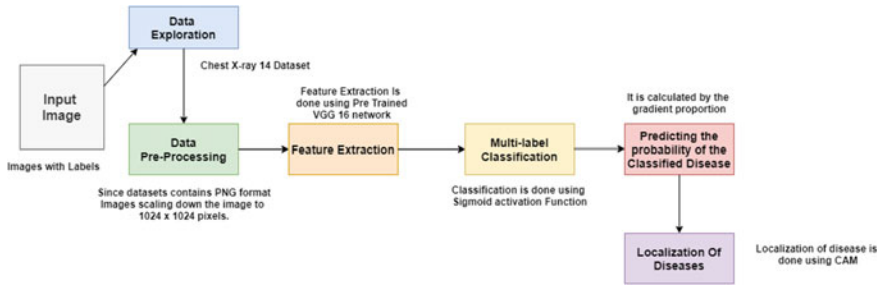


Fig. 1 Block diagram

early prognosis can be used to prevent cardiovascular disease. In this paper, using the machine learning method, create a pooled area curve (PUC) and by the proposed algorithm predict the coronary artery disease. For successful prediction, this type of knowledge-based identification is crucial in medical photographs. Despite the poor pixels surrounding it, this substantial strategy has a good impact on determining variation. This study examines contemporary adaptive image-based classification algorithms and compares them to existing classification methods in order to predict CAD early and with more accuracy [6]. Computerized tomography in lung segmentation has a very crucial role in diagnosis of various lung diseases. Deep learning lung segmentation schema based on the generative adversarial network (GAN) is proposed in the paper. In CT images, this proposed schema is popularized for lung segmentation in various neural networks. The findings of the testing showed that due to its improved performance and efficiency, as well as its simpler procedure, LGAN schema can be employed as a potential approach for lung segmentation [7]. This proposed paper has the motive that by estimating the chance of malignancy and segmenting the lung nodule, radiologists' obstacles will be reduced, and the early growth rate of pulmonary nodules will be detected. Basically the algorithm that the paper proposes is an enhancement filter for enhancing imported pictures and selecting nodules, and for reducing false positives the neural classifier. The findings displayed as response characteristics curves as the internal and exterior nodules are used to test the given model [8] (Fig. 1).

3 Architectures

3.1 Block Diagram

The above proposed system consists of two models that is class activation mapping for localization of diseases and VGG-16 for feature extraction of the images that is to generate feature maps for the input image. Once these two models have been trained, the output of the model will go to sigmoid activation function for multi-label



Fig. 2 VGGNET-16 (feature extraction)

classification that to classify the image into respective diseases. Once the input image is classified into diseases, their probabilities would be measured with the gradient proposition obtained while mapping the image. Once the probability is generated, the model will predict the disease (Fig. 2).

3.2 VGGNET-16 (Feature Extraction)

The VGG-16 architecture, i.e., visual geometry group 16 architecture is reviewed as one of the best observation model architectures to date. Simonyan and Zisserman from Oxford University came up with this idea of the VGG-16 model. In the yearly computer observation competition, i.e., ILSVRC in 2014, this VGG-16 architecture was used. Once a year they compete on two assignments, i.e., the object localization and image classification. This model gained a first and second position in the same heading. The most special feature of the VGG-16 platform is they concentrate on layers with filters of 3×3 and max pool and padding with the filter of 2×2 are used; these positioning of the layers are followed unabated. The 16 is having a weight of 16 layers. This network is a very big network.

Basically here we have added our own layers consisting of a dense and dropout layer. From the pre-trained network, we are removing the classifier layer and excluding the last layer, we are freezing all the weights, attaching our own classifier in the bottom and then training the resulting classifier.

In VGG-16, the convolution that is used is of 3×3 filter and stride is 1, in max pooling that will be using filter size of 2×2 and stride is 2 these things are uniform throughout the network. So here if the input of $224 \times 223 \times 3$ then will convolution it with 64×2 filters as the specification of the filters are given so it will be having 64 filters of $3 \times 3 \times 3$ and then the result getting will be applying ReLU activation function to it, then the result will be getting again do the convolution of 3×3 filter

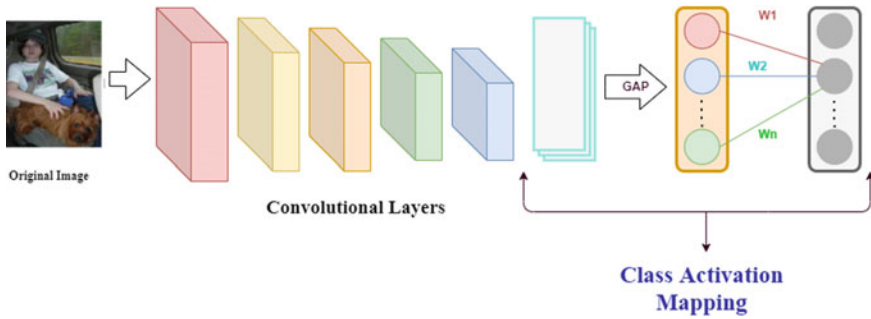


Fig. 3 Class activation mapping

then the output will be getting is of $224 \times 224 \times 64$, here 224 is because are using same padding then will apply max pool and get the result exactly the half of height and breadth, i.e., $112 \times 112 \times 64$ and then will do the convolution again with 128×2 filters and then will apply ReLU and then the output are getting will apply the convolution again with the same given filter get the output of $112 \times 112 \times 128$, will apply max pool to the same then get the output of $56 \times 56 \times 128$, now will apply the convolution of 256×3 filters then will get the $56 \times 56 \times 256$ output will apply the max pool and get the $28 \times 28 \times 256$, will apply convolution again of 512×3 filters, after applying max pool get the exact half, i.e., $14 \times 14 \times 512$ after applying convolution again of 512×3 and get the stable output of $14 \times 14 \times 512$ then applying max pool we get $7 \times 7 \times 512$ after this will flatten the same output. So the positive side of VGG is making the network uniform so after calculating get around 138 M parameters (Fig. 3).

3.3 Class Activation Mapping

Class activation maps are a basic technique used by CNN to define a particular class in the image to obtain the discriminative image regions. Basically CNN refers to the filtering process that happens in the normal type of network.

Numbers of convolutional layers are present in a network and just before the final output layer, execute global average pooling, i.e., before the fully connected layer with the activation function. Softmax the features that are inputted to this final output layer which then gives the expected output. It projects back the weights of the fully connected layer on the feature map. Then define the value of the image regions using this approach we called class activation mapping. The CNN uses these discriminative regions that are highlighted by CAM to identify the category for the given region.

The average of each unit feature map is generated by GAP in the previous convolutional layer which is the weighted summed up which results in concluding output. In the same way, to get the CAM, generate the weighted sum of the last convolutional layer.

3.4 NIH Chest X-Ray Dataset

This section briefly describes the newly presented dataset by National Institution of Health (NIH) and also published the paper regarding the huge dataset[1]. This NIH chest X-ray dataset consists of 110,000 grayscale X-ray images having 14 different categories of diseases labeled collected from 30,000 unique patients. For the creation of this dataset, the authors used natural language processing (NLP) for text mining from various radiological reports. Each and every image in the dataset is having corresponding information Patient age, ID, Gender, and a number of follow up visits to the hospitals. The labels of the image include atelectasis, mass, nodule, pneumonia, cardiomegaly, cardiomegaly, consolidation, edema, emphysema, hernia, fibrosis, pleural thickening, pneumothorax, and last no finding [9].

Since the dataset is not balanced, we are only considering the images and its corresponding labels that is dropping all the information provided in the dataset where each image in the dataset is of 1024×1024 pixels of one channel and due to the images and the feed to the model.

Total 1500 Images are taken where we splitted it into 80–20 in the training and testing purpose. 1200 for training and 250 for testing and 50 for validation (Figs. 4, 5, 6 and 7).

8 out of 14 classes and their X-ray images

For full dataset it can be found here

Fig. 4 Labeled disease with atelectasis and cardiomegaly

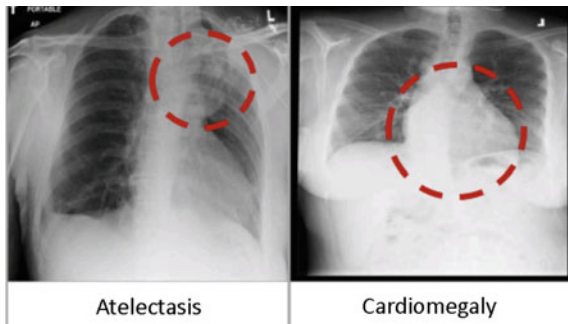


Fig. 5 Labeled disease with mass and nodule

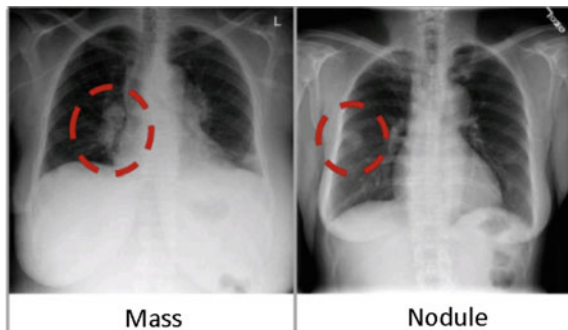


Fig. 6 Labeled disease with effusion and infiltration

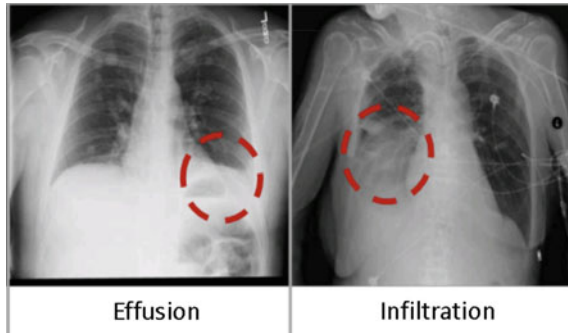
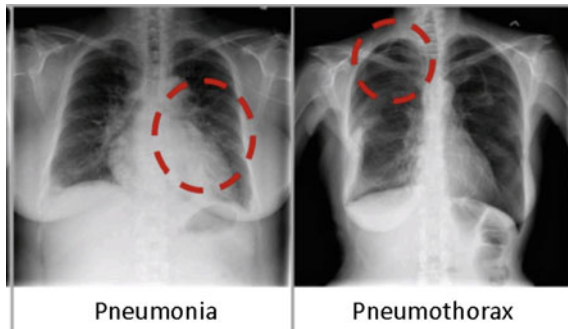


Fig. 7 Labeled disease with pneumonia and pneumothorax



4 Methodology

Classifying the thoracic diseases and recognizing them is a multi-label classification problem. In this, we have used a custom VGG-16 model that is with the help of transfer learning we classified diseases into 14 different categories and on NIH chest-ray dataset and also localized images that are identifying the important regions in an image with the help of class activation mapping.

4.1 Preprocessing

The dataset consists of around 1 lakh images of different pixels, and most of them were of 1024×1024 pixels in PNG format. We resized them into 224×224 pixels which is the default size required by the VGG-16 model. The NIH chest X-ray is a very large dataset, so for training, testing, and validation, we used (1%) of the total dataset and also lack computational requirements. Since the dataset was other fuels like patient id, age, etc., we have dropped all those columns and only used image path columns, we converted all the images corresponding labels into binary to increase the accuracy (Fig. 8).

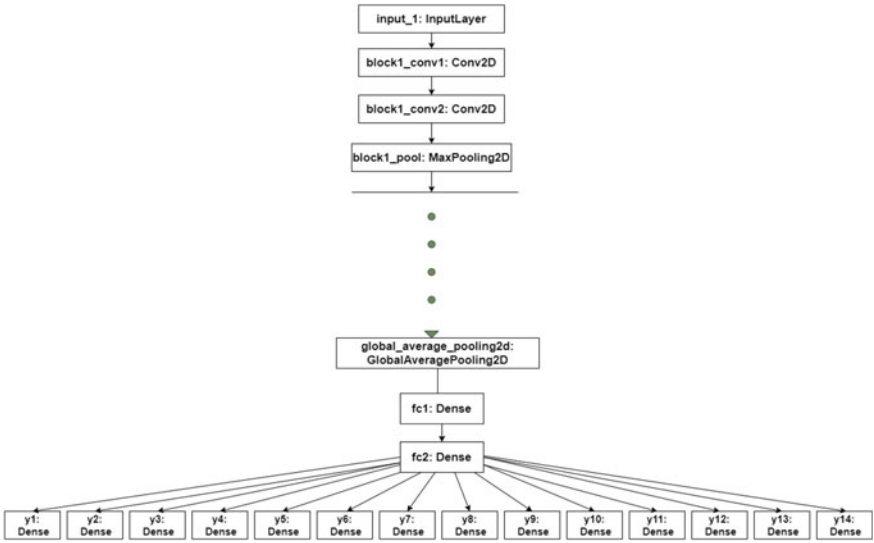


Fig. 8 Custom VGG-16 model architecture having two fully connected dense layers and one GAP layer

4.2 Model Architecture

The input image to our classification model is an RGB scaled image of 244×224 pixels in PNG format having three channels. The output has 14 classes described above in Sect. 3. The model is done into three stages: firstly the data flows initially to the entry phase after that it passes through the middle flow and finally through the exit flow. We have fine-tuned the model by adding the two dense classification layers with loss function as binary cross-entropy that will perform multi-label classification on our dataset.

5 Experimentation and Results

This section describes the details of our experimentation and results. In this experimentation, we have used 1 percent of the entire NIH chest X-ray images, that is, it consisted of 1,10,000 images. We used 1500 images and split them into 80 training data and 20 percent for testing data. As we have seen in the above section, there are royally three stages: the whole model to train and finally the diseases into different categories.

For the classification of diseases in the last layers for measuring losses, we have used binary cross-entropy loss function and ReLU as an activation function. Since we converted the dataset into binary form for getting high accuracy for 14 different classes. We only got a few diseases with higher accuracy scores but it would definitely increase when trained on a large dataset. Below the accuracy table for all 14 diseases and an AUC curve graph for measuring the accuracy levels, the entire experimentation was carried out on Google Colab GPU using the TensorFlow library. For training the model, we trained for 10 epochs and each epoch took 15–20 minutes and the total training time was 2 h. Since we lack computation power, we used a 1 percent dataset, but for improving the accuracy, our models can be used which can be found here.

After training, we also localized images using class activation mapping whether the model has rightly identified the feature extracted (Figs. 9, 10, and 11).

Observation

S. No.	Classification	Prediction
1	Atelectasis	91.3
2	Cardiomegaly	97.8
3	Consolidation	96.2
4	Edema	97.8
5	Effusion	89.7
6	Emphysema	98.6
7	Fibrosis	98.6
8	Hemia	99.1
9	Infiltration	85.01
10	Mass	96.4
11	Nodule	9.6
12	Pleural thickening	97.28
13	Pneumonia	99.4
14	Pneumothorax	96.2

Sample Image 1

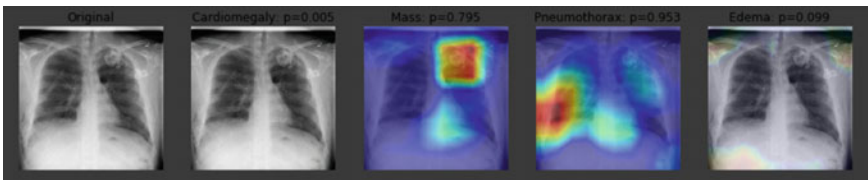


Fig. 9 Localization of mass and pneumothorax using CAM

Sample Image 2

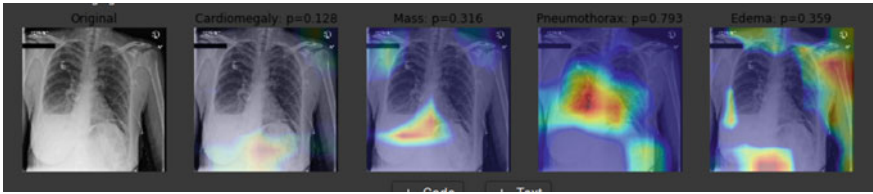


Fig. 10 Localization of mass, edema, and pneumothorax using CAM

Sample Image 3

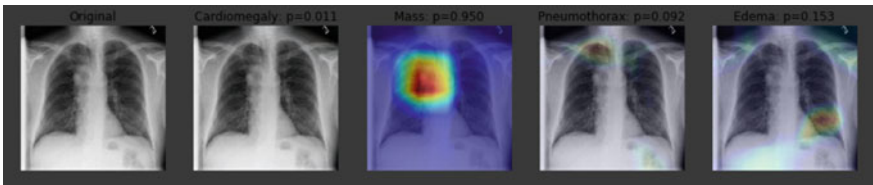


Fig. 11 Localization of mass using CAM

6 Conclusion

In the present paper, a transfer learning approach is used that is by building a custom-based VGG-16 model for the classification of chest X-ray (CXR) diseases. For identifying important regions in an image, we used a localization technique known as class activation mapping and got pretty good results. For the evaluation of our model, we evaluated it with the help of evaluating the function. Since we only used 1% of the dataset due to less computational power, our model can be reused and with the help of GPU, we can achieve better results. As mentioned in the observation table, the diseases with highest percent prediction are pneumonia and hernia with 99% accuracy and the lowest percent diseases that are predicted are infiltration and effusion with 85 and 89% accuracy. The percentage observed for other diseases like pneumothorax, mass, consolidation, cardiomegaly, edema, pleural thickening, emphysema, and fibrosis is 96, 97, and 98% accuracy. Working on enhancing the accuracy and more number of disease predictions in the future.

References

1. X. Wang, et al., Chestx-ray8: hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2017)
2. P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, T. Duan, D. Ding, A. Bagul, R.L. Ball, C. Langlotz, K. Shpanskaya, M.P. Lungren, A.Y. Ng, *CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning*
3. A. Nair, et al., *Detection of Diseases on Chest X-ray Using Deep Learning* (2019)
4. J. Cohen, P.B. Paul, V. Frappier, *Chester: A Web-Delivered Locally Computed Chest X-Ray Disease Prediction System*. [arXiv:1901.11210](https://arxiv.org/abs/1901.11210) (2019)
5. L. Yao, et al., Learning to diagnose from scratch by exploiting dependencies among labels. [arXiv:1710.10501](https://arxiv.org/abs/1710.10501) (2017)
6. J.I.Z. Chen, P. Hengjinda, Early prediction of coronary artery disease (CAD) by machine learning method—a comparative study. *J. Artif. Intell.* **3**(01), 17–33 (2021)
7. J. Tan, L. Jing, Y. Huo, L. Li, O. Akin, Y. Tian, Lgan: Lung segmentation in CT scans using generative adversarial network. *Comput. Med. Imaging Graph.* **87**, 101817 (2021)
8. Samuel Manoharan, Early diagnosis of lung cancer with probability of malignancy calculation and automatic segmentation of lung CT scan images. *J. Innov. Image Process. (JIIP)* **2**(04), 175–186 (2020)
9. National Institutes of Health, *NIH Clinical Center Provides One of the Largest Publicly Available Chest X-ray Datasets to the Scientific Community* (2017)
10. R. Selvaraju, et al., Grad-cam: Visual explanations from deep networks via gradient-based localization, in *Proceedings Of the IEEE International Conference on Computer Vision* (2017)
11. I. Allaouzi, M.B. Ahmed, A novel approach for multi-label chest X-ray classification of common thorax diseases. *IEEE Access* **7**, 64279–64288 (2019)
12. M. Fiszman, et al., Automatic detection of acute bacterial pneumonia from chest X-ray reports. *J. Am. Med. Inform. Assoc.* **7**(6), 593–604 (2000)
13. J. Hsu, P. Lu, K. Khosla, *Predicting Thorax Diseases with NIH Chest X-Rays* (2017)
14. M.T. Islam, et al., Abnormality detection and localization in chest x-rays using deep convolutional neural networks. [arXiv:1705.09850](https://arxiv.org/abs/1705.09850) (2017)
15. G. Litjens, et al., A survey on deep learning in medical image analysis. *Med. Image Anal.* **42**, 60–88 (2017)
16. R.H. Abiyev, M.K.S. Ma’aitah, Deep convolutional neural networks for chest disease detection. *J. Healthcare Eng.* **2018** (2018)

Experimental Evaluation of Adder Circuits on IBM QX Hardware



Divyanshu Singh, Simran Jakhodia, and Babita Jajodia

Abstract This work experimentally evaluated the performance of quantum adders on various IBM quantum hardware. The authors have constructed quantum circuits for one-qubit and two-qubit quantum adders using Quantum Information Science Kit (Qiskit) and run the circuit on seven IBM quantum devices: YorkTown, Ourense, Valencia, Santiago, Athens, Vigo and Melbourne. A detailed experimental analysis of accuracy rate of seven IBM devices are reported in this work. Experimental analysis shows that IBM Athens (5 qubits) provides the best accuracy results (73.7%) in comparison to high-qubit IBM Melbourne (15 qubits) for one-qubit quantum adder. Experimental analysis shows that IBM Melbourne (15 qubits), the only real IBM quantum hardware presently available with qubits higher than 5 qubits, provides an accuracy of 12.8125% over the ideal simulator.

Keywords Quantum adder · Quantum computing · Quantum gates · Quantum information science kit (Qiskit) · Ibm quantum experience (QX) · Quantum systems

1 Introduction

Quantum computing utilizes the properties of quantum entanglement and quantum superposition inherent to quantum mechanics; that is why this is rapidly gaining ground to overcome the limitations of classical computing [1]. Shor's algorithm [2] solving the integer factorization problem in a polynomial time and Grover's

D. Singh

Gautam Buddha University, Greater Noida, Uttar Pradesh, India
e-mail: 19bhp011@gbu.ac.in

S. Jakhodia · B. Jajodia (✉)

Indian Institute of Information Technology Guwahati, Guwahati, India
e-mail: babita@iiitg.ac.in

S. Jakhodia

e-mail: simran.jakhodia@iiitg.ac.in

algorithm [3] making it possible to substantially speed up the search in unstructured databases are one some of the best-known examples of the astounding properties of quantum computing.

Quantum additions form the fundamental component of quantum arithmetic and are applicable to high-computational computations. Quantum addition was the brain-child of Thomas G. Draper [4] in 1998, and he introduced quantum adder circuits. The authors in [5, 6] explored the implementation of quantum addition circuits using quantum gates [7, 8], but at the cost of higher number of qubits. This work proposes quantum addition circuits and demonstrate one-qubit and two-qubit quantum addition circuits implemented using quantum gates (CX, CCX, H and X gates) based on the conventional digital logic. An experimental evaluation study and detailed analysis of one-qubit and two-qubit quantum adders will be discussed by its execution on real IBM quantum devices: YorkTown, Ourense, Valencia, Santiago, Athens, Vigo and Melbourne. This work infer and derive accuracy rates of quantum circuits on various IBM real quantum systems and draw conclusions based on their performance of accuracy.

The rest of the paper is organized as follows: Sect. 2 provides a brief background related to one-bit and two-bit adders, single-qubit quantum states and quantum gates. Section 3 discusses about Quantum Adder circuits, its representation using quantum gates and the working principle of one-qubit and two-qubit adders. Section 4 discusses the evaluation study of executing an illustrative Quantum Adder (an one-qubit and a two-qubit adder) on various quantum hardware, followed by analysis of circuits on each device about its performance and accuracy. This is followed by conclusion in Sect. 5.

2 Background

2.1 One-Bit Adder and Two-Bit Adder

One-Bit Adder An one-bit adder adds two one-bit binary numbers (A and B) as inputs and generates a two-bit binary numbers (SUM) as output. This adder can be designed using a half adder with inputs as A, B each of one bit and outputs as SUM (the concatenation of carry out and sum bit generated by the half adder) of two bits.

Table 1 shows the truth table of one-bit adder for every possible combinations of inputs. The Boolean equation defining the operation of one-bit adder can be given by

$$\begin{aligned} \text{sum}_0 &= a_0 b_0 \\ \text{sum}_1 &= a_0 \oplus b_0 \end{aligned} \tag{1}$$

Two-Bit Adder A two-bit adder adds two two-bit binary numbers (A and B) as inputs and generates a three-bit binary numbers (SUM) as output. This adder can be designed using a combination of half adder and full adder with inputs as A, B each

Table 1 Truth table of one-bit adder

Inputs		Outputs	
$A(a_0)$		$B(b_0)$	
a_0		b_0	
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

of two bits and outputs as SUM (the concatenation of carry out of full adder and sum bits generated by the half adder and full adder) of three bits. The operation of the two-bit adder can be defined by Boolean equation as follows:

$$\begin{aligned}
 \text{sum}_0 &= a_0b_0 + a_1b_1(a_0 + b_0) \\
 \text{sum}_1 &= a_1b_1 \oplus (a_0 \oplus b_0) \\
 \text{sum}_2 &= a_1 \oplus b_1
 \end{aligned}
 \tag{2}$$

following the truth table of two-bit adder for every possible input combinations (Table 2).

Table 2 Truth table of two-bit adder

Inputs				Outputs		
$A(a_0a_1)$		$B(b_0b_1)$		$SUM(\text{sum}_0\text{sum}_1\text{sum}_2)$		
a_0	a_1	b_0	b_1	sum_0	sum_1	sum_2
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	0	0	1	0
0	0	1	1	0	1	1
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	0	1	1
0	1	1	1	1	0	0
1	0	0	0	0	1	0
1	0	0	1	0	1	1
1	0	1	0	1	0	0
1	0	1	1	1	0	1
1	1	0	0	0	1	1
1	1	0	1	1	0	0
1	1	1	0	1	0	1
1	1	1	1	1	1	0

2.2 Single-Qubit Quantum States [9]

A general quantum state can be written in the following form:

$$|\psi\rangle = \alpha |0\rangle + \beta |1\rangle \quad (3)$$

where α and β are complex numbers. This can be given in vector representation as:

$$|\psi\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \quad (4)$$

Here, this only requires two real numbers to describe a single-qubit quantum state. A convenient representation is

$$|\psi\rangle = \cos(\theta/2) |0\rangle + \sin(\theta/2)e^{i\phi} |1\rangle \quad (5)$$

where $0 \leq \phi < 2\pi$, and $0 \leq \theta \leq \pi$. From this, it is clear that there is a one-to-one correspondence between qubit states (\mathbb{C}^2) and the points on the surface of a unit sphere (\mathbb{R}^3).

2.3 Quantum Gates

This subsection briefly introduce quantum gates that are later used in the development of quantum circuits. Quantum gates/operations are usually represented as matrices. A gate which acts on a qubit is represented by a 2×2 unitary matrix U . The action of the quantum gate is found by multiplying the matrix representing the gate with the vector which represents the quantum state as:

$$|\psi'\rangle = U |\psi\rangle \quad (6)$$

A general unitary must be able to take the $|0\rangle$ to the above state. That is

$$U = \begin{pmatrix} \cos(\theta/2) & a \\ e^{i\phi} \sin(\theta/2) & b \end{pmatrix} \quad (7)$$

where a and b are complex numbers constrained such that $U^\dagger U = I$ for all $0 \leq \theta \leq \pi$ and $0 \leq \phi < 2\pi$. This gives three constraints and as such $a \rightarrow -e^{i\lambda} \sin(\theta/2)$ and $b \rightarrow e^{i\lambda+i\phi} \cos(\theta/2)$ where $0 \leq \lambda < 2\pi$ giving

$$U = \begin{pmatrix} \cos(\theta/2) & -e^{i\lambda} \sin(\theta/2) \\ e^{i\phi} \sin(\theta/2) & e^{i\lambda+i\phi} \cos(\theta/2) \end{pmatrix} \quad (8)$$

This is the most general form of a single qubit unitary. An important feature of quantum circuits is that, between initialising the qubits and measuring them, the operations (gates) are always reversible. These reversible gates can be represented as matrices.

The matrix representations of quantum gates using the computational basis of $|0\rangle$ and $|1\rangle$ are as follows:

H Gate (which Puts Qubits in Superposition State)

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = u2(0, \pi) \quad \text{---[H]---} \tag{9}$$

X Gate (which Flips the State of Qubits)

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = u3(\pi, 0, \pi) \quad \text{---[X]---} \tag{10}$$

CX Gate (which Flips the State of Qubit When Control Qubit Is 1)

$$CX = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \quad \begin{array}{c} \bullet \\ | \\ \oplus \end{array} \tag{11}$$

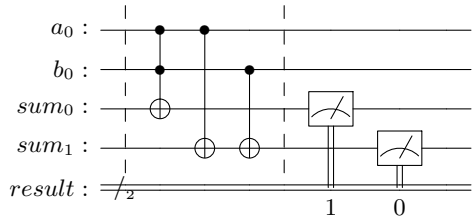
CCX Gate (which Have More Than One Control Qubits)

$$CCX = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix} \quad \begin{array}{c} \bullet \\ | \\ \bullet \\ | \\ \oplus \end{array} \tag{12}$$

3 Proposed Work

This work is related to the approach of designing adders using quantum gates. In this work, the authors have classified the adder circuits on the basis of number of qubits used to add.

Fig. 1 Circuit Diagram of One-Qubit Adder



3.1 One-Qubit Adder Circuits

Figure 1 shows the implementation of quantum circuit for the addition of one qubits with a_0 and b_0 as the input qubits and sum_0 and sum_1 as the output qubits, respectively. This circuit is based on the basic concept of implementation of half-adder using digital design circuit-building logic explained in the truth table (Table 1). This type of adder circuit takes one-qubit each as inputs and add them. This circuit is also known as half adder circuit because this does not have any carry qubit as an input. (Note: $result_0$ and $result_1$ are the measured classical outputs of sum_0 and sum_1 respectively once the quantum state has collapsed into classical state after measurement operator).

3.2 Two-Qubit Adder Circuits

Figure 2 shows the implementation of two-qubit quantum adder circuit for the addition of two qubits with a_0, a_1 and b_0, b_1 as the input qubits and sum_0, sum_1, sum_2 as the output qubits, respectively. It takes two inputs each of two qubits, add them

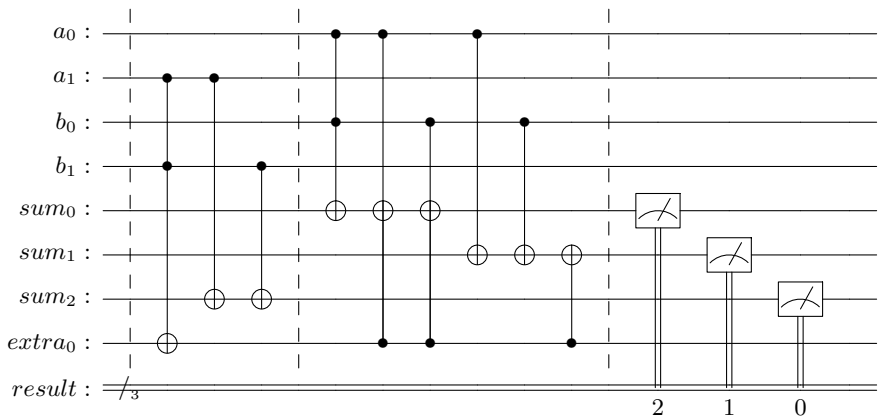


Fig. 2 Circuit diagram of two-qubit adder

and return three-qubits output. This circuit is based on the basic concept of implementation of adder circuit using digital design circuit-building logic explained in the truth table (Table 2). Note: $extra_0$ is the ancilla qubit; $result_0$, $result_1$ and $result_2$ are the measured classical outputs of sum_0 , sum_1 and sum_2 respectively once the quantum state has collapsed into classical state after measurement operator.

4 Experiments of Quantum Addition Circuits on IBM QX Hardware

After construction of the one-qubit adder (Fig. 1) and two-qubit adder (Fig. 2) as presented in Sect. 3 using Qiskit package [1] offered by IBM [10], experiments were conducted on seven different IBM quantum devices and one Quantum Assembly Language (QASM) Simulator. Although IBM presently provide eight real quantum systems, the eighth IBM system `ibmq_armonk` is not considered for experiments as it supports only one qubit. Moreover, experiments on two-qubit adder (Fig. 2) were performed only on IBM Melbourne since the other seven IBM systems have limitations of five qubits. IBM Melbourne is the only IBM quantum system that consists of more than five qubits and a maximum of fifteen qubits. The authors have considered 1024 shots in ten runs on IBM devices. Results from each run are used to compute the probabilities of all the possible states and helped to determine the accuracy of quantum circuit on different IBM Quantum systems.

The accuracy rate can be calculated by taking the summation of all probabilities of all expected output states divided by the number of all possible input states. Mathematically, the accuracy rate can be calculated as

$$\text{accuracy_rate} = \frac{1}{2^{2N}} \sum_{j=1}^{2^{N+1}} \sum_{i=1}^{2^{2N}} p_{\text{ideal}}^{(i,j)} \times p_{\text{real_hw}}^{(i,j)} \tag{13}$$

where, p_{ideal} and $p_{\text{real_hw}}$ are the probabilities of the ideal simulator and that of the real quantum hardware respectively for N-qubit adders; i and j represents the possible input states and output states respectively.

The experimental analysis of accuracy rate on one-qubit and two-qubit adder circuits over different IBM Systems are as follows.

4.1 Analysis of One-Qubit Adder Circuits

The authors have created all possible inputs states using X gates, tested the output and then analysed the results by plotting histogram. The histograms are shown in Fig. 3, 4, 5, 6, 7, 8, 9 and 10.

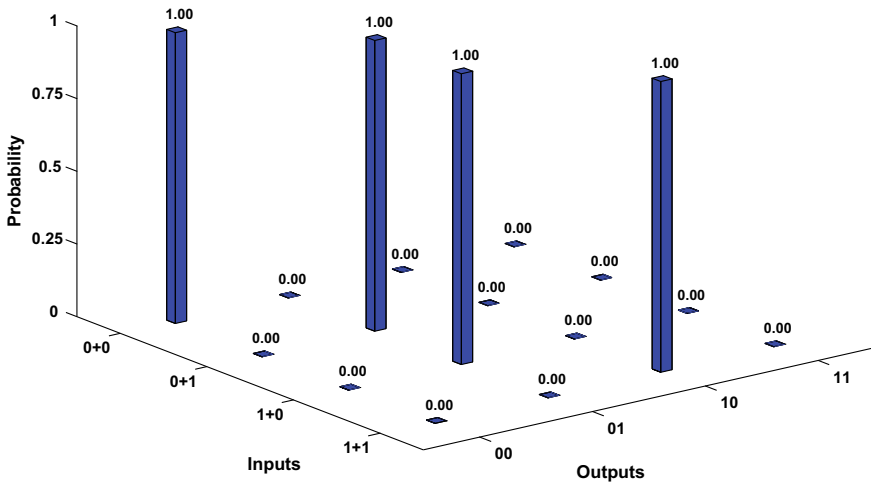


Fig. 3 Experimental results of one-qubit adder on IBM quantum device: IBM QASM (32 qubits) [10]

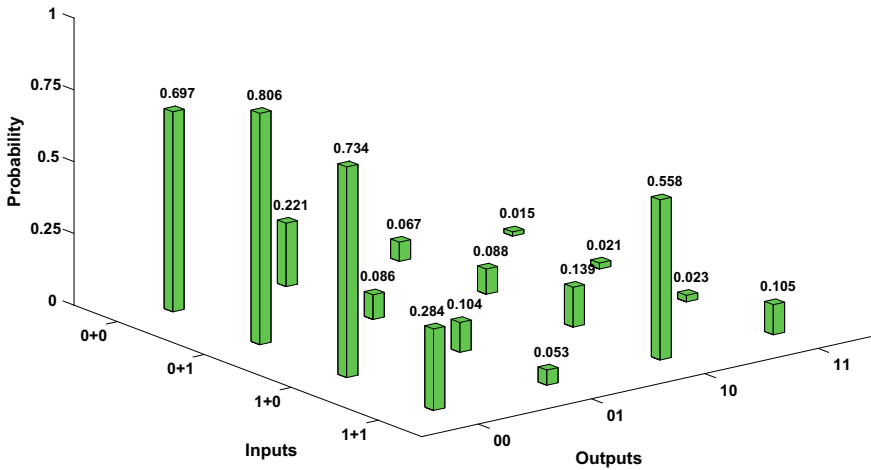


Fig. 4 Experimental results of one-qubit adder on IBM quantum device: IBM YorkTown (5 qubits) [10]

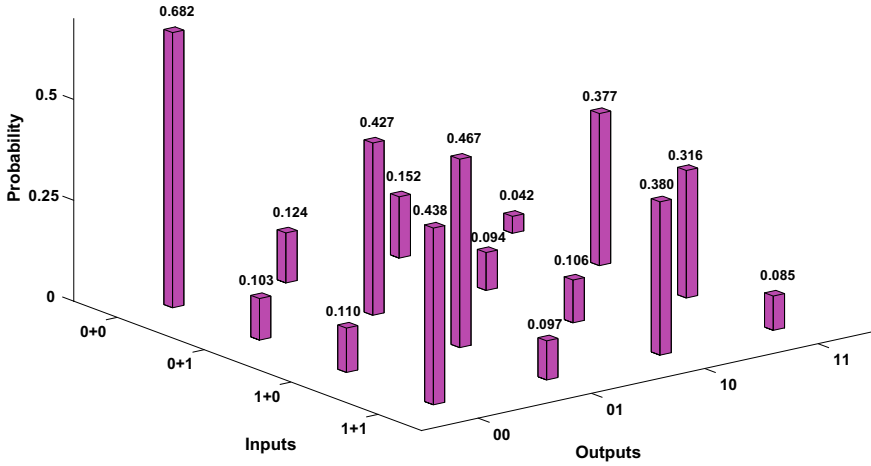


Fig. 5 Experimental results of one-qubit adder on IBM quantum device: IBM Ourense (5 qubits) [10]

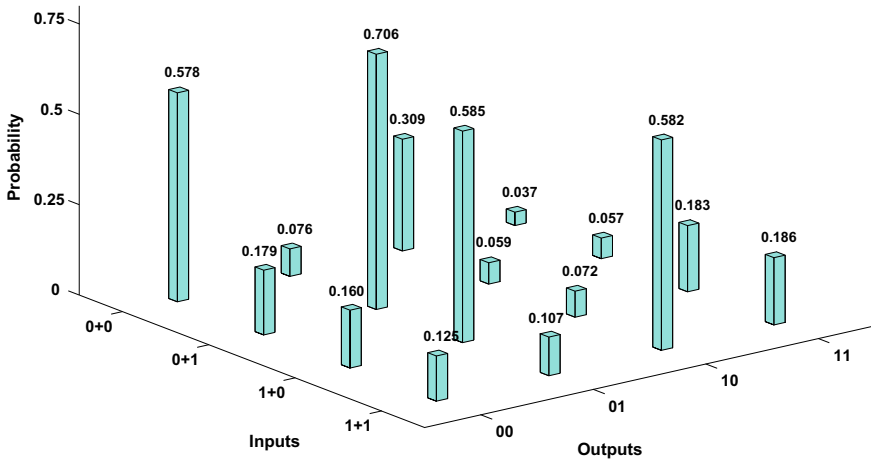


Fig. 6 Experimental results of one-qubit adder on IBM quantum device: IBM Valencia (5 qubits) [10]

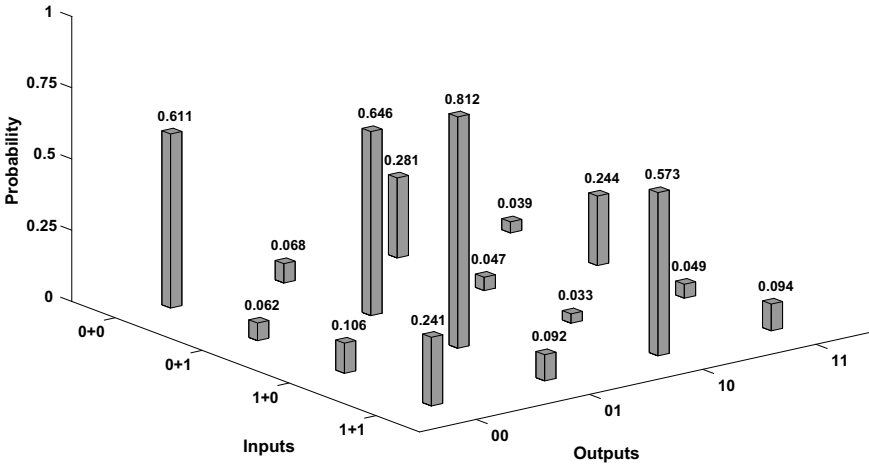


Fig. 7 Experimental results of one-qubit adder on IBM quantum device: IBM Santiago (5 qubits) [10]

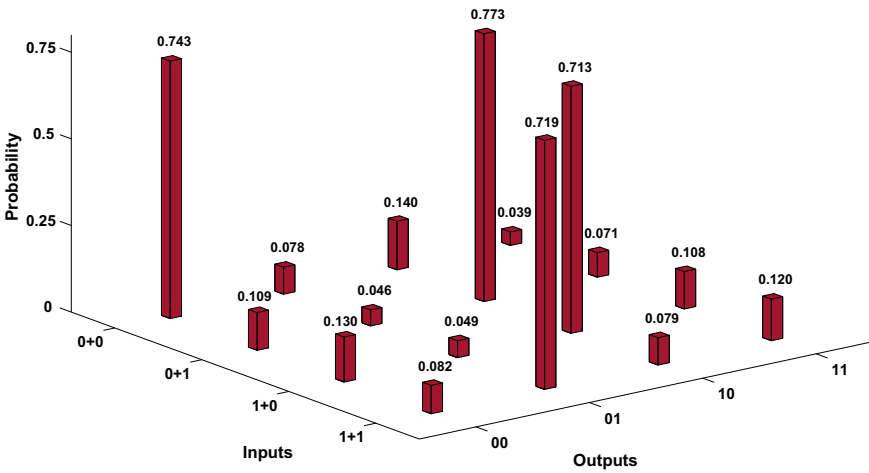


Fig. 8 Experimental results of one-qubit adder on IBM quantum device: IBM Athens (5 qubits) [10]

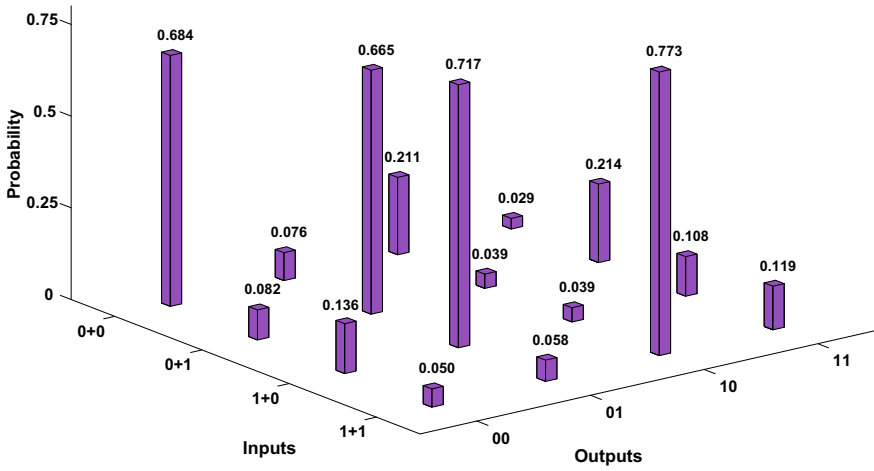


Fig. 9 Experimental results of one-qubit adder on IBM quantum device: IBM Vigo (5 qubits) [10]

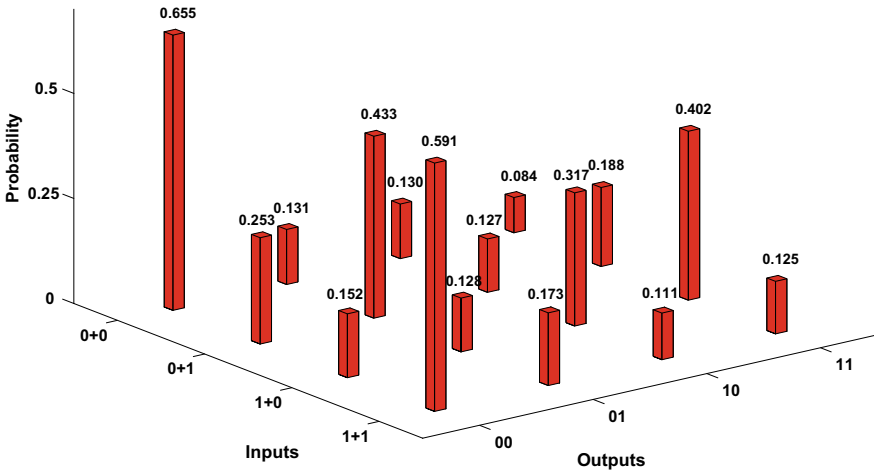


Fig. 10 Experimental results of one-qubit adder on IBM quantum device: IBM Melbourne (15 qubits) [10]

Table 3 Accuracy rate of one-qubit adder on real IBM quantum devices [10]

S. No.	IBM quantum devices	Accuracy rate (in %)
1.	IBM QASM Simulator (32 qubits)	100
2.	IBM YorkTown (5 qubits)	36.125
3.	IBM Ourense (5 qubits)	48.9
4.	IBM Valencia (5 qubits)	61.275
5.	IBM Santiago (5 qubits)	66.05
6.	IBM Athens (5 qubits)	73.7
7.	IBM Vigo (5 qubits)	70.975
8.	IBM Melbourne (15 qubits)	33.175



Fig. 11 Experimental results of two-qubit adder on IBM quantum device: IBM QASM (32 qubits) [10]

By comparing the results of histogram with the truth table (Table 1) according to expected outputs, the accuracy is calculated. The accuracy rate of one-qubit adder circuit on different Quantum Systems are given in Table 3. The QASM simulator is the ideal simulator, so it gives 100% accuracy. Other systems like IBM Athens provide better accuracy results (73.7%) in comparison to high-qubit IBM Melbourne with an accuracy rate (33.175%). Note: The text colour in Table 3 is referred as same as the colour of histograms.

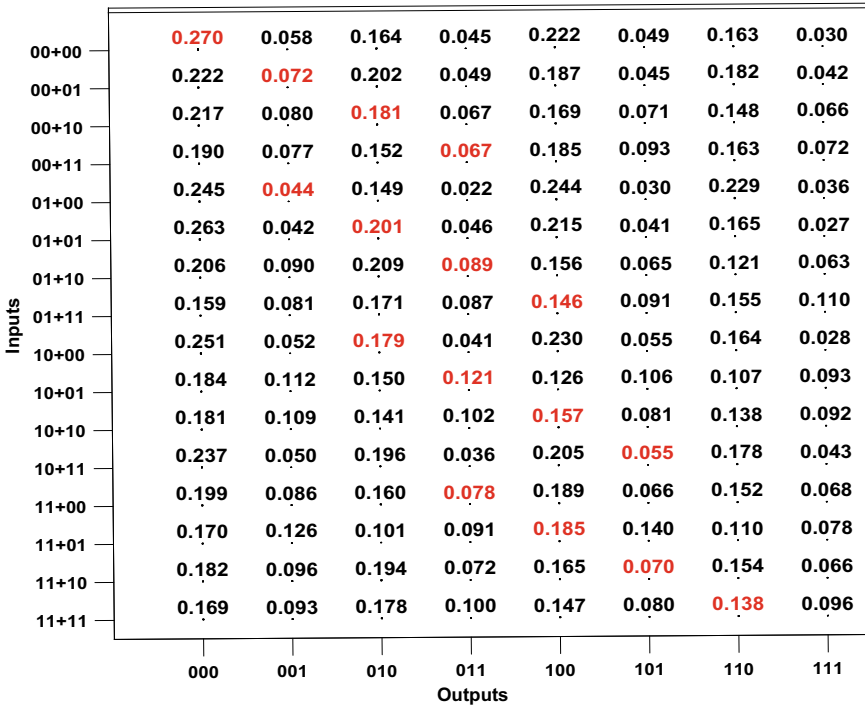


Fig. 12 Experimental results of two-qubit adder on IBM quantum device: IBM Melbourne (15 qubits) [10]

4.2 Analysis of Two-Qubit Adder Circuits

The authors have created all possible inputs states using X gates, tested the output and then analysed the results by plotting histograms. Figures 11 and 12 show the output probabilities of different possible output states with reference to input states as per the truth table (Table 2). Table 4 provides accuracy results of two-qubit adders. Table 4 clearly illustrates that IBM Melbourne (15 qubits) provides an accuracy of 12.8125% over the ideal simulator.

This two-qubit adder circuit is tested on real IBM Melbourne systems [10] only having 15 qubits, as the minimum number of qubits required are eight qubits (Fig. 2). This was the only limitation of which authors could present simulation results on two-qubit adder circuit for real IBM Melbourne (15 qubits) and the ideal QASM simulator.

Table 4 Accuracy rate of two-qubit adder on real IBM quantum devices [10]

S. No	IBM quantum devices	Accuracy rate (in %)
1.	IBM QASM Simulator (32 qubits)	100
2.	IBM Melbourne (15 qubits)	12.8125

5 Conclusions

This work discusses the experimental evaluation of the performance of one-qubit and two-qubit quantum adder circuits on seven IBM Quantum Experience hardwares (YorkTown, Ourense, Valencia, Santiago, Athens, Vigo and Melbourne). The authors developed quantum circuits for representation of one-qubit and two-qubit quantum adders using Qiskit for simulating quantum computation. The proposed quantum adder circuit was set to run on various real quantum hardware, and the performance was compared against accuracy rate of different quantum hardwares. The reason why these particular IBM quantum devices are performing better than others is that they both have less gate error rate and have less noise in compared to other devices. The reason why the authors decided to select the conventional design of circuit is that to keep it more general adder circuit and more focus on experimental evaluation of IBM quantum devices. The graphs in this work represent the probability distribution on different IBM quantum devices. These graphs do not represent the performance of IBM quantum devices. Authors decided to use the conventional adder circuit to show the performance of different IBM Quantum Devices.

The authors will consider error mitigation techniques to manage various types of hardware noise to improve the simulation performance of quantum adders, and also add different approach to the adder circuit to improve the quality of circuit as their future works and investigate techniques to develop circuits with less number of quantum gates (depth) as these can result in less noisy results.

References

1. Learn Quantum Computation using Qiskit. Available at <https://qiskit.org/textbook/preface.html>. Accessed on April 20, 2021
2. P.W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.* **26**(5), 1484–1509 (1997)
3. L.K. Grover, A fast quantum mechanical algorithm for database search, in *STOC'96* (1996). <https://arxiv.org/abs/quant-ph/9605043>
4. T.G. Draper, *Addition on a Quantum Computer*. Available at <https://arxiv.org/pdf/quant-ph/0008033.pdf> (2000)
5. E.H. Shaik, N. Rangaswamy, Implementation of quantum gates based logic circuits using IBM Qiskit, in *2020 5th International Conference on Computing, Communication and Security (ICCCS)*, pp. 1–6 (2020)

6. W. Methachawalit, P. Chongstitvatana, Adder circuit on ibm universal quantum computers, in *2020 17th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, pp. 92–95 (2020)
7. M.A. Nielsen, I.L. Chuang, *Quantum Computation and Quantum Information*, 10th edn. Cambridge University Press, USA (2011)
8. M.A. Sohel, N. Zia, M.A. Ali, N. Zia, Quantum computing based implementation of full adder, in *2020 IEEE International Conference for Innovation in Technology (INOCON)*, pp. 1–4 (2020)
9. Summary of Quantum Operations. https://qiskit.org/documentation/tutorials/circuits/3_summary_of_quantum_operations.html
10. IBM Quantum Backends. Available at <https://quantum-computing.ibm.com/docs/manage/backends/>. Accessed on June 5, 2021

Development of the InBan_CIDO Ontology by Reusing the Concepts Along with Detecting Overlapping Information



Archana Patel and Narayan C. Debnath

Abstract The COVID-19 pandemic is a global emergency that badly impacted the economies of various countries. COVID-19 hits India when the growth rate of the country was at the lowest in the last ten years. To semantically analyze the impact of this pandemic on the economy, it is curial to have an ontology. CIDO ontology is a well-standardized ontology that is specially designed to assess the impact of coronavirus disease and utilize its results for future decision forecasting for the government, industry experts, and professionals in the field of various domains like research, medical advancement, technical innovative adoptions, and so on. However, this ontology does not analyze the impact of the COVID-19 pandemic on the Indian banking sector. On the other side, COVID-19-IBO ontology has been developed to analyze the impact of the COVID-19 pandemic on the Indian banking sector, but this ontology does not reflect complete information of COVID-19 data. Resultantly, users cannot get all the relevant information about the COVID-19 and its impact on the Indian economy. This article aims to extend the CIDO ontology to show the impact of COVID-19 on the Indian economy sector by reusing the concepts from other data sources. We also provide a simplified schema matching approach that detects the overlapping information among the ontologies. The experimental analysis proves that the proposed approach has reasonable results.

Keywords COVID-19 · Ontology · CIDO · COVID-19-IBO · IRI · Schema matching · Banking sector

1 Introduction

The new business models and the different government regulatory interventions during the COVID-19 pandemic have led to various behavioral and structural changes

A. Patel (✉) · N. C. Debnath

Department of Software Engineering, School of Computing and Information Technology, Eastern International University, Binh Duong, Vietnam

N. C. Debnath

e-mail: narayan.debnath@eiu.edu.vn

in peoples' lives across the globe. These changes cut across the width of the financial landscape thereby posing challenges to the key functions of the banking sector as well. In addition to other challenges, the major technological challenge is the inability to access systems and data because of various constraints. The existing artificial intelligence technologies should be leveraged to their full potential to enable data integration, analysis, and sharing for remote operations. Especially, for financial institutions like the banking sector, artificial intelligence could offer better data integration and sharing through semantic knowledge representation approaches. Many reports containing plenty of data stating the effects of this pandemic on the banking sector are available in the public domain as listed below:

- <https://dea.gov.in/>
- <https://finmin.nic.in/>
- <https://www.indiabudget.gov.in/economicsurvey/>
- <https://www.who.int/emergencies/diseases/>
- <https://www.mygov.in/covid-19/>

These Web sites offer a static representation of COVID-19 data which is largely unstructured in nature (text, audio, video, image, newspaper, blogs, etc.) creating a major problem for the users to analyze, query, and visualize the data. The data integration task gets highly simplified by the incorporation of knowledge organization systems (taxonomy, vocabulary, ontology) as background knowledge. Storing the knowledge using the semantic data models enhances the inference power also. Ontology is a data model that is very useful to enable knowledge transfer and interoperability between heterogeneous entities due to the following reasons: (i) They simplify the knowledge sharing among the entities of the system; (ii) it is easier to reuse domain knowledge; and (iii) they provide a convenient way to manage and manipulate domain entities and their interrelationships. An ontology consists of a set of axioms (a statement is taken to be true, to act as a premise for further reasoning) that impose constraints or restrictions on sets of classes and relationships allowed among the classes. These axioms offer semantics because by using these axioms, machines can extract additional or hidden information based on data explicitly provided [1]. Web ontology language (OWL) is designed to encode rich knowledge about the entities and their relationships in a machine-understandable manner. OWL is based on the description logic (DL) which is a decidable subset (fragment) of first-order logic (FOL), and it has a model-theoretic semantics.

Various ontologies have been developed in OWL language to semantically analyze the COVID-19 data. CIDO ontology is a well-standardized ontology and specially designed for coronavirus disease [2]. However, this ontology does not analyze the impact of the COVID-19 pandemic on Indian banking sectors. On the other side, COVID-19-IBO ontology has been developed to analyze the impact of the COVID-19 pandemic on the Indian banking sector, but this ontology does not reflect the complete information of COVID-19 [3]. The contributions of this article are as follows:

- To extend the CIDO ontology to encode the rich knowledge about the impact of COVID-19 on the Indian banking sector

- To detect the overlapping information by designing a simplified schema matching approach

The rest of the part of this article is organized as follows: Sect. 2 illustrates the available COVID-19 ontologies. Section 3 focuses on the development of the proposed InBan_CIDO ontology. Section 4 shows the proposed schema matching approach, and the last section concludes this article.

2 A Glance on Available COVID-19 Ontologies

Ontology provides a way to encode human intelligence in a machine-understandable manner. For this reason, ontologies are used in every domain specifically in the emergency domain. As COVID-19 pandemic is a global emergency, various countries are badly impacted by it. India is one of the countries that is badly impacted by second wave of this pandemic. Various ontologies are offered to semantically analyze the COVID-19 data. These ontologies are listed below:

- IDO ontology [4] provides a strong foundation to the other ontologies; therefore, this ontology is extended by various ontologies, namely VIDO, CIDO, IDO-COVID-19. The extended ontologies only import those concepts or entities that are required as per domain.

The coronavirus infectious disease ontology (CIDO) is a community-based ontology that imports COVID-19 pandemic-related concepts from the IDO ontology [2]. CIDO is a more specific standard ontology as compared to IDO and encodes knowledge about the coronavirus disease as well as provides integration, sharing, and analysis of the information. The latest version of CIDO is released in May 2021.

- Dutta and DeBellis [5] published the COVID-19 ontology for case and patient information (named CODO) on the web as a knowledge graph that gives information about the COVID-19 pandemic as a data model. The CODO ontology's primary goal is to characterize COVID-19 cases and COVID-19 patient data.
- Mishra et al. [3] have developed an ontology called COVID-19-IBO ontology that semantically analyzes the impact of COVID-19 on the performance of the Indian banking sector. This is only one ontology that reflects the COVID-19 information as well as its impact on the Indian economy.

Table 1 shows the value of the metrics of the available COVID-19 ontologies. CIDO ontology has the highest number of classes as compared to other ontologies. These classes describe the concept of the domain. Properties (data and object) increase the richness of the ontology. The axioms-imposed restriction on the entities of the ontology provide an ability to semantically infer the information of the imposed queries from the ontology.

As of now, various methodologies for the development of ontology are proposed [6]. The four most famous ontology methodologies, namely TOVE, Enterprise model

Table 1 Metrics value of the available ontologies

Metrics	Ontologies					
	IDO	VIDO	CIDO	IDO-COVID-19	CODO	COVID-19-IBO
Axioms	4103	4647	104,336	5018	2047	313
Logical axioms count	833	956	17,367	1032	917	137
Class count	362	429	7564	486	91	105
Object property count	43	43	409	43	73	28
Data property count	0	0	18	0	50	43
Attribute richness	0.0	0.0	0.056	0.0	0.549	0.409
Inheritance richness	1.237	1.258	1.547	1.242	1.010	0.895
Relation richness	0.384	0.360	0.873	0.340	0.471	0.229

approach, METHONTOLOGY, and KBSI IDEF5 are presented in Fig. 1. It is quite clear that developing ontologies is not focused on understanding the engineering process, but it is a matter of craft skill. The selection of the methodologies heavily depends on the application and requirement of the ontology. Therefore, there is no perfect methodology available that can be used universally.

There are two ways to create an entity inside the ontology, (a) to reuse concepts from the available ontologies and (b) to create all concepts separately (without reusing concepts). It is good practice to reuse the existing concepts that offer a common understanding of the domain. With the help of IRI, we can reuse the concepts. IRI is internationalized resource identifier which avoids multiple interpretations of entities of ontology.

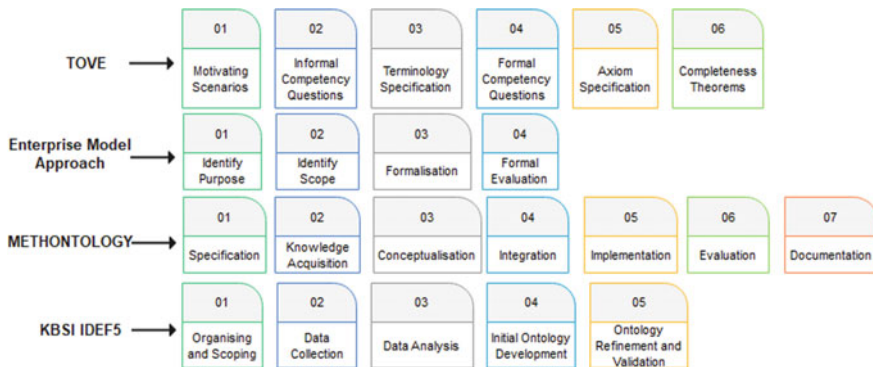


Fig. 1 Available methodologies and their steps

After studying the literature, we state that available ontologies contain detailed information about the COVID-19 disease. However, they do not have complete (or partial) information about the impact of COVID-19 on the Indian banking sector (the sector which plays a vigorous role in the growth of the Indian economy) along with COVID-19 data. The different COVID-19 ontologies create the heterogeneity problem that needs to resolve in order to achieve interoperability among the available ontologies.

3 InBan_CIDO Ontology

Ontology is a semantic model that represents the reality of a domain in a machine-understandable manner. The basic building block of an ontology is classes, relationships, axioms, and instances [7]. Nowadays, ontologies are used everywhere because of their ability to infer semantic information of the imposed queries. As, literature shows available ontologies are not capable to analyze the impact of COVID-19 on the Indian economy along with COVID-19 detailed information. Therefore, we extend the CIDO ontology to fill this gap. On behalf of the above-listed methodologies, we have selected five phases for the extension of the CIDO ontology (called InBan_CIDO) that will offer complete information about the impact of COVID-19 on the performance of the Indian banking sector along with information on the COVID-19 pandemic. These five phases are scope determination, extraction of the concept, organization of the concept, encoding, and evaluation. The detailed description of these phases is stated below:

- *Scope Determination*: The objective of this phase is to determine the scope of the ontology. We use competency questions to fix the scope and boundary of the proposed ontology. Some selected questions are mentioned below that are framed after the discussion with the expert of the domain.
 - (a) How to the central bank will tackle the situation that arises due to non-collection of debt recovery in the moratorium period during the COVID-19 lock down?
 - (b) What will be the impact on the balance sheet of banks when the NPA number will be added for the period of COVID-19?
 - (c) What is the cumulative impact on the banking industry due to the loss of other industries like aviation, tourism, marketing?
 - (d) How risk assessment and planning should be done for the upcoming COVID-19 wave (if any)?
 - (e) How effectively the risk assessment and mitigation mechanism worked during the first and second waves?
- *Concept Extraction*: This phase aims to extract the concepts or entities from the different sources as per the specified domain. These concepts can be marked as

classes, properties (data and object properties), and instances in the further phases. We use the following sources for the development of the InBan_CIDO ontology.

- (a) Research articles from various data sources and indexed by Scopus, SCI, etc.
- (b) Ontology repositories like OBO library, bio-portal
- (c) Ontologies like COVID-19-IBO
- (d) Databases provided by WHO and Indian government
- (e) Interview with the experts of the domain

We have extensively gone through these data sources and extracted all the entities that are required for the extension of CIDO ontology to fulfill the proposed scope. All the extracted entities are stored in the Excel sheet for further analysis.

- *Concept Organization*: This phase aims to organize the extracted concepts in a hierarchical manner. Firstly, we classified the extracted concepts, as classes, properties, and instances based on their characteristics. All the identified concepts are organized in a hierarchal manner (parent–child relationship). For example, a class *private bank* and a class *government bank* should become the subclasses of class *Bank*. We import some concepts inside InBan_CIDO ontology from the other ontologies like COVID-19-IBO. Some imported concepts are mentioned below:

Reused classes

Current_Challenging_conditions_of_banking_industry, NPA, Loan, Bank, InfectedFamilyMember, Scheduled_Banks, Doubtfull, ETB, Cooperative_Banks, Impact, Commercial_Banks, Financial, Employee, Bankers, Digital_optimization, Loan_repayment, NRE, Detect_probable_defaults_in_early_phase, New_Assets_Quality_Review, IndividualCurrentAccount, Robust_digital_channels, OnHuman, BankingRetailCenters, Contactless_banking_options, High_credit_risk, Private_Sector, Deposit, Policies

Reused data properties

ContainedIn, EmployeeID, hasBankingRelationship, has_status, has_temp_of_human, number_of_account, number_of_credit_card, number_of_loans_reported, sanctioned_strength, working_strength, has_date

Reused object properties

Return, negative_return, positive_return, hasStatistics, has_cause, has_close, has_gender, has_nationality, has_open, type_of_relationship, via_account, via_card, via_loan, via_insurance, city_wise_statistics

- *Encoding*: We encode the InBan_CIDO ontology by using Protégé 5.5.0 tool [8]. Protégé tool is freely available on the web, and it has a very interactive interface. User can encode the ontology in protégé without having any technical knowledge about any programming language. InBan_CIDO ontology has two types of classes: old classes and new classes. Old classes are those classes that already available in CIDO ontology (CIDO is the base ontology that we have extended). New classes are categorized into two groups:

- (a) Classes are imported inside InBan_CIDO from other available ontologies (source ontologies) by using the IRI of that ontology
- (b) New classes are added as per need just by creating classes under the thing class

The process of Importing Classes inside InBan_CIDO Ontology: For the reusability purpose, we import some classes in the InBan_CIDO ontology from the COVID-19-IBO ontology. The process to import the classes inside destination ontology is required the IRI of that source ontology where these classes are defined. After getting the IRI of the ontology, we open the Protégé tool and create the name of the class under the thing class (which is a default class) or any other classes as per need, then go to *the new entity option* (Fig. 2a) and click on *specified IRI* (Fig. 2b) and write the IRI of the source ontology where that concept defined and then click on the ok option. Now, the IRI of the class will be changed (Fig. 2c). For example, Fig. 2 shows the process to import the class *Person* in InBan_CIDO ontology from the FOAF ontology which is an upper ontology. The class *person* of COVID-19-IBO ontology is also imported from FOAF ontology [9].

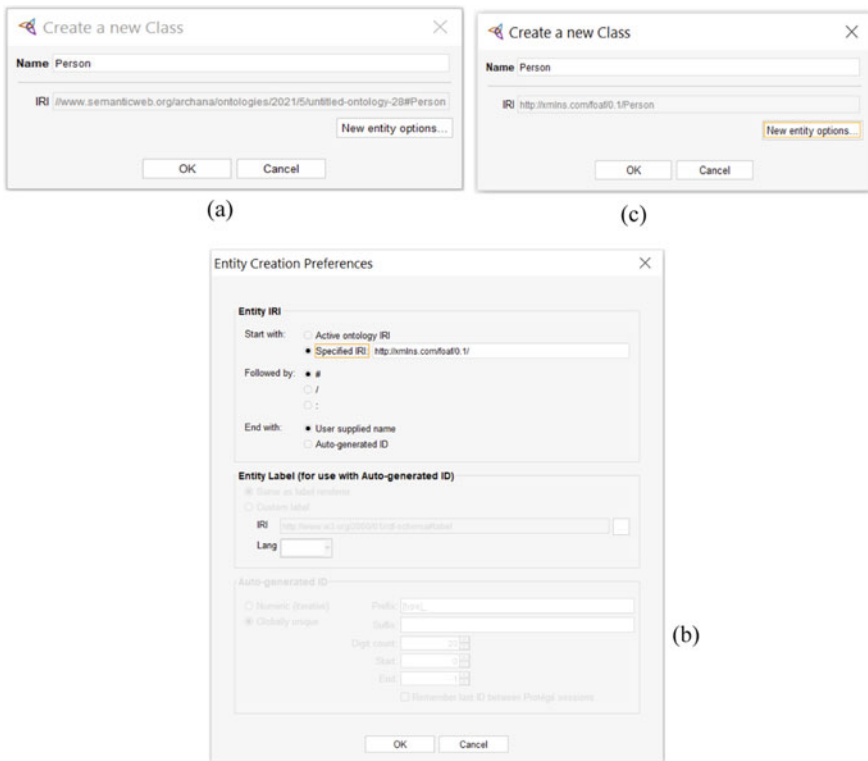


Fig. 2 Process to import the concepts in InBan_CIDO ontology

4 A Simplified Schema Matching Approach for COVID-19 Ontologies

The developed InBan_CIDO ontology contains information about the COVID-19 disease as well as information of the impact of COVID-19 on the performance of the Indian banking sector which is imported from COVID-19-IBO ontology. The COVID-19-IBO ontology has also contained information about the COVID19 disease as per need. Therefore, it is required to investigate the overlapping information from the existing COVID-19 ontologies with respect to COVID-19-IBO ontology. Matching systems use the matching algorithm and detect the relationships between the entities of the ontologies [11].

We propose a schema matching approach (SMA-COVID-19) to find out the overlapping information among the developed COVID-19 ontologies. The first step of the SMA-COVID-19 algorithm is to select two ontologies (O_S and O_T) from the ontology repository and then extract all the labels of the concepts in both ontologies with the help of Id and IRI. The labels of the concepts of source and target ontologies are stored in a n, m dimensional array ($a[n]$ and $b[m]$) separately where $n =$ number of classes in O_S and $m =$ number of classes in O_T . The SMA-COVID-19 algorithm picks one label of O_S and then matches it with all the labels of O_T . The matching between the labels is performed according to the Levenshtein and synonym matchers. If two labels are matched based on synonyms, then a 0.9 similarity value will be assigned to them. The matching result (similarity value between the labels) is stored in the $n \times m$ matrix ($Avg[n][m]$). All the pairs whose similarity value is greater than α (where α is a threshold for the similarity value) are considered to be correspondences.

Pseudocode for SMA-Covid19

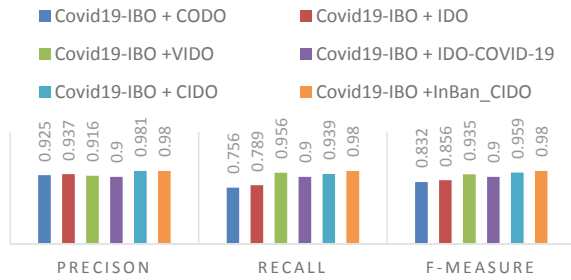
- Select O_S and O_T from the ontology's repository
- Extract all labels of both O_S and O_T ontologies

<pre> for (i=1, i ≤ n, i++) // n ∈ number of classes in O_S fetch Id of the ith concept if (Id has label) fetch and store label in a[i] else fetch IRI of the ith concept and find label by splitting IRI store label in a[i] </pre>	<pre> for (j=1, j ≤ m, j++) // m ∈ number of classes in O_T fetch Id of the jth concept if (Id has label) fetch and store label in b[j] else fetch IRI of the jth concept and find label by splitting IRI store label in b[j] </pre>
--	--

- **Matching**

<pre> for (i=1, i ≤ n, i++) for (j=1, j ≤ m, j++) Sim₁ ← Run synonym matcher over a[i] and b[j] Sim₂ ← Run Levenshtein matcher over a[i] and b[j] Avg[i][j] ← Average of Sim₁ and Sim₂ Display all the pairs where Avg[i][j] ≥ α </pre>

Fig. 5 Performance result of SMA-COVID-19 approach



Experimental configuration and analysis: We have run SMA-COVID-19 in windows server 2012 R2 standard with an Intel Xeon, 2.20 GHz (40 cores) CPU, and 128 GB RAM. The proposed approach is implemented in the Python programming language. During experiments, we have set parameter $\alpha = 0.8$. We used libraries like Numpy (NumPy is a Python library, supporting large, multi-dimensional arrays and matrices, along with a large number of high-level mathematical functions for these arrays) and pandas (Open source, BSD-licensed library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language). We have imported the Python library for string matching and fetched a list of words from the nltk library (from `nltk.corpus import stopwords`).

Figure 5 shows the matching results of available COVID-19 ontologies, namely CODO, VIDO, CIDO, IDO, IDO-COVID-19 with respect to COVID-19-IBO ontology in terms of performance parameters, namely precision, recall, and F-measure. The precision parameter explains the correctness of the algorithm, whereas the Recall parameter measures the completeness of the algorithm. The parameter F-measure is the harmonic mean of precision and recall [12]. The obtained results of these parameters show that the proposed matching algorithm has reasonable performance.

5 Conclusion

We have extended the CIDO ontology by reusing the concepts from other data sources. The developed InBan_CIDO ontology offers accurate and precise knowledge about the impact of COVID-19 on the Indian economy as well as detailed information about the COVID-19 data. To detect the overlapping information, we have provided the SMA-COVID-19 algorithm approach that matches the schema of COVID-19 ontologies. The experimental analysis shows that the proposed approach has reasonable results in terms of precision, recall, and F-measure.

References

1. Web Ontology Language (OWL). https://en.wikipedia.org/wiki/Web_Ontology_Language
2. Y. He, H. Yu, E. Ong, Y. Wang, Y. Liu, A. Huffman, B. Smith et al., CIDO, A community-based ontology for coronavirus disease knowledge and data integration, sharing, and analysis. *Sci. Data* **7**(1), 1–5 (2020)
3. A.K. Mishra, A. Patel, S. Jain, in *Impact of Covid-19 Outbreak on Performance of Indian Banking Sector*. CEUR Workshop Proceedings, vol. 2786 (2021)
4. S. Babcock, L.G. Cowell, J. Beverley, B. Smith, *The Infectious Disease Ontology in the Age of COVID-19* (2020)
5. B. Dutta, M. DeBellis, *CODO: An Ontology for Collection and Analysis of Covid-19 Data* (2020). arXiv preprint arXiv: 2009.01210
6. D. Jones, T. Bench-Capon, P. Visser, *Methodologies for Ontology Development* (1998)
7. A. Patel, S. Jain, A partition-based framework for large scale ontology matching. *Recent Pat. Eng.* **14**(3), 488–501 (2020)
8. M.A. Musen, The protégé project: a look back and a look forward. *AI Matters* **1**(4), 4–12 (2015)
9. D.L. Gomes, T.H.B. Barros, in *The Bias in Ontologies: An Analysis of the FOAF Ontology*. Knowledge Organization at the Interface (Ergon-Verlag, 2020), pp. 236–244
10. OOPS! Tool. <http://oops.linkeddata.es/>
11. A. Patel, S. Jain, A novel approach to discover ontology alignment. *Recent Adv. Comput. Sci. Commun. (Formerly: Recent Patents on Computer Science)* **14**(1), 273–281 (2021)
12. P. Shvaiko, J. Euzenat, Ontology matching: state of the art and future challenges. *IEEE Trans. Knowl. Data Eng.* **25**(1), 158–176 (2011)

Prevention of Phishing Attacks Using QR Code Safe Authentication



M. Taraka Rama Mokshagna Teja and K. Praveen

Abstract Phishing is a type of attack in which attackers obtain personal information such as usernames, passwords, credit card information, and network credentials. They deceive victims by impersonating a reputable individual or entity and conducting specific acts, such as clicking on a harmful connection or attachment or intentionally revealing sensitive information over the phone or through email. In general, phishing sites attempt to deceive victims by pretending they are on a legitimate website to steal their account credentials and other sensitive information. In this paper, we implemented a safe authentication system using secret-key sharing and QR codes. This authentication system has a dedicated mobile application for authentication, which will eliminate the process of entering the website's credentials and as a result, it will provide robustness for phishing.

Keywords Phishing · Safe authentication · QR codes · Secret key sharing

1 Introduction

Phishing is one of the main potential threats in the current digital world. It is an activity where an attacker tries to trick the victim with malicious links to steal their sensitive information by masquerading as a trustworthy entity. In comparison to the previous year, there has been a 15% increase in phishing incidents in 2020. According to F5 Labs 2020 Report [1], access control layer failures account for 52% of all breaches in the United States of America, including credentials theft due to phishing and brute-force attempts. According to a report by the Australian Information Commissioner (OAIC), phishing attacks are the most common cyber incident, accounting for 36% of all recorded incidents. In those cases, 29% of passwords were stolen.

M. Taraka Rama Mokshagna Teja (✉) · K. Praveen
TIFAC-CORE in Cyber Security Amrita School of Engineering,
Amrita Vishwa Vidyapeetham, Coimbatore, India
e-mail: cb.en.p2cys19019@cb.students.amrita.edu

K. Praveen
e-mail: k_praveen@cb.amrita.edu

According to the APWG report, the number of unique phishing websites detected in the fourth quarter itself is 637,302 [2]. Attackers are becoming more creative; they are attempting to create more realistic Web URLs, and nearly 55% of phishing websites target well-known entities. The credentials obtained from phishing attack were attempted to be used by the attacker within four hours, while some of them happening in real time. Recently, malicious QR codes have been used in phishing attacks often redirected users to cloned websites. The attacker can steal the credential through it without the user's knowledge and has no idea whether the URL is modified or not [3]. Five major characteristics of phishing are:

1. The email makes unrealistic threats or demands.
2. There's a catch.
3. Poor spelling and grammar.
4. A mismatched or dodgy URL.
5. You are asked for sensitive information.

Authentication is the process of identifying whether someone or something is, who, or what intended to be. It allows organizations only to allow authenticated users to protect their assets. Assets may include computers, databases, websites, and networks [4]. For example, User A can have only access to his data and cannot see User B's data after authentication. Authentication is critical in preventing unauthorized access, which can result in disastrous data breaches. When user authentication is not safe, intruders can gain access to the system and steal confidential data. Companies like Adobe, Yahoo, and Equifax have all experienced data breaches of their failure to protect user authentication. Without a secure user authentication process, any organization could be in jeopardy. There are plenty of authenticating mechanisms; the most four common types of authentication mechanisms are [5]:

1. **Password Authentication:** In this method of authentication, the user is authenticating with credentials that typically contain a user ID and password. Passwords, on the other hand, are vulnerable to phishing attacks. Most of the users use the same set of credentials across multiple online accounts. Since they are much easier to recall, most of them use basic passwords. As a result, users prefer convenience over protection.
2. **Multi-factor Authentication:** Multi-Factor Authentication (MFA) allows the user to have multiple authentication layers like OTP, fingerprint, facial recognition, or code generated in the authenticator app. MFA has a good defense against phishing attacks. The users are not able to get an authentication code if they lost their mobile or SIM card.]
3. **Certificate-based Authentication:** Certificate-based authentication uses digital certificates to identify the identity of the users or computers. A digital certificate associates a user's digital identity with a public key and a certification authority's digital signature. Users must have their digital certificates to sign in to a server. The digital signature and certificate authority are checked for validity by the server.

- 4. Token-based Authentication:** This method of authentication uses a specific encrypted string of random characters is returned if the user credentials are valid. To access resources, the user may use the token instead of re-entering their credentials. RESTful APIs used by different frameworks and clients are examples of token-based authentication use cases..

OWASP suggests following NIST password length and complexity guidelines when developing your application's login page. They also recommend that all of the most commonly used passwords be instantly dismissed. Make use of an identity and access management (IAM) framework that makes it simple to build and execute a password strategy. A reliable password storage policy is essential for preventing data breaches that jeopardize an organization's credibility. The basis of safe password storage is hashing. By looking at the hashed password, an attacker cannot infer the original password.

The hash value would be the same if two users chose the same password. To ensure that the hashing process's performance is unique, you can add random data to the password called password salting. When a user tries to log in, the Web application passes the password through a hashing function and compares the result to the database value. The login is successful if the password hashes match. Hashing with salts can shield you from a variety of attack vectors, including rainbow table attacks and dictionary and brute-force attacks, while also slowing down dictionary and brute-force attacks.

Organizations need to think a step ahead of single-factor authentication techniques. There is a necessity to think about authentication as a way to enhance the experience of users. Multi-factor and other authentication mechanisms remove the need to recall long and complicated passwords.

This paper is organized as follows: Sect. 1 provides an overview of phishing and different authentication mechanisms. Section 2 explains various related works to prevent phishing attacks. Section 3 describes the proposed architecture. Section 4 provides the security features of the Django framework. Section [5] explains the implementation and analysis of proposed architecture. Section [6] briefs on the conclusion and future scope of the paper

2 Related Works

Phishing Phil [6] help the user to educate about the phishing site. The motto of this game is to provide internet users with conceptual information about phishing attacks. In this game, Phil is the small fish to examine the URL next to the worm. Phil is about to eat and determine whether its associated URL is legitimate or a phishing site. Phil's father offers some advice. After spending 15 min on the game, the user gets to know how to identify phishing websites. Users who played this game were more likely to recognize phishing websites than those who did not. Similarly, there is a scope for a comprehensive phishing study using deep learning methods like [7].

A Phish indicator [8] is a browser extension that will detect and classify the URL as phishing or not with the help of the Levenshtein algorithm. Whitelisting is always restrictive, which means whenever the user needs to browse the URL, not in the classification that is not on the whitelist. It shows it as a phishing site even though it is legitimate. Similarly, for the identification of malicious URLs, a visual alert system was carefully developed and implemented. Google Safe Browsing API and PhishTank API are two well-known security APIs that are being used. It allows the applications to search URLs against a database of suspected phishing and malware websites that is continuously updated. The resulting user gets to know about the URL is reliable or not [9]. Similarly, anomaly detection techniques like [10] are used to find phishing anomalies.

Phishing websites can take any data as input for authentication. In this model, the specially designed mobile application takes advantage of it. A fake login account with fake credentials is created by using a mobile app. Whenever the user opens the login page, it mimics the user login procedure and generates an alert. By monitoring hash code changes of the URL when the page is loading, UnPhishMe [11] helps to determine whether the current login page redirects to another web page after an authentication attempt. It listens to the `HttpURLConnection` status code. It decides whether the website is phishing or not.

In one of the architectures [12], the hybrid approach to phishing detection decreases the false positive rate by evaluating the website's content. In each step, the legitimate website is filtered out before moving on to the next. Similarly, another model [13] uses a two-stage method to detect phishing websites, with the first stage focusing on the RDF model of the website. The second stage was based on a machine learning technique. In a real scenario, there is a possibility that genuine sites may be represented as phishing because of their unpopular domain name. One of the anti-phishing methodologies was the model [14] constructed by considering the features extracted from the 14 inherent characteristics of the suspicious phishing web page.

To avoid phishing and man in the middle attacks, most of the communication in the model [15] is in the form of QR codes. In the beginning, the user initiates the single sign-on (SSO) to the authentication server. Then the server processes the request and returns the QR code to the desktop browser. The customized QR code app is used to scan the encrypted QR code. After the scanning, the user has to enter the credential into the app. After that, the credentials are encrypted in the form QR code, and the resulting QR code is scanned on the desktop using the Web camera and then sent to the authentication server. If the user is authenticated, the user will get access.

In one of the models [16], after the user verifies his identity by using his credentials in the first stage, then the user's smartphone and a pseudo-randomly generated alphanumeric QR code are then used as the one-time password token sent to the user via email as the second factor of authentication. The most prominent safe authentication model is proposed by combining the concepts of QR code and OTP [17]. In this authenticating system, the user verifies his identity by using his credentials in the first stage. If the credentials are authenticated, then an encrypted string that consists of user details (registered IMEI number) is displayed in the form of a QR

code. The user needs to scan the QR using a QR code reader of the linked mobile IMEI number. If the details are valid, only then the OTP is generated. The second step of authentication is done using that generated OTP.

The technology for authentication is evolving [18] and, the models mentioned above have limited their scope to 1–2 attack vectors. The most prominent authentication models like multi-factor authentication [16, 17] will protect the users to an extent, but credential theft through compromised phishing websites or MiTM persists [19, 20]. However, compromising login credentials lead to highly adverse consequences. So to address this problem, we designed and developed the authentication model to eliminate the need for a user password in the Web application by compensating with an effective approach to the user with a custom mobile application for authentication.

The overall aim of this paper is to design and develop an effective QR code safe authentication system that is user-friendly and has less response time. In detail, our contribution has three objectives mentioned below.

1. Design and Develop a safe QR code authentication system to mitigate the credentials theft due to Phishing and Man in the Middle attacks
2. Mitigating the common web vulnerabilities like CSRF, XSS attacks
3. A user-friendly authentication model having minimal response time to authenticate.

3 Proposed Model

In proposed QR code safe authentication model, the user need not enter the password in the Web browser at any instance; there is a dedicated mobile app for authentication. Figure [1] depicts the proposed model's architecture, and it is developed using the Django framework in section [4].

The user needs to install the mobile application and register with the service, and then the unique authentication token is created and linked with that corresponding username. This authentication token is used in later stages as explained in Fig. [1].

The login authentication of the proposed model will be done in six steps as follows:

- Step 1: When the user enters the username, the session started with the server with the corresponding user, sends the request to the backend server. A unique ID is generated and forwards the request to the authentication server.
- Step 2: The authenticating server generates the secret-key shares using Shamir's secret-key sharing algorithm and Sends one of the mandatory secret-key shares with a hashed unique ID to the backend server. The backend server will generate the QR code with the first key secret and hashed unique id, the generated QR displays on the Web browser to the user.
- Step 3: The user has to log in to the customized Android or iOS application already registered with the service. When the user successfully logged in to that app by

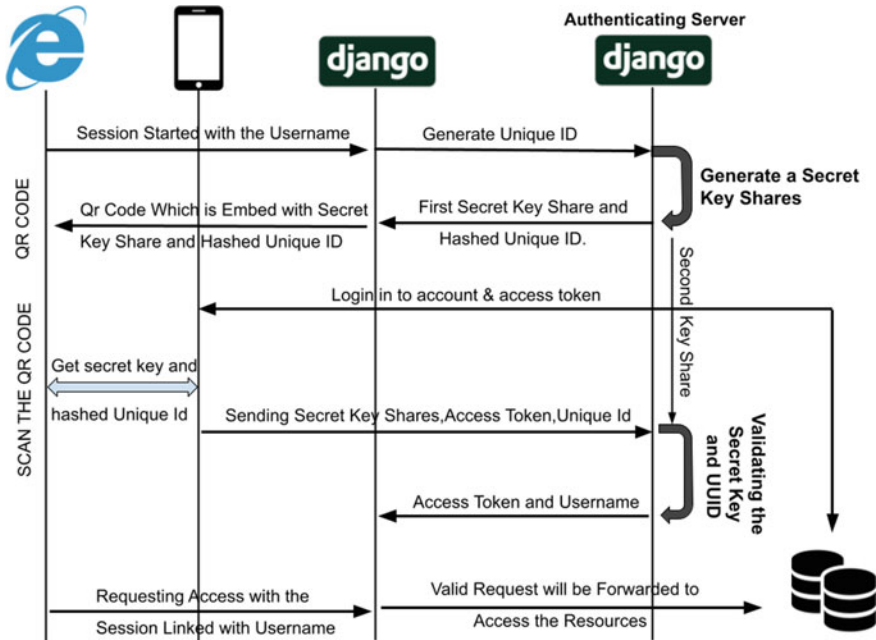


Fig. 1 Architecture of proposed model

default, the mobile application will access the authentication key created during the account creation time.

- Step 4: The mobile application will decode the QR code, which contains a secret key and unique ID (hashed) and will be sent along with the access token of the user account to the authentication server linked to a specific user account.
- Step 5: The authentication server validates the secret-key share and the unique ID of the requested user. In case of a valid request, it sends the access token linked to that user account to the corresponding backend server.
- Step 6: The backend server validates the session is linked with the user and gives access to the valid user to the resources with the access token.

In the designed authentication model, the authentication server is isolated from the backend server to follow the defense-in-depth strategy and also user has to scan the QR code before timeout, which is 20s. If the user failed, so the user has to initiate the login request again and the user has to follow the same login procedure explained above.

4 Security Feature in Django Framework

Django is a high-level open-source Python Web framework that allows stable and sustainable websites and provides a hassle-free environment for Web development. Django architecture [16] is logically structured, and it mainly consists of three important components [21]:

1. **Model:** It is used to handle the database. It is a data access layer that takes care of the data.
2. **Template:** The template is a presentation layer that takes care of the whole user interface.
3. **Views:** The view is used to hold data and render a template while also executing business logic and interacting with a model.

Django is a secure framework [22, 23], it ensures the developers to not make common mistakes that were once left open for the backend developer to complete. Django's user authentication system makes it easy to handle user accounts and passwords. This framework provides the mitigation of the most common Web vulnerabilities, as mentioned in Table 1.

By default, Django will protect the Web application from SQL injection and XSS with parameterized query and input validation. To make our Web application secure against the CSRF attack, the developer needs to use the CSRF token at the end of every HTML script. To mitigate vulnerabilities like clickjacking, session timeout, and host validation, the developer needs to add the below middlewares:

1. `django.middleware.security.SecurityMiddleware`
2. `django.contrib.sessions.middleware.SessionMiddleware`
3. `django.middleware.csrf.CsrfViewMiddleware`
4. `django.middleware.common.CommonMiddleware`
5. `django.contrib.messages.middleware.MessageMiddleware`
6. `django.contrib.auth.middleware.AuthenticationMiddleware`
7. `django.middleware.clickjacking.XFrameOptionsMiddleware`

Table 1 Mitigation of common web application threats using django framework

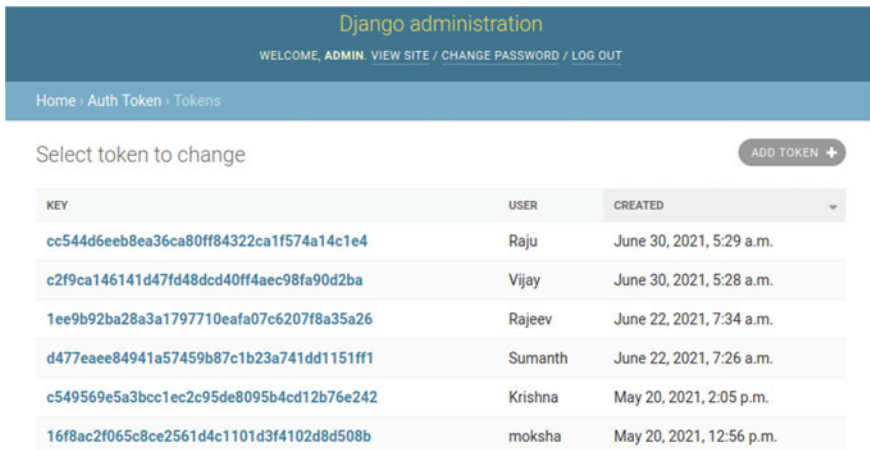
S. No.	Threat	Mitigation in Django
1	SQL injection	Parameterized query
2	Cross site request forgery	Csrf token
3	Cross site scripting (XSS) protection	Input validation
4	Clickjacking	X-Frame-Options
5	Cache poisoning	Host validation

5 Implementation and Analysis

When the user registers with the service, it creates an auth token respective to the username, as shown in Fig. 2. That create auth token is used for authentication in the later stage.

If the user wants to log in to the todo planner website first, they need to login to their account on mobile, and then they have to enter the username on the website as shown in . The session started with the server with the corresponding user with a unique id. The unique ID is an alphanumeric of length 32 characters is generated in the backend server. To avoid the adversaries finding the randomness of the unique id. The unique ID is hashed using the SHA-256 algorithm. It is nearly impossible to find the randomness of the unique ID with an advanced frequency test, and the unique ID is different whenever the user initiates the request. After the hashed unique id, the backend server forwards the request to the authentication server. The reason why we separated the authentication server from the backend server is to follow the defense-in-depth strategy. In simple terms, if the backend server is taken down or compromised, it does not affect the authentication server as it is isolated.

The authenticating server generates the key shares using the Shamir secret-key sharing algorithm. In this case, the secret key is divided into three key shares. The threshold is set to 2, which means it required two key shares to reconstruct the secret. The first key share and hashed unique ID is embedded in the QR code and sends to the backend server. The second secret-key share is within the authentication server, and the third secret-key share is kept and accessed by the admin for auditing purposes. The user may find it difficult to enter the extended code in the authentication process. We embedded the secret-key share and unique ID in the QR code, and the user needs to scan that QR using the mobile to get access.



Django administration

WELCOME, ADMIN VIEW SITE / CHANGE PASSWORD / LOG OUT

Home · Auth Token · Tokens

Select token to change ADD TOKEN +

KEY	USER	CREATED
cc544d6eeb8ea36ca80ff84322ca1f574a14c1e4	Raju	June 30, 2021, 5:29 a.m.
c2f9ca146141d47fd48dcd40ff4aec98fa90d2ba	Vijay	June 30, 2021, 5:28 a.m.
1ee9b92ba28a3a1797710eafa07c6207f8a35a26	Rajeev	June 22, 2021, 7:34 a.m.
d477eae84941a57459b87c1b23a741dd1151ff1	Sumanth	June 22, 2021, 7:26 a.m.
c549569e5a3bcc1ec2c95de8095b4cd12b76e242	Krishna	May 20, 2021, 2:05 p.m.
16f8ac2f065c8ce2561d4c1101d3f4102d8d508b	moksha	May 20, 2021, 12:56 p.m.

Fig. 2 User accounts linked with a random authentication tokens

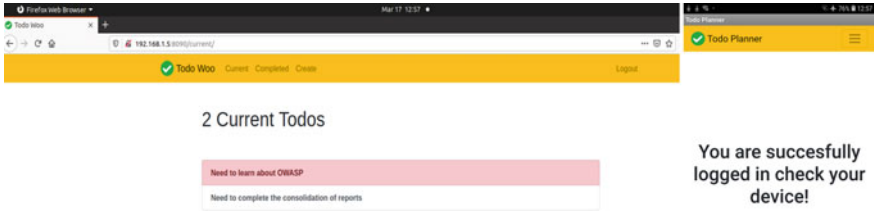


Fig. 3 Snippets of successful authentication

When the user scans the displayed QR code, the mobile app sends the code with the access token of that account to the authentication server. The authentication server validates the key share using Lagrange’s polynomial. The concept is to form Lagrange’s identities first, and the summation of Lagrange identities gives the required functionality, which is needed to construct the secret. Equations (1) and (2) and are used to reconstruct the secret from the key shares where k is the threshold.

$$f(x) = \sum_{i=0}^{K-1} y_i l_i(x) \tag{1}$$

$$l_i = \frac{x - x_0}{x_j - x_0} \times \dots \times \frac{x - x_{j-1}}{x_j - x_{j-1}} \times \frac{x - x_{j+1}}{x_j - x_{j+1}} \times \dots \times \frac{x - x_{k-1}}{x_i - x_{k-1}} \tag{2}$$

If the request is valid, it sends the access token, as shown in Fig. 1. The backend server validates the session is linked with the user and gives access to the user to access the resources with the access token, as shown in Fig. 3. The access token is a key to the specific account that will create at the time of creation, as shown in Fig. 2. Similarly, when user-1 tries to access another user’s account through his app, the user did not get access because user-1 does not have the authentication token of user-2. It results in failed authentication and shows a message on the mobile application as shown in Fig. 4. In the whole authentication process, the model eliminated the need to enter the credentials on the website. Resulting, it will protect the users/employees of the organization robust against phishing sites.

QR code safe authentication system eliminates the need for a user password in the Web application by compensating with an effective approach to the user with a custom mobile application for authentication and far better than other authentication techniques. This model does not take more than 10–15 seconds for the user authentication and far better than MFA and nearly equals the time taken for SFA as mentioned in Table 2.

The Security features in the QR code safe authentication:

1. Hashing Algorithm SHA-256 is used for hashing Unique User IDs. First, with the avalanche effect of SHA-256, it is almost impossible to reconstruct the initial data from the hash value. Secondly, a collision (two messages with the same hash value) is infinitesimally small.

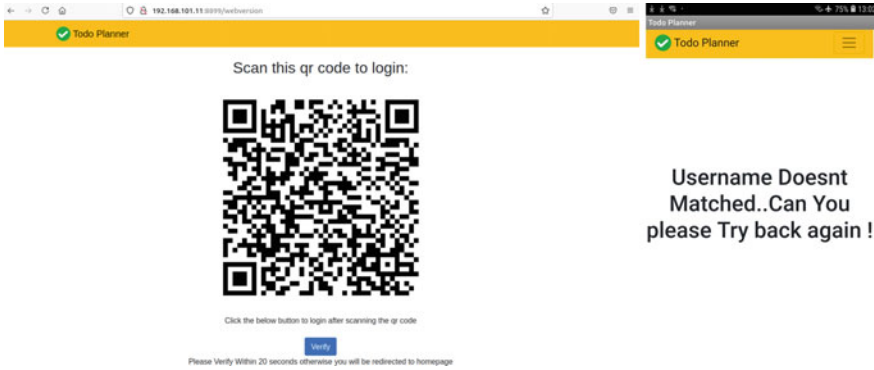


Fig. 4 Snippets of failed authentication

Table 2 Comparison of QR code safe authentication with existing models

Authentication method	Single factor authentication	Multi factor authentication	QR code safe authentication
Usability	Easy	Moderate	Easy
Time taken for authentication	10–20s (less)	25–60s (more)	10–15s (very less)
Credentials theft through phishing websites	Easy	Moderate	Highly difficult
MitM (Man in the Middle) attack	Possible	Possible	Not-possible

2. Secret Key sharing is used to avoid MitM attacks. The intruder does not know the second key share as it lies inside authentication, which is used for generating the key.
3. Sql injection, XSS, CSRF, Clickjacking are mitigated by using the security features in Django as mentioned in Table 1.
4. Authentication server is separated from the backend server, which results in a defense-in-depth strategy.
5. In the authentication, we eliminate the need for a user password in the Web application by compensating with an effective approach to the user with a custom mobile application for authentication.

6 Conclusion and Future Work

In this paper, we have showcased a safe login system in which the user is given a different unique ID each time they log in. To make the login mechanism more resistant to attacks, we employed concepts like the SHA-256 algorithm, secret-key

sharing, and the defense-in-depth technique. With the Django framework, we could mitigate the most common vulnerabilities such as SQL injection, XSS, and CSRF. The user need not enter the password in the Web browser because there is a dedicated mobile app for authentication. As a result, there is no risk of credential theft through a fake website. This work could be extended in the future by including a Single Sign-On (SSO), as well as integrating HTTPS and SSL in real-time for hardening security practices on mobile apps.

References

1. F5 Labs: 2020 Phishing and Fraud Report (Detailed Investigation Report) (2020). <https://www.f5.com/labs/articles/threat-intelligence/2020-phishing-and-fraud-report>
2. APWG: Phishing Activity Trends Report (Detailed Investigation Report) (2020 Q4). <https://apwg.org/trendsreports/>
3. P. Kieseberg, et al., QR code security, in *Proceedings of the 8th International Conference on Advances in Mobile Computing and Multimedia* (2010)
4. Authentication. <https://en.wikipedia.org/wiki/Authentication>
5. Authentication Methods That Can Prevent the Next Breach. <https://www.idrnd.ai/5-authentication-methods-that-can-prevent-the-next-breach>
6. S. Sheng, et al., Anti-phishing Phil: the design and evaluation of a game that teaches people not to fall for phish, in *Proceedings of the 3rd Symposium on Usable Privacy and Security* (2007)
7. S. Smys, J.S. Raj, Analysis of deep learning techniques for early detection of depression on social media network-A comparative study. *J. Trends Comput. Sci. Smart Technol. (TCSST)* **3**(01), 24–39 (2021)
8. S. Aparna, K. Muniyasamy, Phish indicator: an indication for phishing sites, in *Artificial Intelligence and Evolutionary Algorithms in Engineering Systems* (Springer, New Delhi, 2015), pp. 481–487
9. H. Yao, D. Shin, Towards preventing qr code based attacks on android phone using security warnings, in *Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security* (2013)
10. G. Ranganathan, Real time anomaly detection techniques using PYSARK frame work. *J. Artif. Intell.* **2**(01), 20–30 (2020)
11. S.A. Robila, J.W. Ragucci, Don't be a phish: steps in user education. *ACM SIGCSE Bullet.* **38**(3), 237–241 (2006)
12. A.A. Athulya, K. Praveen, Towards the detection of phishing attacks, in *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)* (IEEE, 2020), p. 48184
13. V. Muppavarapu, A. Rajendran, S.K. Vasudevan, Phishing detection using RDF and random forests. *Int. Arab J. Inf. Technol.* **15**(5), 817–824 (2018)
14. G. Ramesh, K. Selvakumar, A. Venugopal, Intelligent explanation generation system for phishing webpages by employing an inference system. *Behaviour Inf. Technol.* **36**(12), 1244–1260 (2017)
15. K. Choi, et al., A mobile based anti-phishing authentication scheme using QR code, in *International Conference on Mobile IT Convergence* (IEEE, 2011)
16. M. Eminagaoglu, et al., A two-factor authentication system with QR codes for web and mobile applications, in *2014 Fifth International Conference on Emerging Security Technologies* (IEEE, 2014)
17. B. Rodrigues, A. Chaudhari, S. More, Two factor verification n using QR-code: a unique authentication system for Android smartphone users, in *2016 2nd International Conference on Contemporary Computing and Informatics (IC3I)* (IEEE, 2016)

18. M.H. Barkadehi, et al., Authentication systems: a literature review and classification. *Telemat. Inform.* **35**(5), 1491–1511
19. K. Abhishek, et al., A comprehensive study on multifactor authentication schemes, in *Advances in Computing and Information Technology* (Springer, Berlin, Heidelberg, 2013), pp. 561–568
20. A. Bruun, K. Jensen, D. Kristensen, Usability of single-and multi-factor authentication methods on tabletops: a comparative study, in *International Conference on Human-Centered Software Engineering* (Springer, Berlin, Heidelberg, 2014)
21. Django Project MVT Structure. <https://www.geeksforgeeks.org/django-project-mvt-structure/>
22. Security in Django. <https://docs.djangoproject.com/en/3.1/topics/security/>
23. Django web application security. https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/web_application_security

Machine Learning Approach to Recognize and Classify Indian Sign Language



Smriti Pillai, Adithya Anand, M. Sai Jishnu, Siddarth Ganesh, and S. Thara

Abstract In these present circumstances, the future for differently abled students is a big question mark. As the education is turning entirely toward online in which the differently abled students are the most affected ones because their principal way of learning was physical, i.e., using gestures. In this present scenario of pan-epidemic siege, the value of time cannot be ignored for the students who are progressive citizens for a better future. During this intricate time, there is a need to sustain the pace of education for every child and the most important for the differently abled children who are always more enthusiastic in taking on the challenges of life. We at this time, pledge to do our best for the rightful e-learning. In the era of technology, providing education on digital platforms, our idea is to provide some assistance in the field of education technology. The idea is to train a model which will help us to identify and classify Indian Sign Language in the most reliable way. In the previously proposed solutions, the user is restricted to have a definite background so as their model could work accurately. In our system, that limitation is withdrawn. The user can be anywhere, and yet our model would perform the most desirable. We are using OpenCV for pre-processing and a machine learning model is used to recognize hand gestures. This model can then be employed in an Android application for greater perks.

Keywords Machine learning · Sign language · Classification algorithms · Gesture detection · Skin color detection · Feature extraction · OpenCV · Decision tree

S. Pillai (✉) · A. Anand · M. Sai Jishnu · S. Ganesh · S. Thara
Department of Computer Science and Engineering, Amrita School of Engineering, Amrita
Vishwa Vidyapeetham, Amritapuri, India
e-mail: siddarthganesh@am.students.amrita.edu

S. Thara
e-mail: thara@am.amrita.edu

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
S. Smys et al. (eds.), *Inventive Computation and Information Technologies*, Lecture Notes
in Networks and Systems 336, https://doi.org/10.1007/978-981-16-6723-7_28

373

1 Introduction

Sign language is a natural language mainly used in the deaf communities. It demands using hands for performing gestures to communicate. Indian Sign Language differs from American Sign Language in terms of the way the gestures are represented. Many alphabets vary from each other in contrast. Sign language can be mainly classified into contact-based and vision-based techniques. In a contact-based approach, the person can interact using external devices like sensors. It uses an instrumented glove that utilizes IoT devices to perceive the orientation, position, or angle data of the hands while displaying the sign [1]. Vision-based approaches use data collected from static images or video frames that were taken using a camera from the laptop or using an external Web camera. There are methods where the closed and opened fingers are identified using Douglas–Peucker algorithm by taking the boundary of the gesture into consideration [2]. Our project focuses on the vision-based approach. Extracting and identifying the sign language involves three phases, namely segmentation, feature extraction, and classification. The most important goal of segmentation is to eliminate the undesirable background and noises, giving only the region of interest (ROI). In the feature extraction phase, the distinctive features of the ROI will be deduced. The feature extracted will undergo classification where each gesture will be organized into its classes, and a machine learning model will be employed which can help to determine the gestures appropriately (Table 1).

2 Related Works

In one of the papers, P. Loke et al. proposed a system where they used hue, saturation, and intensity (HSV) model to segregate the hand from the image using the OpenCV library. They have used the Sobel edge detection method to get the boundary of the hand. After segmentation of the hand from all the images, they trained the dataset using supervised learning for training neural networks. The network is trained using the scaled conjugate gradient backpropagation algorithm to classify images. They captured the images using an Android application and sent those to an online Web server from which the input is fed to the neural network on MATLAB for recognizing the patterns in the hand gestures and corresponding text is being displayed [3].

In another work, Hanene Elleuch et al. proposed a method for static hand gesture recognition for mobile device monitoring in real-time. They used skin color algorithms and methods to detect just the hand region, eliminating the face region by using the Viola–Jones algorithm. They used various methods like convex hull detection, contour extraction, palm center detection, and convexity defects extraction, and these features were used on support vector machine (SVM) to recognize the gesture and deployed the model on an Android system. This experiment was carried out for five gestures as proposed in their paper [4].

Table 1 Comparison of existing works

Paper title	Dataset	Contact or vision-based	Feature extraction and algorithms used	Limitations
Indian sign language converter system using an Android App	Dataset collected using Android phone	Vision-based approach	HSV for hand tracking and segmentation, neural Network for pattern recognition	Models performance decreases when under different light conditions and background noise
A static hand gesture recognition system for real-time mobile device monitoring	Images collected through a webcam	Vision-based approach where five gestures are used to control mobile devices	Skin color extraction using HSV and face subtraction using Viola and Jones algorithm, SVM classifier to recognize the gestures.	Processing time is more
Sign language translator for mobile platforms	Static gestures of numbers and alphabets	Vision-based approach where we can also add a new gesture to the dataset	Skin detection using RGB, YCbCr, HSI and identifying gestures using descriptors	Lower recognition rate
Hand raising gesture detection in real classroom	Images of student's hand gestures	Different hand gestures	R-FCN architecture	Gesture size limitation
Arm gesture detection in a classroom environment	Images of hands in class	Robust vision-based method of identifying raising hands	Canny edge detection	Background color limitation

This work says about hand raise detection in classroom surroundings. When it comes to classrooms, the hand raise detection is challenging due to more complex situations and scenarios, different gestures, and pretty low resolutions. So they started building up a large hand-raising data set. The method used is R-FCN Architecture which has an automatic template detection algorithm and feature pyramid, whereas an automatic template detection algorithm is used to detect various hand rising gestures. Next for better detection and identification of small-sized hands, they use a feature called feature pyramid to capture simultaneously the detail and high-definite features [5].

In this paper, it tells about various techniques including identification of skin color, shape, and features are considered and addressed. In this process, it is presumed that the student's heads are randomly scattered around the class and lie roughly on the

same horizontal line, which is identified by a reference bar. By fixing the camera module at the front of the class, with a center part of the focus set parallel to this line, here we check the study of the parts of capturing image right above the student's heads in the classroom. Next, the raw edges are obtained by application of a Canny edge detector which is based on cognate of the gray-level intensity of the feed, which is obtained from R, G, B assembly. By presuming that the color of the background wall of a usual classroom is proportionately different from that of student's skin tones, using skin distribution data within the edges of the object of interests could be useful for arm identification and from the output of the skin map, the map is extricated which indicates the outer lines of the human skin [6].

In another work, Geetha M et al. proposed a sign language Android application where it captures the image using the phone camera and does the processing to identify the gesture. Initially, they did a contrast using a histogram matching to classify the gestures that are closer to the testing data, and later, those samples were subjected to oriented fast and rotated binary robust independent element features (BRIEF) thus bringing down the CPU time. Skin detection was done used RGB, HSI, and YCbCr algorithms. Their solution worked with an accuracy of 69.41%. They also had an additional feature where the application user itself can add different new gestures to the already existing data [7].

This work proposes the progress of a sign language recognition system that helps recognize South Indian languages. In the proposed method, 32 sequences of number signs are developed by using hand palm images from the right hand. An image taken at run time is examined to identify fingertip regions of the five fingers namely thumb finger, index finger, middle finger, ring finger, and little fingers as easier to use those edge images for obtaining the fingertip position for further processing. The whole process is comprised of four processes, i.e., data procurement, getting the palm images, detecting the sign, training, and conversion to text [8].

3 Proposed Methodology

3.1 Dataset Description

Dataset plays a major role when working with machine learning. Machine learning has been emerging as data around us keeps expanding, but there is a hindrance to the measure of labeled data. When it comes to sign language, there is a contrast within the alphabets in Indian and American Sign Languages. Even though there are existing datasets for American Sign Language but no fitting dataset for Indian Sign Language, we generated a dataset for Indian Sign Language (Fig. 1).



Fig. 1 Dataset with 7 different people with varying backgrounds

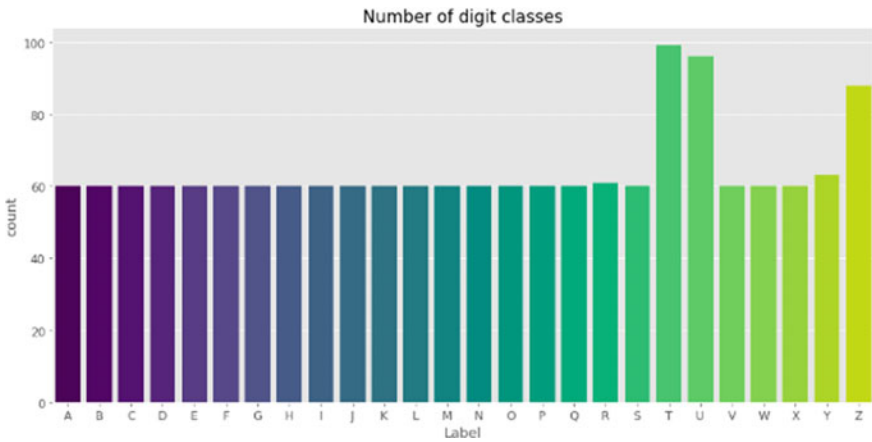


Fig. 2 Count of number of images for each class in the Train dataset

The dataset has roughly 3000 images of various people with distinctive backgrounds contributed by seven different people. The images were captured from an external webcam using a Python script. After running the Python script, it would ask the number of images to be captured and the folder name to collect the images. Every image would be captured with a pause of 3 s. This assisted to easily form a dataset for Indian Sign Language (Fig. 2).

3.2 *Skin Color Detection*

After obtaining the dataset, we mask the image to distinguish only the skin region. For masking the skin region, there are mainly three methods. The first one is the RGB (red, green, blue) color model. RGB has values that range from 0 to 255. For example, when all the values are set to the highest value, i.e., RGB (255,255,255) delivers the color white. Next is the YCbCr (luminance, chrominance) model where the YCbCr signals are developed from the corresponding RGB gamma-configured source. Lastly, there is an HSV (hue, saturation, value) model which is similar to an analog of RGB. HSV shows colors in an instinctive and precise manner than the standard RGB. Every model has its shortcomings, but when consolidated performs well according to our experiment. So we used an RGB-YCbCr-HSV model to get precise skin segmentation. Since skin colors can be distinctive, we mask the image through a range of lower and higher threshold values, thus achieving the final mask.

3.3 *Feature Extraction*

We run the masked images on a MediaPipe Hands model. MediaPipe Hands is a hand and finger tracking solution. It employs machine learning to infer 21 points of a hand from just a single frame [9]. We collect these 21 points in an array and this array helps us to identify and crop out the hand region by applying padding with the largest and smallest values of the stored array points. After receiving the hand region with the MediaPipe points, we mask out all other components except for the 21 points on the image and resize all the images. Next, we transform every image to a 2D NumPy array and then flatten the 2D image array into a 1D array and transform these values into a CSV file. Thus, we acquire a set of normalized values in a CSV file which can further be used to train the machine learning model.

Algorithm 1 Pre-processing Steps: Obtain CSV file to train ML Models

Require: Images in a folder

```

while img in images.items() do
    Apply SkinMasking(img) using RGB-YCbCr-HSV algorithm
    Apply MediaPipe(img) to obtain 21-points and stored in list
    Get min-max value from list
    Crop out ROI (Hands) using min-max values
    Resize ROI and convert to subplot
    Flatten 2D subplot array to 1D array
    Convert 1D array to CSV file

```

end while

3.4 Models Used for Training

We tested our dataset on three distinctive models, namely logistic regression, decision trees, and random forest.

3.4.1 Logistic Regression

Logistic regression is a model that uses statistics to determine the probability of an event with the aid of previous data. The sigmoid function is like an 'S'-shaped curve used to convert values to probabilities. It takes values between the range 0 and 1 and squishes to either 0 or 1 label. 0.5 is considered as a threshold value in the graph.

3.4.2 Decision Tree

The decision tree algorithm comes under the supervised algorithms. The idea of this algorithm is to create a model that predicts the value of a target variable. For this, the decision tree uses the tree illustration to solve the problem where the leaf node resembles a class label and internal nodes of the tree correspond to the attributes.

3.4.3 Random Forest

It is a supervised learning algorithm. It is a pack containing many decision trees having a diverse set of parameters trained on varying data. This can be used for both classification and regression tasks. It is one of the most applied algorithms because of its integrity and stability.

Our dataset was partitioned in such a way where 70% of the dataset was used for training and the rest 30% was used for testing purposes. After training the models, it was found that the decision trees performed the best among the other three models with an accuracy of 94.61%. Logistic regression showed an accuracy of 78.64% and random forest provided an accuracy of 85.63%. Thus, a decision tree was used to categorize the 26 alphabets into their respective classes (Fig. 3).

4 Experimental Results

The step-by-step experimental results that were achieved after pre-processing the image are shown in Fig. 4.

We have utilized three models for our project, namely logistic regression, random forest, and decision tree. After training the models, we got an accuracy of 78.64% for the logistic regression model, 85.63% for the random forest model, and 94.61% for

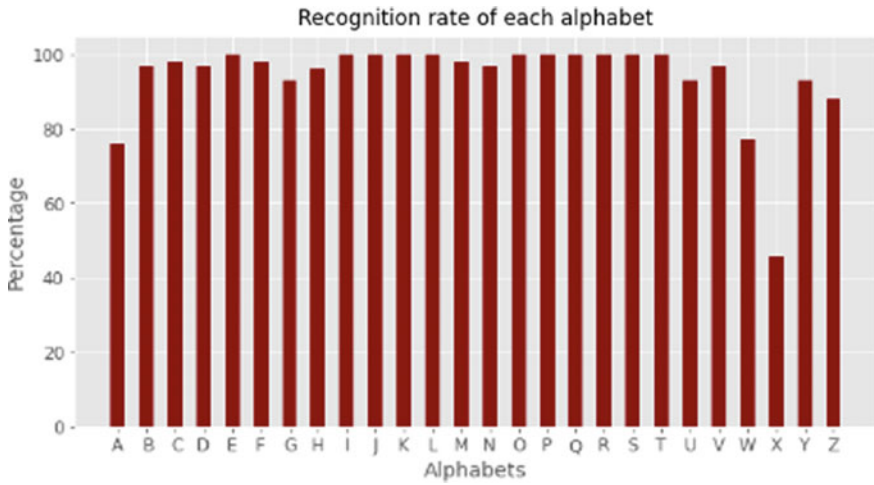


Fig. 3 Recognition rate for each alphabet

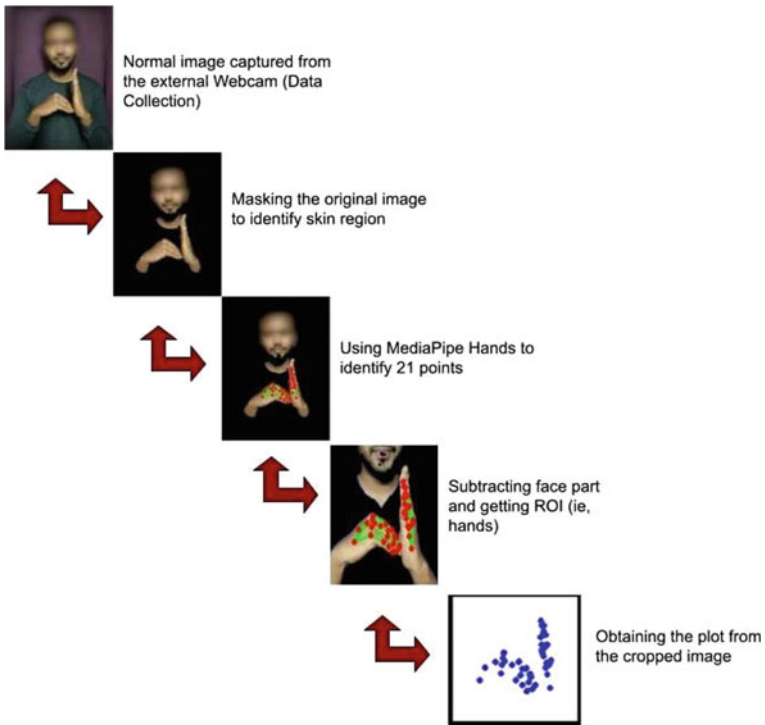
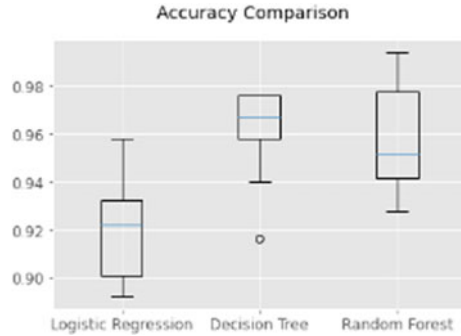


Fig. 4 Workflow of results achieved

Fig. 5 Accuracy comparison of each model



the decision tree model, in which decision tree was identified as the best model. The following results in Fig. 5. were obtained by following the steps mentioned in the methodology section. The k-fold cross validation technique is used to evaluate each ML algorithm and the box plot graph shows the spread of accuracy scores across every cross validation fold for every algorithm used.

Our model can work with great accuracy given varying light conditions or backgrounds. Only very few existing sign language papers use both hands for the gesture [10], our model is one of them.

The bar graph in Fig. 4. represents the recognition rate for each alphabet which is formed by taking the true/false positive and true/false negative from a confusion matrix. Confusion matrix basically compares the predicted and target values and computes the precision, recall, and $F1$ scores. From the bar graph, we could analyze that letters A, W, and X made the least accurate predictions and thus would require tuning the model to make relevant predictions. Every other alphabet could be predicted with more than 80% accuracy.

5 Conclusion and Future Work

There have been several types of research done in this field, mostly in the vision-based approach [11]. After looking through various methods, we have come up with a distinct approach to classify alphabets. In this paper, a novel and efficient approach was used to classify Indian Sign Language. The focus of this project is applying the RGB model, YCbCr model, and HSV collectively for feature extraction. Simultaneously, we have utilized a new technology, MediaPipe hands, which plays a vital role in feature extraction as well. The experimental results showed that our model could achieve 94% accuracy using the decision tree classifier. Other sign language-related papers limit users to a particular environment with fixed background and appropriate lighting conditions. But in our model, we surpass that limitation, which opens new lanes for future advancements. This model can be implemented on an Android device which would help to identify gestures in real-time thus making

it easily usable to people with speaking and hearing difficulties with exceptional performance in any environment. We can also generate datasets and train them for a selective situation, for example, previous works have been done for bank scenarios [12]. This can be extended and be made possible for hospitals, shops, etc., thus providing every person in the deaf community a normal life like everyone else.

References

1. R.K. Megalingam, C. Chacko, B.P. Kumar, A.G. Jacob, P. Gautham, Gesture controlled wheel chair using IR-LED TSOP pairs along with collision avoidance, in *2016 International Conference on Robotics and Automation for Humanitarian Applications (RAHA)*, Kollam, 2016, pp. 1–7. <https://doi.org/10.1109/RAHA.2016.7931872>
2. M. Geetha, R. Menon, S. Jayan, R. James, G.V.V. Janardhan, Gesture recognition for American sign language with polygon approximation, in *Proceedings—IEEE International Conference on Technology for Education, T4E 2011*, Chennai, Tamil Nadu, 2011, pp. 241–245
3. P. Loke, J. Paranjpe, S. Bhabal, K. Kanere, Indian sign language converter system using an android app. *Int. Conf. Electron. Commun. Aerosp. Technol. (ICECA)* **2017**, 436–439 (2017). <https://doi.org/10.1109/ICECA.2017.8212852>
4. H. Elleuch, A. Wali, A. Samet, A.M. Alimi, A static hand gesture recognition system for real time mobile device monitoring, in *2015 15th International Conference on Intelligent Systems Design and Applications (ISDA)*, 2015, pp. 195–200. <https://doi.org/10.1109/ISDA.2015.7489224>
5. J. Lin, F. Jiang, R. Shen, Hand-raising gesture detection in real classroom, in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Calgary, AB, 2018, pp. 6453–6457. <https://doi.org/10.1109/ICASSP.2018.8461733>
6. J. Yao, J.R. Cooperstock, Arm gesture detection in a classroom environment, in *Sixth IEEE Workshop on Applications of Computer Vision, (WACV 2002). Proceedings*. Orlando, FL, USA, 2002, pp. 153–157 (2002). <https://doi.org/10.1109/ACV.2002.1182174>
7. M. Mahesh, A. Jayaprakash, M. Geetha, Sign language translator for mobile platforms, in *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2017, pp. 1176–1181. <https://doi.org/10.1109/ICACCI.2017.8126001>
8. P.S. Rajam, G. Balakrishnan, Real time Indian sign language recognition system to aid deaf-dumb people, in *2011 IEEE 13th International Conference on Communication Technology*, 2011, pp. 737–742. <https://doi.org/10.1109/ICCT.2011.6157974>
9. Mediapipe documentation. <https://google.github.io/mediapipe/>
10. M. Geetha, U.C. Manjusha, A vision based recognition of indian sign language alphabets and numerals using B-spline approximation. *Int. J. Comput. Sci. Eng. (IJCSE)* **4**, 3 (2012)
11. N. Aloysius, M. Geetha, Understanding vision-based continuous sign language recognition. *Multimedia Tools Appl.* **79**(31), 22177–22209 (2020)
12. G. Jayadeep, N.V. Vishnupriya, V. Venugopal, S. Vishnu, M. Geetha, Mudra: convolutional neural network based Indian sign language translator for banks, in *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*, Madurai, India, India, 2020

Comparison of Concurrent Program Behavior Using Java Interactive Visualization Environment



M. Shobitha, R. Prakash Sidharth, P. K. Sreesruthi, P. Varun Raj, and Jayaraman Swaminathan

Abstract It's important for software practitioners to understand mutual exclusion in different systems because most of the problems in concurrent systems boil down to achieve mutual exclusion. Mutual exclusion for different types of concurrent scenarios-multithreaded, parallel, distributed can be achieved in different ways by the constructs provided by the programming language. For each of the mentioned types, the performance (or behavior) varies in different ways. The performance of mutual exclusion algorithms is measured by mainly six metrics. This paper shows the comparison between the performance of a chosen synchronization algorithm by analyzing the sequence diagrams obtained from Java Interactive Visualization Environment (JIVE), a dynamic analysis framework for Java program visualization. This paper also presents the results and observations obtained after comparing dining philosophers problem in the above mentioned scenarios. The results are based on the metrics - Message complexity, Synchronization delay and Response Time. The analysis is done on low load performance.

Keywords Distributed mutual exclusion · Visualization · Comparison · Parallel · Analysis · Multithreading

1 Introduction

With technology rising on a never before seen scale, concurrent systems have been playing an increasingly prominent role in software development. Three forms of concurrent systems have been widespread over the years, namely multithreaded, parallel and distributed. While the multithreaded and parallel programs pertain to cooperating processes/threads on a single system, distributed programs span multiple systems, possibly spread over a large geographical area. It is a complex field that consists of devices that communicate and organize tasks to act as a single-coherent system. It

M. Shobitha · R. P. Sidharth · P. K. Sreesruthi (✉) · P. Varun Raj · J. Swaminathan
Department of Computer Science Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India
e-mail: swaminathanj@am.amrita.edu

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
S. Smys et al. (eds.), *Inventive Computation and Information Technologies*, Lecture Notes
in Networks and Systems 336, https://doi.org/10.1007/978-981-16-6723-7_29

383

combines the power of multiple machines to improve the performance of a program with unmatched scalability. The only reason that distributed programming is yet to come to the forefront can be attributed to its increased complexity in deployment and maintenance.

Concurrent systems are characterized by some challenges which are entirely different and non-existent in the sequential systems. Although each of these systems throws its unique challenges, the fundamental theme underlying these challenges revolve around safety, liveness and fairness. Dealing with these challenges by techniques such as mutual exclusion and other forms of synchronization introduces second level challenges such as message complexity, response time and other overhead. However, despite the challenges and the overhead involved in concurrent systems, they continue to play a dominant part in software development, due to the performance and scalability benefits they provide.

In this paper, we have analyzed the performance of mutual exclusion in the case of three different concurrent programming paradigms, namely multithreaded, parallel and distributed programming. There are predominantly six metrics used for measuring the performance of mutual exclusion algorithms. These include message complexity, synchronisation delay, response time, system throughput, low and high load performance and best and worst case performance. Since our observations deal with low load performance, more emphasis has been given to message complexity, synchronization delay and response time.

The goal of this project is to directly contrast the performance of distributed programs against their parallel and multithreaded equivalent using sequence diagrams obtained from Java Interactive Visualization Environment (JIVE) and further compare them using the aforementioned three metrics. This analysis has been carried out using the dining philosophers problem in all three programming implementations. The further observations and results based on this analysis are specified in the upcoming sections with an array of scope for future work.

The rest of the paper is structured as follows. Section 2 discusses some of the closely related work. Section 3 describes the implementation of multithreaded, parallel and distributed versions of the dining philosophers problem. Section 4 provides the analysis of the different concurrent versions providing key insights into their implementations. Section 6 summarizes the work and provides directions for future work.

2 Related Work

In this section, some closely related works of previous papers and their contribution to the concurrent program analysis are summarized. Concurrency around the scenarios-multithreaded, parallel and distributed systems are discussed. This is then, followed by the significance of JIVE-an interactive visualization tool used for the analysis.

Dining Philosophers problem [1] is addressed to acknowledge the working of concurrent programs. Java Threads [2] is used for the implementation of multi-threaded programming. MapReduce and Fork/Join are popular java based frameworks for achieving parallelism. In the dining philosophers problem, there are only five philosophers. Hence, the Fork/Join framework is a better choice for parallel program implementation [3, 4]. RMI framework [5] is used to build remote communication between Java programs that help to build distributed applications. But, all these frameworks have a drawback that they can't access shared objects at the same time, which makes synchronization [6] a vital part in the implementation of the concurrent program.

Sequence diagrams are interaction diagrams that record the relationships between objects when the programs work collaboratively. Object-to-object interaction of a program using sequence diagrams makes it easy to understand the workflow and the complexity of the program [7–9]. Sharp and Rountev [10] explain sequence diagrams and their working principles. Various limitations of the UML Sequence diagrams are pointed out and a set of techniques to overcome this limitation is highlighted. UML diagrams are extremely large and clustered making them hard to interpret properly. So the earlier unreadable UML sequence diagram is expanded upon to interactively explore different aspects of the diagram, to focus on subsets of the expressed behavior. Thus, the earlier sequence diagram can be elaborated for better understandability and proper interpretation by the programmer.

The sequence diagram of Java programs can be obtained through the Eclipse plugin tool Java Interactive Visualization Environment [11]. JIVE visually portrays both the call history and the runtime state of a program. The call history can be seen in the sequence diagram, where each execution thread is depicted in a different color, simplifying the object interactions. JIVE also generates runtime models, verifies and validates those models against design time models [12]. Ajaz et al. [13] already presented the performance study of a parallel program using JIVE on multicore systems. Kishor et al. [12] emphasize a methodology to interpret the sequence diagram in the form of a finite state diagram. The key state variables are annotated by the user. This is later combined with the execution trace to obtain the sequence diagram. Since state diagrams are smaller in magnitude compared to sequence diagrams, they provide more insight into program behavior and detect subtle errors.

3 Program Implementation

To shed light and raise discussion on concurrency, the dining philosophers algorithm has been implemented in this paper [14]. The dining philosophers problem is a classical problem of synchronization. Five philosophers sit around a circular table. Each of them has two states - thinking and eating. Five forks are placed on the table. In order to eat, the philosopher must have two forks in hand. No two adjacent philosophers can eat simultaneously.

3.1 Multithreaded Programming

To start the thread, the start() method must be invoked. Java provides Thread class to implement multithreaded programming. Thread class provides various methods to create and perform operations on a thread. One of the methods to create a thread process in Java, using Threads, is by extending the Thread class and overriding its run() method.

```
public void run() {
    while(true) {
        thinking();
        //Philosopher gets hungry
        fork.take();
        eating();
        fork.release();
    }
}
```

Code 1: run() method of the Philosopher class

Since in the dining philosophers problem, no two adjacent philosophers can get hold of both the forks simultaneously, it's important to have synchronization in the availability of the forks. When several threads are trying to access a common resource, it is necessary to have control over who has access. And this is what synchronization does. In the multithreaded program, two synchronized methods, namely take() and release(), are used. The synchronized methods lock an object for any shared resource; in this scenario - the forks. The message complexity, the number of messages required for the execution of the critical section, is $2(N-1)$, where N represents the number of philosophers.

```
synchronized void release() {
    Philosopher philosopher = (Philosopher)
        Thread.currentThread();
    int number = Philosopher.Number;
    fork[number] = false;
    fork[(number+1)%5] = false;
    notifyAll();
}
```

Code 2: synchronized method - release()

```
synchronized void take() {
    Philosopher philosopher = (Philosopher)
        Thread.currentThread();
    int number = Philosopher.Number;
    while(fork[number] || fork[(number+1)%5]) {
        try {
            wait();
        }
        catch(InterruptedException e){}
    }
    fork[number] = true;
}
```

```

        fork[(number+1)%5] = true;
    }

```

Code 3: synchronized method - take()

The program begins by initializing the threads of five philosophers. The threads start executing the run() method, where the philosopher has to go to the thinking state for a random period and then go to the eating state. The mutual exclusion part will be handled by the two synchronized methods, while taking and releasing the forks.

3.2 Parallel Programming

Fork/Join framework is a framework in Java that sets up and executes parallel programs by taking advantage of multiple processors, which is accomplished by identifying the availability of processor cores and allocating the tasks accordingly. It uses a divide-and-conquer strategy: divide a very large problem into smaller parts. These smaller parts can be further divided into even smaller ones, recursively until a part can be solved directly. In the parallel program, ForkJoinTask is used. Through this mechanism, a small number of actual threads in ForkJoinPool controls a large number of tasks to be executed. ForkJoinTask.invokeAll() method combines fork() and join() in a single call and starts the instances of all the philosophers.

```

for (int i = 0; i < philosophers.length; i++) {
    Object leftFork = forks[i];
    Object rightFork = forks[(i + 1) % forks.length];
    philosophers[i] = new Philosopher(leftFork,
        rightFork);
    subtasks.add(philosophers[i]);
}
ForkJoinTask.invokeAll(subtasks);

```

Code 4: using invokeAll() method

To attain mutual exclusion, the concept of the synchronized block is used. For any shared resource, the synchronized block is used to lock an object. Nested synchronized blocks help to get hold of both the forks the philosopher needs. So, the message complexity becomes $4(N-1)$. The instances of all the five philosophers are invoked by ForkJoinTask.invokeAll() method. The instances then run the compute() method, where the synchronized block is defined. Thus, mutual exclusion and concurrency are achieved.

```

synchronized (leftFork) {
    synchronized (rightFork) {
        //eating
        gotForks(leftFork, rightFork);
        try {
            TimeUnit.MILLISECONDS.sleep((int)(Math.random()*50));
        } catch (InterruptedException e) {}
    }
}

```

Code 5: synchronized block

3.3 Distributed Programming

Remote communication is an inevitable part of distributed programming. In Java, Remote Method Invocation (RMI) helps to achieve this communication between the systems. RMI allows an object running in one Java virtual machine to invoke methods on an object running in another Java virtual machine. In an RMI application, there will be a client program and a server program. The server program will generate a remote object and a reference of that remote object is made available for the client (Code 6). The remote objects on the server can be accessed by client requests, and thus, the client can invoke the server's methods (Code 7).

```
public static void main(String[] args) {
    try {
        String name = "ForkServer";
        ChopstickInterface server = new
            ChopstickServer();
        Registry registry =
            LocateRegistry.createRegistry(8000);
        registry.rebind(name, server);
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

Code 6: main() method of the server program

Unlike multithreaded and parallel programs, synchronized keyword can't be used in distributed systems because they can only be used to control access over a shared object within the same JVM. Hence, five semaphores have been used to keep track of the availability of the five forks. A semaphore uses a counter to keep track of the status of a shared resource and controls its access. The message complexity is $2(N-1)$ here.

```
public static void main(String[] args) {
    try {
        String name = "ForkServer";
        Registry registry =
            LocateRegistry.getRegistry();
        frk= (ChopstickInterface)
            Naming.lookup("//localhost:8000/"+name);
        for (int i = 0; i <= 4; i++)
            new Philosophers(i, registry);
    }
    catch(Exception e) {
        System.err.println(e);
    }
}
```

Code 7: main() method of the client program

To execute the program, the server program has to run and then the client program. By running the server, it creates an object. Then, using reBind() method, it registers this object with the RMRegistry. To make use of the server methods, the client needs a reference of the object that the server created. With the help of the lookup() method,

the client fetches the object from the registry using its bind name. Remote communication is thus established. The server program includes `getForks()` and `returnForks()`, methods to handle the forks, which were included in the remote interface. In the client program, five philosophers are initialized using Java Threads and they run simultaneously.

```
import java.rmi.Remote;
import java.rmi.RemoteException;

public interface ChopstickInterface extends Remote{
    int getForks(int philNum) throws
        RemoteException;
    int returnForks(int philNum) throws
        RemoteException;
}
```

Code 8: Remote interface of the program

4 Analysing Program Behavior

4.1 Multithreaded Program Sequence Diagram

There are five philosophers present in Fig. 1 namely Philosopher 1, Philosopher 2 and so on till Philosopher 5. As in Fig. 1, we can see that Philosopher 5 goes to the thinking state for a while and then starts eating. After a certain amount of time Philosopher 5 releases the fork which allows its adjacent Philosopher 1 to gain access on one of the forks. The time gap between one process leaving the critical section and the next process accessing it is known as the Synchronization delay. In this case, the time interval between one philosopher exiting the eating state(releasing left and right forks) and the next philosopher entering the eating state(gaining control over left and right forks) is the synchronization delay. In Fig. 1, there is a time gap between `take1` and `eating1`. This time gap between sending the request and then getting control over both the forks is known as the Response time. When comparing Synchronization delay and Response time, the delay is relatively lower and response time is very high.

4.2 Parallel Program Sequence Diagram

Figure 2 shows the sequence diagram on parallel implementation of the Dining philosophers diagram. Philosopher 1, initially being in the thinking state, goes to the hungry state (requests for forks) after a while. This time gap between sending a request for forks(`hungry 1`) and getting access to it(`getForks1`) is known as Response Time. Philosopher 1 then puts back the fork, i.e., `returnForks()`, making it available

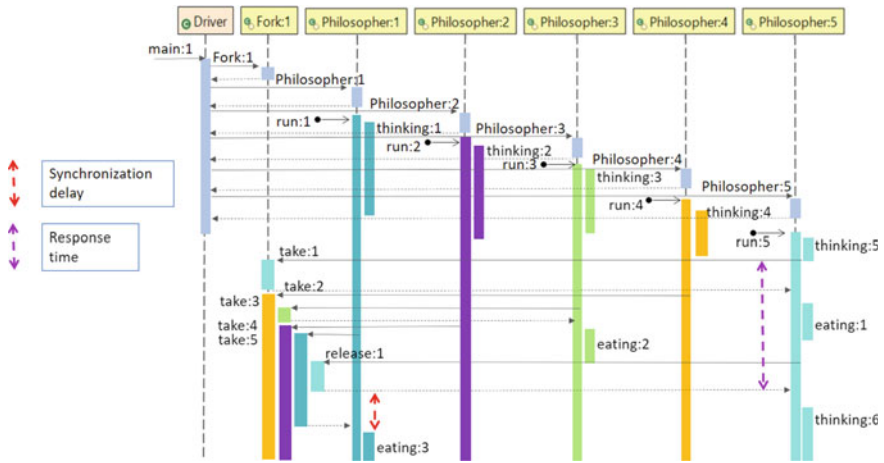


Fig. 1 Sequence diagram of multithreaded implementation

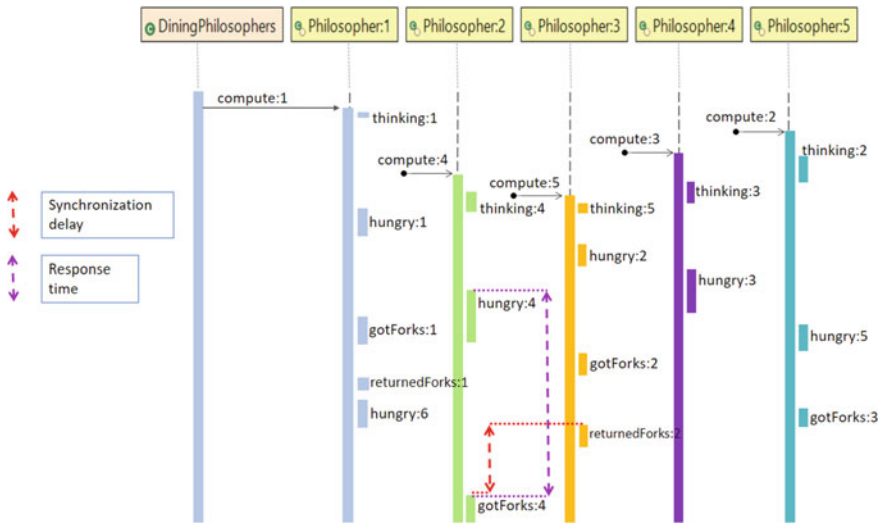


Fig. 2 Sequence diagram of parallel implementation

for adjacent philosophers. The time gap between returnedForks1 and GotForks4 is the Synchronization Delay. When looking at the variations of Synchronization delay, in most of the cases it is low. There are a few exceptions where the delay is drastically high, but on an average it is low. When comparing the Response time, for most of the cases it is relatively high. In this observation, there has been no low response time.

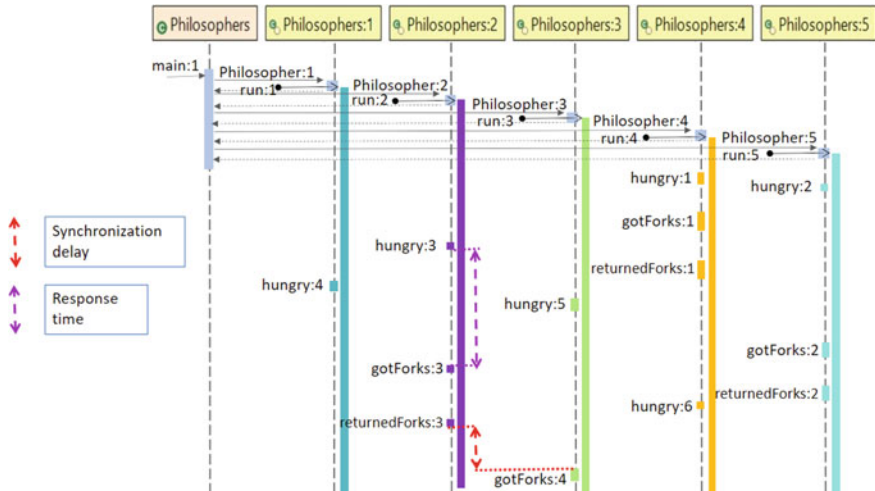


Fig. 3 Sequence diagram of distributed implementation

Table 1 Comparison based on three metrics

Metrics	Multithreaded	Parallel	Distributed
Message complexity	$2(N-1)$	$4(N-1)$	$2(N-1)$
Synchronization delay	Low	Low	very low
Response time	High	High	Low

4.3 Distributed Program Sequence Diagram

For distributed programs, the communication processes are distributed across different hosts. So, different sequence diagrams will be obtained for the server program and client program. But, here our main motive is to check upon the states of the philosopher, only the client program is taken into consideration (Fig. 3). Initially, all the philosophers will be in the thinking state and after some random amount of time they will move to the hungry state. Mutual exclusion takes place at this point. Philosophers check whether the forks are available for them. If both the forks are free, it's denoted by gotForks. By checking the diagram, it can be observed that the time gap between hungry and gotForks is most of the time high. So, it can be concluded that the response time of distributed programs is high. Comparing two adjacent philosophers, the time gap between returnedForks of one philosopher and gotForks of the other philosopher is very low, making the synchronization delay of the distributed program too low (Table 1).

5 Results

The following are observed from the sequence diagrams obtained from the concurrent programs:

- In the multithreaded program, the synchronization delay is low and the response time is high.
- In the parallel program, the synchronization delay is mostly low, but in some cases it is drastically high. Also, the response time is high.
- In the distributed programs, the synchronization delay and response time are low.
- When comparing synchronization delay of the three executions, Multithreaded execution and parallel execution take relatively lower time, while distributed execution takes the lowest time.
- When comparing the response time of the three executions, multithreaded execution and parallel execution took higher time while distributed execution took relatively lower time.

6 Conclusion

When the concept of mutual exclusion is taken into consideration, distributed programming rarely springs to mind owing to its increasingly complex nature. So we have directly compared the efficiency of distributed programming to its more frequently used counterparts such as parallel and multithreaded programming. The main observations were based on the comparison of performance metrics of Dining philosophers problem, on a small load, run in three different ways-multithreaded, parallel and distributed systems.

Java Threads are used for multithreaded programming and synchronization is achieved by using the synchronized methods. The message complexity of the multithreaded program is $2(N-1)$, N is the count of the threads/philosophers. To attain parallelism, Fork/Join framework is used and the synchronized blocks help in synchronization. The message complexity of the parallel program is $4(N-1)$, N is the count of philosophers. And, distributed programs are built using RMI framework and semaphores are used to achieve synchronization. Its message complexity is $2(N-1)$, N is the number of philosophers [14].

The performance analysis of the concurrent programs is done using the metrics message complexity, synchronization delay and response time. The message complexity is the number of messages required per execution of a critical section. It was found that in terms of message complexity the program run in parallel was the most complex while the programs run in distributed and multithreaded manner shared the same complexity. Synchronization delay is the time required for a process to enter the critical section after another process exits the critical section. It was also observed that Synchronization delay for parallel and multithreading is almost similar

but Synchronization delay for distributed systems is less compared to the other two. Response time is the time interval a request waits for its critical section execution to be over after its request messages have been sent out. When comparing the response time of parallel and multithreaded programming, they have almost same response time but, while comparing response time of distributed and multithreading or parallel, response time of multithreading is more than response time of distributed.

In conclusion, for a mutual exclusion program of low load, if RT is the response time and SD is the synchronization delay then,

$$RT (\text{Multithreading}) \sim RT (\text{Parallel}) > RT (\text{Distributed}) \quad (1)$$

$$SD (\text{Multithreading}) \sim SD (\text{Parallel}) > SD (\text{Distributed}) \quad (2)$$

As part of future work, the paper can be extended by experimenting on high load and comparing results.

References

1. E.W. Dijkstra, *Hierarchical Ordering of Sequential Processes* (Academic Press, 1972)
2. S. Oaks, H. Wong, *Java: Threads* (O'Reilly & Associates, Inc., 1999)
3. R. Stewart, J. Singer, *Comparing Fork/Join and MapReduce*
4. J. Ponge, *Fork and Join: Java Can Excel at Painless Parallel Programming Too!* (2011). <https://www.oracle.com/technical-resources/articles/java/fork-join.html>
5. S.P.A.R. Quintao, *Performance Evaluation of Java RMI: A Distributed Object Architecture for Internet Based Applications*
6. D. Lea, *Concurrent Programming in Java: Design Principles and Patterns* (1997)
7. N.S. Nair, A. Mohan, S. Jayaraman, Interactive exploration of compact sequence diagrams—JIVE based approaches, in *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*. <https://doi.org/10.1109/ICSSIT48917.2020.9214261>
8. S. Jayaraman, B. Jayaraman, D. Lessa, Compact visualization of Java program execution. *Softw. Pract. Exper.* **47**, 163–191. <https://doi.org/10.1002/spe.2411>
9. K. Jevitha, S. Jayaraman, M. Bharat Jayaraman, Sethumadhavan, *Finite-State Model Extraction and Visualization from Java Program Execution* (Practice and Experience, Software, 2020). <https://doi.org/10.1002/spe.2910>
10. R. Sharp, A. Rountev, Interactive exploration of UML sequence diagrams, in *3rd IEEE International Workshop on Visualizing Software for Understanding and Analysis*. <https://doi.org/10.1109/VISSOF.2005.1684295>
11. P. Gestwicki, B. Jayaraman, Methodology and architecture of JIVE, in *Proceedings of the 2005 ACM Symposium on Software Visualization, SoftVis'05* (ACM, New York, NY, USA, 2005), pp. 95–104
12. S. Jayaraman, D. Kishor Kamath, B. Jayaraman, Towards program execution summarization: deriving state diagrams from sequence diagrams, in *2014 Seventh International Conference on Contemporary Computing (IC3)*. <https://doi.org/10.1109/IC3.2014.6897190>
13. A.A. Aziz, M. Unny, S. Niranjana, M. Sanjana, S. Jayaraman, decoding parallel program execution by using Java interactive visualization environment (JIVE): behavioral and performance analysis, in *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*. <https://doi.org/10.1109/ICCMC.2019.8819754>

14. Distributed Mutex. <https://github.com/01shobitha/Distributed-Mutex/tree/master/Dining-Philosopher>. Last accessed 30 June 2021
15. S.K. Gandhi, P.K. Thakur, *Analysis of Mutual Exclusion Algorithms with the significance and Need of Election Algorithm to solve the coordinator problem for Distributed System* (2013)
16. K. Karippara, A.D. Nayar, V. Illikkal, N. Vasudevan, S. Jayaraman, Synthesis, analysis and visualization of networked executions, in *2020 5th International Conference on Communication and Electronics Systems (ICCES)*. <https://doi.org/10.1109/ICCES48766.2020.9138091>
17. J. Maassen, R. Van Nieuwpoort, R. Veldema, H. Bal, T. Kielmann, C. Jacobs, R. Hofman, Efficient Java RMI for parallel programming. *ACM Trans. Programm. Lang. Syst.* (2001)
18. J.M. Kannimoola, B. Jayaraman, K. Achuthan, Run-time analysis of temporal constrained objects, in eds. by D. Seipel et al. (DECLARE 2017, LNAI 10997, Springer Nature, Switzerland AG, 2018), pp. 20–36. https://doi.org/10.1007/978-3-030-00801-7_2
19. M. Abraham, K.P. Jevitha, *Runtime Verification and Vulnerability Testing of Smart Contracts*, eds. by M. Singh et al. (ICACDS 2019, CCIS 1046, Springer Nature Singapore Pte Ltd. 2019), pp. 333–342. https://doi.org/10.1007/978-981-13-9942-8_32
20. C. Artho, A. Biere, Applying static analysis to large-scale, multi-threaded Java programs, in *Proceedings 2001 Australian Software Engineering Conference*. <https://doi.org/10.1109/ASWEC.2001.948499>

Message Forwarding Scheme with Max-Delivery and Min-Delay for Delay Tolerant Network



Sudhakar Pandey, Nidhi Sonkar, Sanjay Kumar, Danda Pravija, and Sanchit Mahto

Abstract Delay tolerant networks are the best and reliable network in state of emergency such as earthquakes by enabling communication without end-to-end connectivity. For creating communication, store-carry-forward technique is used, that means, if connectivity does not exist between nodes, they store the message till connectivity does not exist and then transfer to other nodes. In this study, we proposed a protocol that tried to deliver the message to the destination node by selecting the best intermediate node based on three features: speed of the node, residual energy of the node, and distance between the neighbor nodes. We also tried to minimize the delay and maximize the delivery ratio by increasing the transmission speed. We simulate our proposed protocol on ONE simulator and compared our method with other three best pre-found protocols. The experimental results convey that our protocol has achieved the delivery ratio of 90% and minimized the delay 4600 s.

Keywords Delay minimization · Delay tolerant network (DTN) · Delivery ratio · ONE simulator · Routing · Opportunistic networks

1 Introduction

Delay tolerant networks (DNT) known as intermittently connected networks in open literature are modeled in such a way that they can transfer data in difficult environments such as during natural calamities like earthquakes, cyclones where devices may lack of continuous network connectivity. Communication plays an important role during natural calamities. Loss of communication or network connectivity leads to the delay of equipment and rescue operations [1]. Hence, it is important to have stable network connectivity. DTN [1] uses fault tolerant methods and technologies in order to achieve continuous network connectivity. Delay tolerant networks [2] are used in many areas such as military, oceans, and in disaster-prone areas. The performance of the delay tolerant networks mainly relies on the use of efficient routing protocols. Many routing protocols [3] are proposed in the past ten years.

S. Pandey · N. Sonkar (✉) · S. Kumar · D. Pravija · S. Mahto
Department of Information Technology, National Institute of Technology Raipur, Raipur, India

In MinVisted [4] protocol, they have considered the two parameter, i.e., farthest node and number of encounters to choose intermediates that deliver the message to destination node. By taking those parameters, they have successfully reduced the number of hops, but there is transmission delay in the network. In our protocol, we have taken transmission speed, residual energy of the node, and distance between the nodes as parameters in order to further minimize the message transmission delay in the network.

Here, we proposed a new protocol that aims to transmit a message from the origin node to an objective node with minimum transmission delay and maximum delivery ratio. We have achieved this by choosing a node that is less distance from the current node, and at the time, the node must have maximum energy and high transmission speed. In this way, we can increase the probability of reaching a neighbor node in minimum time and, thus, results in reaching the destination node faster with maximum delivery ratio. We evaluated our protocol using the ONE simulator which is popularly used specially for DTN [5]. We compared the protocol with some popularly known protocols such as FirstContact, spray and wait, and MinVisited [4].

2 Related Work

There are two categories of protocols for transmitting message in DTN [6]. Flooding-based protocol and forwarding-based protocol.

In forwarding-based, at any given instance, a solitary duplicate of message is present in the network which permits to lessen the inefficient use of network resources. For example, FirstContact protocol [7].

In flooding-based protocols, all the node just flood the message to every encountered node. That means, when any node wants to send message to the destination, it just forwards message to the all node that encounters in the communication range of that node and then every node follows the same procedure till the message does not send to the destination. Epidemic [8] and spray and wait [9] protocols are examples of flooding-based protocols. Epidemic routing protocol follows full-flooding protocols that means node spreads message to the all encountered node till the message transmission to the destination. Spray and wait follows the $L/2$ flooding protocol that means node transmits $L/2$ number of copies to the encountered node and wait. If the message is transferred to the destination, the transmission is done, and if message is not transmitted to the destination, it again transfers $L/2$ copies to the encountered node.

Maxprop [10] protocol is considered to sort out which bundles should be erased when supports are space low-lying. This convention organizes the less number of leaps to the last nodes.

PROPHET is routing protocol of delay tolerant networks that extends for probabilistic routing protocol [11] using the history of encounters and transitivity. In this protocol, nodes use history of encounters of other nodes that they travel in the

history. Node finds the probability of each that they can transmit the message to the destination.

MinVisit [4] is a protocol that considers the distance of node and number of encounters to find the suitable node for transmission to minimize the hop count. But, it did not focus on the delay between nodes in transmission that is important factor in delay tolerant networks, so for recovering this problem in this paper, we proposed a MaxDelivery with minimum delay technique to minimize the delay between the source and destination and increase the delivery ratio by considering the parameters of node, that is, speed, distance, and energy of the node. Here, we can choose the node that has the highest speed among all the encountered node, and at the same time, the node must have the highest energy, and it should be the nearest node among the all encountered nodes.

The whole scenario of delay tolerant network is irregular [12], and we have to simulate the environment with this irregularity. The ONE simulator [13] is specially designed for this type of environment and applications of DTN so that in this paper, we simulate the environment in ONE simulator for getting the exact results.

3 Proposed Methodology

Our proposed routing method aims to transmit the message from source node to the destination node with high delivery ratio and minimum delay in the network. Each node has their own transmission speed that is dynamic in nature, i.e., changes during the runtime. We have considered three parameters for selection of relay node, i.e., speed, residual energy, and distance from the source node. To achieve the goal, protocol follows some rules:

1. There is certain fixed number of duplicates of all messages that is refreshed as message are effectively conveyed to one of two, i.e., objective node or to the halfway node.
2. The message can be transmitted on a small number of neighboring nodes during a period of time (or window time).
3. Each node maintains a table called vector table that consists of a pair of key and value that maps the distance between a node and their neighboring node.
4. A node can only transmit the message to a node, which has the best transmission speed. That node is termed as intermediate halfway node.

There is time to live (TTL) for every message inside a node. Once a time to live has terminated, then the node erases the message. The count of intermediates can be altered to expand the reachability of the convention or to diminish the traffic correspondence.

- In Fig. 1, at the time instance T_0 , the origin node A wants to transmit message to Node P. “#4” denotes the number duplicates of message. There is a vector table of distance between neighbor’s node and the source node that every node in the

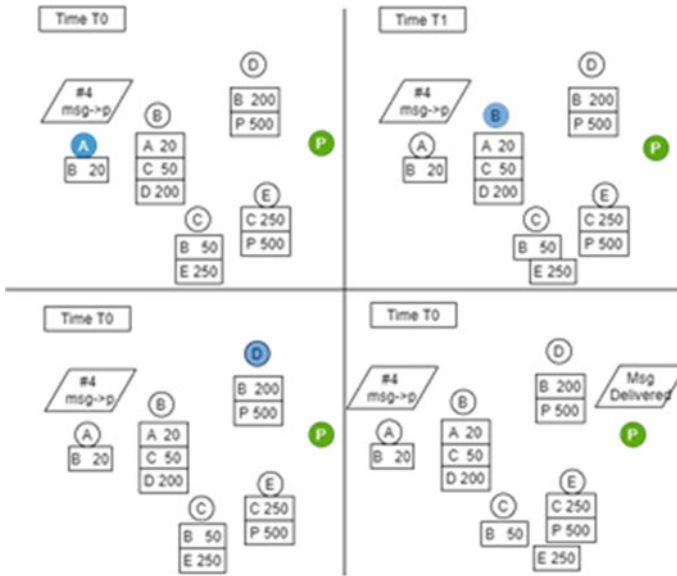


Fig. 1 Steps of algorithm

network maintains. In the transmission path of objective node, Node A finds Node B. From the vector table, it seems B is nearest because it has distance than any other node. And, we assume in the runtime if speed and energy of Node B is greater than speed of any other node. In this manner, B is chosen as an intermediate node to transmit the message.

- In time T1, source node B wants to transmit the carrier message to destination node P, but it is not in the direct contact of B. Therefore, it has to again find the intermediate node. So, Node B has neighboring nodes as D and C. Since distance of D is less than distance of C, therefore, Node D is chosen as the intermediate node for B.
- In time T2, source node D wants to transmit to destination node P. Since node P in the range to node D, i.e., it has direct contact with the source node. Therefore, it will deliver the message to the destination node.
- At long last, on the off chance that a node has a solitary duplicate of the message, that node holds up until and unless it experiences the objective node, i.e., the node looking for transmitting does not look for a transition. The confirmation step is shown in Fig. 2.

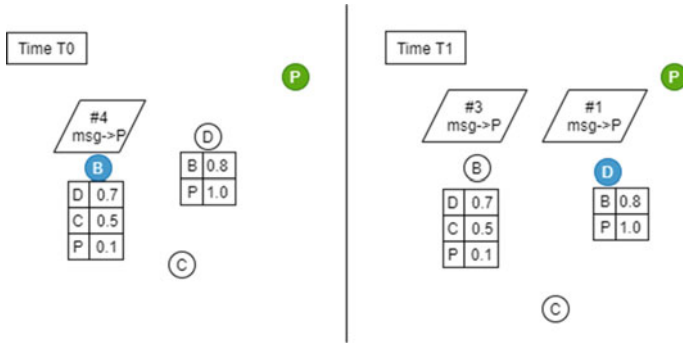


Fig. 2 Confirmation step

4 Simulation and Results

We tried to perform simulation on for delay tolerant network (DTN) using opportunistic networking environment (ONE) simulation. The simulation was performed on the set of parameters given in Table 1.

For evaluation of performance of our method, we have taken two criteria into consideration, i.e., count of nodes and buffer size.

4.1 Number of Nodes

Figure 3 represents the delivery ratio of the method as the count of nodes in the network increases. We can say that as the number of nodes increases, delivery ratio increases. Figure shows that our method has the best performance among others.

Figure 4 shows the message delay of the method as the number of nodes increases. Our method tends to minimize the message delay in the network. Figure shows as the number of nodes increases, the delay decreases.

Table 1 Experiment parameters

Parameters	Values
Message size	100 kb
Buffer size	0–30 mb
Transmission range	100 m
Time to live (TTL)	300 min
Number of nodes	0–50
Simulation time	5000 s
Data rate	2048 bytes per second
Mobility model	Random waypoint

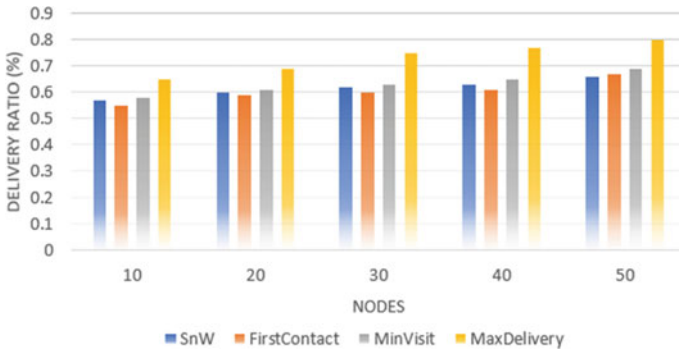


Fig. 3 Delivery ratio versus number of nodes

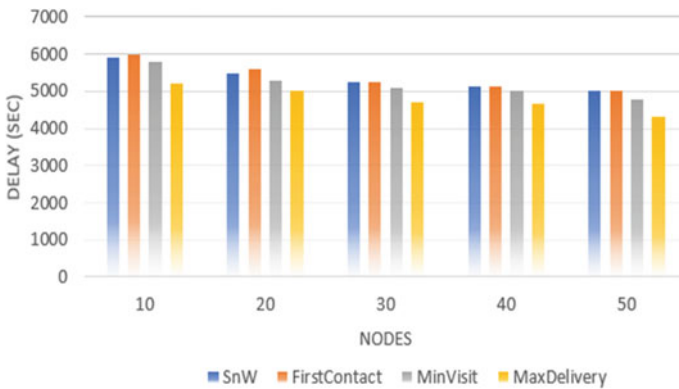


Fig. 4 Delay versus number of nodes

4.2 Buffer Size

We tried to show the delivery ratio and delay using buffer size with different sizes, ranging from 0 to 30 MB. With different buffer size, our method shows as close to best method which is spray and wait protocol.

In Fig. 5, we can interpret that as the size of buffer increases, delivery ratio increases, but around 15 MB, our method tries to stabilize the delivery ratio.

Figure 6 shows that as the buffer size increases, delay increases, but after 16 MB, it stabilizes, and it shows the performance as close to the spray and wait protocol.

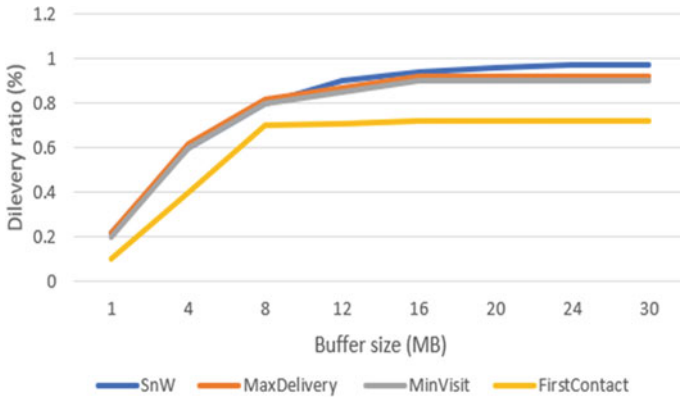


Fig. 5 Delivery ratio versus buffer size

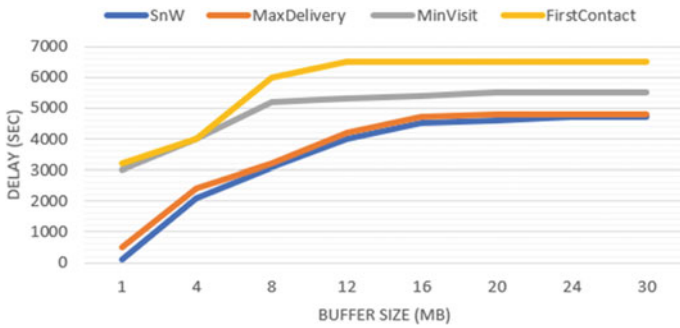


Fig. 6 Delay versus buffer size

5 Conclusion

We proposed a new protocol named MaxDelivery with minimum delay technique in delay tolerant networks. The protocol aims to deliver messages to the neighbor which has opportunity to transmit the message in minimum time by considering the transmission speed and energy of each node and distance between the nodes. Additionally, this convention is assessed against some notable protocols, for example, FirstContact, SnW, and MinVisit by thinking about various boundaries and by utilizing various measurements of the specialized writing. Results show that our proposal reports the successful message delivery ratio of 90% which is greater than many protocols such as FirstContact, SnW, and MinVisit with delay of 4600 s when the size of buffer is more than 16 MB.

6 Future Work

In this study, we proposed a protocol to deliver the message from origin node to destination node by taking the transmission speed, energy, and distance as a parameter which help to minimize the transmission delay time and maximize the message delivery ratio. We have taken two parameters in our protocol. In the future, the challenge is to take different parameters and try to further minimize the delay time and increase the message delivery ratio. Also, self-learning technologies like machine learning and artificial intelligence can also be applied to obtain the best selection value for the protocol that will give the maximum performance.

References

1. E. Wang, W.S. Yang, Y.J. Yang, J. Wu, W. Bin Liu, An efficient message dissemination scheme for minimizing delivery delay in delay tolerant networks. *J. Comput. Sci. Technol.* **33**(6), 1101–1124 (2018). <https://doi.org/10.1007/s11390-018-1875-7>
2. MILCOM 2008-2008, *IEEE Military Communications Conference* (IEEE, 2008)
3. K. Massri, A. Vitaletti, A. Vernata, I. Chatzigiannakis, Routing protocols for delay tolerant networks: a reference architecture and a thorough quantitative evaluation. *J. Sens. Actuator Netw.* **5**(2) (2016). <https://doi.org/10.3390/jsan5020006>
4. L. Veas-Castillo, G. Ovando-Leon, V. Gil-Costa, M. Marin, MinVisited: a message routing protocol for delay tolerant network, in *Proceedings—26th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing, PDP 2018*, Jun 2018, pp. 325–328. <https://doi.org/10.1109/PDP2018.2018.00057>
5. A. Keränen, J. Ott, T. Kärkkäinen, The ONE simulator for DTN protocol evaluation (2009)
6. E. Rosas et al., Survey on simulation for mobile Ad-Hoc communication for disaster scenarios. *J. Comput. Sci. Technol.* **31**(2), 326–349 (2016). <https://doi.org/10.1007/s11390-016-1630-x>
7. S. Jain, K. Fall, R. Patra, Routing in a delay tolerant network, in *Proceedings of the 2004 Conference on Applications, Technologies, Architectures, and Protocols For Computer Communications—SIGCOMM'04*, 2004, p. 145. <https://doi.org/10.1145/1015467.1015484>
8. A. Vahdat, D. Becker, *Epidemic Routing for Partially-Connected Ad Hoc Networks*. Accessed Dec 01, 2020 [Online]
9. T. Spyropoulos, K. Psounis, C.S. Raghavendra, Spray and wait: an efficient routing scheme for intermittently connected mobile networks, in *Proceedings of the ACM SIGCOMM 2005 Workshop on Delay-Tolerant Networking, WDTN 2005*, 2005, pp. 252–259. <https://doi.org/10.1145/1080139.1080143>
10. J. Burgess, B. Gallagher, D. Jensen, B.N. Levine, MaxProp: routing for vehicle-based disruption-tolerant networks, 2006. <https://doi.org/10.1109/INFOCOM.2006.228>
11. A. Lindgren, A. Doria, O. Schelén, Probabilistic routing in intermittently connected networks. *ACM SIGMOBILE Mobile Comput. Commun. Rev.* **7**(3), 19–20 (2003). <https://doi.org/10.1145/961268.961272>
12. N. Choudhary, K. Mittal, K.D. Kota, P. Nagrath, S. Aneja, Analyzing and designing energy efficient routing protocol in delay tolerant networks. *CSI Trans. ICT* **4**(2–4), 285–291 (2016). <https://doi.org/10.1007/s40012-016-0094-z>
13. M.J. Khabbaz, C.M. Assi, W.F. Fawaz, Disruption-tolerant networking: a comprehensive survey recent developments and persisting challenges. *IEEE Commun. Surveys Tutorials* **14**(2), 607–640 (2012). <https://doi.org/10.1109/SURV.2011.041911.00093>

Design and Development of Access Control and Face Mask Detector in Real Time Using Deep Learning to Prevent COVID-19



Manu Gupta, Gadhiraaju Hari Priya, Nandikonda Archana Reddy, and A. Sanjana

Abstract After the breakout of this worldwide pandemic situation COVID-19, there arises a severe need for protection mechanisms, wearing a face mask being the primary one. The main aim of the project is to detect the presence of a face mask on human faces on real-time live streaming video. The proposed model is developed using MobileNetV2 which is a deep learning algorithm. The architecture takes the image as input, assigns weights to various objects in the image, differentiates one from another and the neural network output which tells us whether there is a mask or not, and the result is given to the Arduino module by using PY serial software. This model gives an accuracy of 99.9%, and it is connected to the servo-motor which is attached to the Arduino and acts as an automatic sensor door present at various public places. The door will be opened or remains closed based on the output value given to Arduino by the mask detection model designed in proposed study. The door opens only when a person is wearing a mask; otherwise, it remains closed.

Keywords MobileNetV2 · Face mask detection · PY serial · Access control · Deep learning

1 Introduction

Due to the increase in COVID-19 cases, taking precautions has become necessary. One of the most important things is to wear a mask. It has become a necessity for everyone to follow this. People are not allowed to go to many public places without masks. This leads to the need of a person to check who is wearing and who is not and give access to people at a given place according to it. However, this takes up a lot of manpower, so to reduce this, automated solution is required. Face mask detection is one such idea that eradicates human presence and makes use of technology to give access to a person [1]. COVID-19 has led to the death of many people, and the rate at which people are getting infected is growing. A lot of countries that are affected by this virus are increasing.

M. Gupta (✉) · G. H. Priya · N. A. Reddy · A. Sanjana
Department of ECM, Sreenidhi Institute of Science and Technology, Hyderabad, India

It started in China on December 31st, 2019, at Wuhan and emerged in various countries [2]. The coronavirus transmission is occurring due to the spread of virus-containing airborne particles, and the contraction of this infectious disease is caused when they are in close proximity [3]. In India, the mortality rate is around 354,000, and lakhs of people are dying every day [2]. The main cause of spreading is lack of concern and less awareness. Daily, people are getting access to their offices and shopping malls without any mask, and it is very hard to keep an eye on people every second. The face mask detection software helps people to understand and to make them aware of the importance of wearing a mask. Nowadays, there have been great advancements in the security systems which lead to a remarkable change in our daily life, and these changes help in reduction of manpower. So, it is important that we create a system that monitors people daily. The face mask recognition system monitors people and checks whether a person is wearing a mask or not. From the survey carried in Maryland, the statistics show that if a person wears a mask, 95% of the infections can be avoided [4]. This shows how important it is to wear a mask, and technology helps us in ensuring people to wear masks. Machine learning and artificial intelligence are the technologies that have shown technical advancements and are proven to provide solutions to problems without human intervention [5].

In this paper, we have proposed a face mask detection model that will check whether a person is wearing a mask or not using image processing algorithm MobileNetV2 [6] which is a convolutional neural network (CNN) architecture. The purpose of using CNN is it takes images as input and allocates importance to various features present in the image in order to differentiate them. It classifies images accordingly, and it does not require much preprocessing compared to other models. It is one of the most efficient and accurate image classification algorithms. Our model detects people who are wearing masks accurately and grants access by allowing people to enter the door. The door is controlled by a servo-motor which is designed to rotate at a particular angle as soon as it encounters a person with a mask automatically. This helps in reducing manpower and gives more accurate results.

2 Literature Survey

Wearing a face mask has become mandatory in our daily life. Due to advancements in technology, multiple systems with face mask detection software have been implemented. There has been the usage of various machine learning models and methods to detect a face. Some of the methods from literature [7–16] are discussed as follows:

An automated system by Rahman et al. [7] detects the mask of a person using convolution neural networks. The images are taken from real-time video using surveillance. It uses feature extraction that means when the face is detected, it extracts all the features. The system proposed by Matthias et al. [5] uses a deep learning model in order to detect a face mask. It captures the person in real time and notifies the respective person in charge. The face mask detection presented by Redmon et al. [8] uses you only look once (YOLO) for the identification of masks. It is faster and is

useful for object identification. Das et al. [9] used CNN for the image recognition, and various libraries were imported for the detection. TensorFlow, OpenCV, and Keras are used in this work to identify the mask and also check whether the mask is N95 or not. Sanjaya et al. [10] presented a mask detection using two datasets, one is a masked dataset, and the other is a Kaggle dataset. Convolution neural networks are required for image recognition, and the dataset which is trained is applied to people in daily life. This is used for checking the percentage of people who are wearing masks in the city. The work proposed by Jaiswal et al. [11] used convolutional neural networks for visualization. Here, the face mask model used is retina masks, and it gives good precision. It detects masks and spreads the information about the people who wear masks or not using IoT-based sensors, and it gives notification to the authorities. Islam et al. [12] used CNN for image recognition. In this, the complexity is less and it is faster. The face mask is detected in the live stream, and it checks whether a person is wearing a mask or not. It gives a security alert indicating that there is no mask. The authors in [13–16] also used deep learning-based model for face mask detection. Table 1 provides the performance analysis of different face mask detection systems from literature.

Drawbacks of Existing System

Most of the existing systems are not able to give the best accuracy, and it is taking time to detect a face. The algorithms used are slower and complex. Even though there have been systems like detecting faces and notifying the officer, it actually takes time for the authority to reach the destination. Due to the large movement of faces, there

Table 1 Performance analysis of different methods from literature

Authors	Method used	Accuracy
Rahman et al. [7]	Convolutional neural network, max pooling,	The model detects a face mask with an accuracy of 98.7%
Redmon et al. [8]	You only look once	Accuracy obtained was 63.4%
Das et al. [9]	Sequential convolutional neural network	The model detects a face mask with accuracy 94.58%
Sanjaya et al. [10]	MobileNetV2	Accuracy obtained was 96.85%
Islam et al. [12]	Convolutional neural network	The model detects a face mask with accuracy 98%
Meenpal et al. [4]	Fully convolutional networks	Accuracy obtained was 93.884%
IsunuriB et al. [13]	MobileNetV2 and global pooling	The accuracy of the model was 98%
Gupta et al. [14]	CNN model, Haar cascade algorithm	The accuracy of the model on image dataset was 95%
Jiang et al. [15]	CNN, MobileNet, ResNet	The accuracy of the model was 89% when MobileNet was used, and it was 94% when ResNet was used
Batagelj et al. [16]	MTCNN, MobileNetV2, OCVSSD, AlexNet	The overall accuracy for the model was 97%

may be some inconsistencies in capturing the image of a person which will lead to wrong results. One of the major drawbacks of the existing systems is they did not mention the following step after face mask detection. In public places after detection, what we need is access to the place. However, the proposed systems above require manpower to do this which is a time-consuming and subjective process. Considering the pandemic situation, everything needs to be digitized as it is necessary to protect the authorities or customers who are visiting by making the system more reliable and accurate.

3 The Proposed Model

The proposed model consists of a face mask detection system that checks whether a person is wearing a face mask [17] or not with the help of a real-time video stream integrated into the system using OpenCV library. The captured image will be passed onto the face mask detection model which is built using MoblieNetV2 which is a convolutional neural network model integrated with the face detection model to check whether a person is wearing a face mask or not. Here, we are using a servo-motor as an access control system. If the person is wearing a mask, it gives access to a person by rotating the motor, or else the motor will not rotate which indicates that access is not granted. The advantage of our proposed model is it restricts people who are not wearing masks and violating COVID-19 guidelines. With the help of this system, we can reduce the workforce; i.e., there is no need for manual checking whether a person is wearing a mask or not by a security official at the entrance. It automatically gives access as soon as it identifies the person who is wearing a mask properly.

3.1 Data Preprocessing

In the dataset, there are two types of images; one is masked face images, and another is unmasked face images. In total, there are 3833 images, out of which 1915 are masked face images and the remaining 1918 are unmasked face images. From this, we give 3066 images to training and 766 images to testing. The dataset was taken from Kaggle. Sample dataset is shown in Fig. 1. Initially, the dataset had a lot of noise and lots of duplicate images. The image quality issue can be solved by using the dataset cleaning process, whereas the quantity issued can be solved by using the image augmentation method. Under this method, we have a function known as image data generator. This image data generator is used for creating many images using a single image by adding various properties to a single image like flipping, rotating, shifting, zooming in and zooming out, and many other properties of an image.

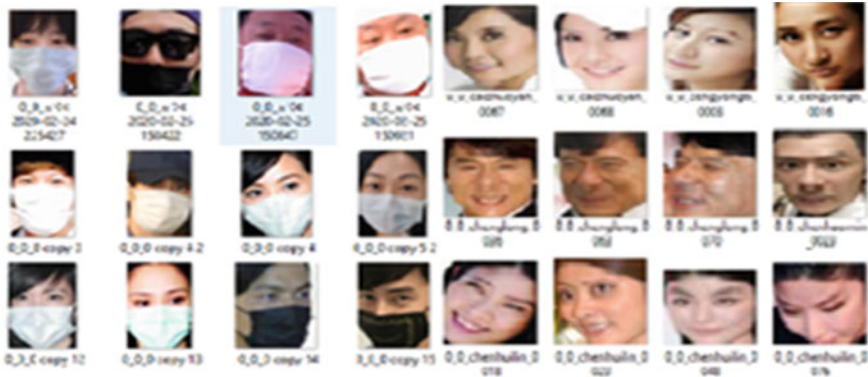


Fig. 1 Sample from dataset

3.2 Packages Incorporated

TensorFlow: TensorFlow is a free- and open-source software library for data flowing symbolic differentiation and differentiable programming across a spread of multiple tasks. TensorFlow uses data flow graphs to build models. In the proposed system, TensorFlow is used in the backend for image recognition.

Keras: Keras is an open-source-software library that has a Python interface; for artificial neural networks, Keras acts as an interface for the TensorFlow library. Here, Keras is used to convert dataset images into arrays. By converting it to arrays, it becomes easy to build a deep learning model.

OpenCV: OpenCV is an open-source and cross-platform library using which we will develop real-time computer vision applications, which is employed to find the difference between them and recognize objects, faces, group movements in recordings, track eye movement, track camera movements, find comparative pictures from a pictures database. It mainly focuses on image processing, video capture, and analysis including features like face detection and object detection.

MobileNetV2: MobileNetV2 is a CNN-based model which tends to perform well on mobile devices. MobileNetV2 generally has two layers. One is the base model, and the other is the head model. MobileNet is used to build base models. In the base model, we do preprocess using MobileNet.

PySerial: PySerial is a library that is employed to attach Arduino code and Python code. This module encapsulates the access for the serial port. It provides backends for Python running on Linux, Windows, OSX, BSD, and IronPython. The module named “serial” automatically selects the suitable backend.

3.3 *Hardware Used*

Arduino: Arduino is an open-source electronics platform which is based on easy-to-use hardware and software. Arduino boards are able to read inputs—light on a sensor or a Twitter message—and switch it into an output—activating a motor, turning on an LED, a finger on a button, publishing something online. We can even tell our board what to do by sending a collection of instructions to the microcontroller on the board. Designs of Arduino board use a range of controllers and microprocessors. In this project, we use Arduino UNO.

Servo-motor: Servo-motor is a rotary actuator or linear actuator that enables for precise control of angular or linear position, velocity, and acceleration. A servo-motor may be a self-contained electrical device that rotates parts of a machine with high efficiency and with great precision. It consists of an appropriate motor coupled to a sensor for position feedback. The output shaft of this motor is moved to a selected angle, position, and velocity that a usual motor does not have. Servo-motor utilizes a usual motor and couples it with a sensor for positional feedback. In this project, it is used for showing access control.

3.4 *Architecture*

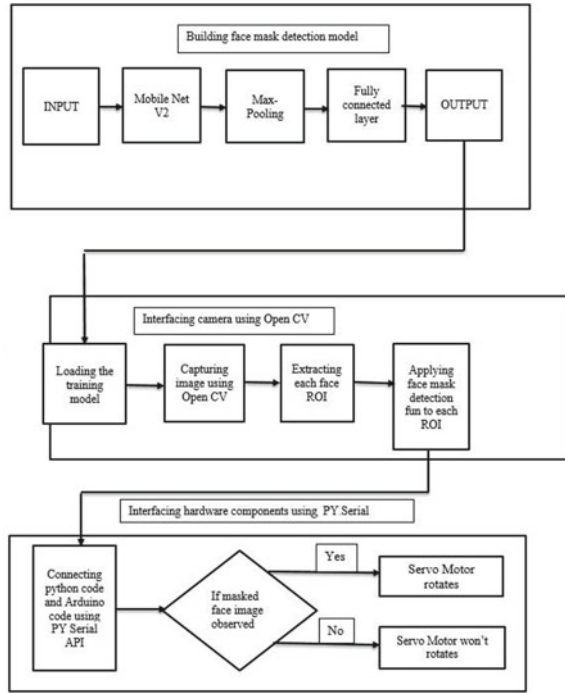
Figure 2 shows the architecture of the proposed model. It has three main sections: (i) building face mask detection model, (ii) interfacing camera, and (iii) interfacing hardware components. The detailed explanation of each section is given as follows:

(i) **Building Face Mask Detection Model**

The mask detection model is built using the MobileNetV2 algorithm. MobileNetV2 [15] is an image classification algorithm that is used for assigning importance to various objects in an image to differentiate one feature from the other features in the image. MobileNetV2 algorithm has a function known as ImageNet which has some pre-trained model weights, and this function can be assigned as the model weight while training the model. This ImageNet function is imported from the TensorFlow package. There are some pre-trained models for images, so when ImageNet is used, those weights will be assigned to the training model, and better results are obtained. The base model layers freeze, so that they will not change while ongoing development of the model. When the new training layers are added, these added layers are trained on a dataset to identify and determine the features needed for classifying whether a person is wearing a mask from the one who is not wearing. After the training process, the model is tuned with its weights.

Now, once the base model is constructed, its output is as input for building the head model which is used for the construction of a fully connected layer. To construct a fully connected layer, we will be doing max pooling [13]. Max pooling is done

Fig. 2 Architecture of the proposed model



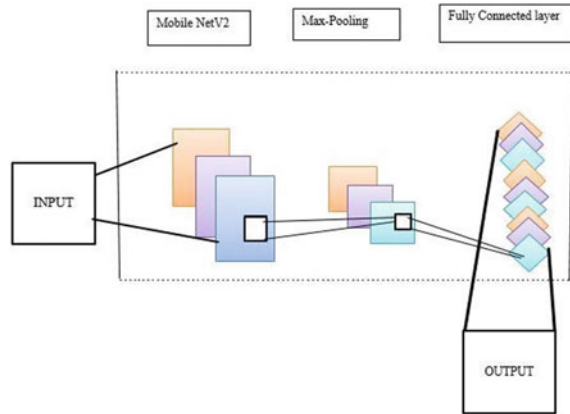
for selecting the most prominent features for image recognition from an image. Additional layers were added at the end of the model to achieve good accuracy and to save the model from overfitting. A flattening layer was added to map pooled features into a single column so that they can be easily passed on to a fully connected layer. A dense layer of 128 neurons is created with ReLU as its activation function; this dense layer adds a fully connected layer to a neural network. We have also added a dropout layer to avoid overfitting of our model, which occurs during the training process. The flow diagram of proposed mask detection model is shown in Fig. 3.

(ii) Camera interfacing

The mask detection model was developed for the detection of masked faces, and for face detection, we have used a face detection algorithm known as the Haar cascade algorithm. Haar cascade is a machine learning object recognition algorithm that recognizes objects in images and videos. The face detection model and mask detection model are integrated to check whether a person is wearing a mask or not. So, now with the help of the face detection model, the model will detect the face of the person, and with the help of the mask detection model, the model will check whether a mask is present on the face of a person. For camera operations, the OpenCV library was used.

To load the camera, the video stream function from OpenCV software was used. For face mask detection, a function called detect and predict mask was created which

Fig. 3 Mask detection model flow diagram



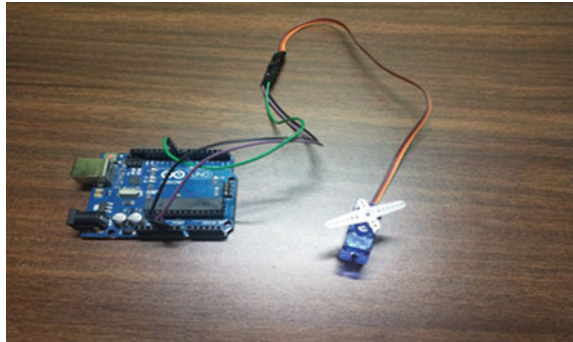
has three arguments, first one is the mask-net variable which is loaded with mask detection model, the second one is the facing-net variable which is loaded with face detection model, and the last one is a frame which is the image read by the camera using a `vs-read` function from OpenCV package. Every image is a frame. The movement of these frames per second at the illusion of our eye is seen as video. The `input-tensor()` method was used for giving the shape of the image going through the model which is (224, 224), and we are giving a three-channel color image which is RGB.

The defined function returns location and predictions. Where location is the x and y coordinates of the rectangular box surrounding the frame and prediction is the percentage accuracy of the person wearing a mask, it will be like 99% he is wearing a mask and 10% he is not wearing a mask. And to make our video streaming more visually appealing, we have drawn a rectangular box around the frame, and the color of the rectangular box is given by RGB colors. The box will be displayed green only when a person is wearing a mask; otherwise, the box will be displayed red. The proposed model even works with multiple images, when the model encounters more than one face in a frame it works accordingly.

(iii) **Interfacing hardware components**

For interfacing the Python code and the Arduino code, we have used the PySerial library. It is an API that connects the software part with the hardware part. For this connectivity, the serial package was imported in Python code, and a loop was given that runs only when the image of a person with the mask is captured by the camera; otherwise, the loop will not run. Within this loop, the servo-motor rotational angle was given which is to be given to the hardware component which rotates only when an image of a person with the mask is captured. Arduino IDE was used for executing servo-motor code connected to the Arduino. In the Arduino IDE, the port no. of the Arduino was mentioned where the servo-motor is connected to it and has also mentioned the baud rate. By default, the baud rate of Arduino is 9600 bps. And, the position of the servo-motor is read by using serial API which was mentioned in

Fig. 4 Hardware connections



the Python code. So, as we have mentioned rotation angle and instruction regarding motor rotation in Python code, these instructions will be transferred to Arduino using PySerial API. So, now, the servo-motor gives access by rotating only when the image of a person with the mask is being observed; otherwise, access will be denied; that is, the motor will not rotate. The hardware connections are shown in Fig. 4.

4 Implementation

While training the model, a method known as image data generator is used. This image data generator is used for creating many images using a single image by adding various properties to a single image like flipping, rotating, shifting, zooming in and zooming out, and many other properties of an image. For optimizing our model, we have used the Adam optimizer algorithm. The model was evaluated by using the model-predict method. For this, the Numpy argument-max-method was used. Accuracy, precision, recall, and *F1*-score metrics were calculated in order to measure the accuracy of the model. To calculate these metrics, the confusion matrix was plotted. A confusion matrix as shown in Table 2 is a matrix that is used for describing the performance of a classification model on a set of test data for which the true values are known.

The performance parameters accuracy, precision, recall, and *F1*-score are calculated from obtained confusion matrix as explained below:

Table 2 Confusion matrix

	Actually positive (1)	Actually negative (0)
Predicted positive (1)	True positive (TP)	False positive (FP)
Predicted negative (0)	False negative (FN)	True negative (TN)

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

$$\text{Precision} = \text{TP} / (\text{FP} + \text{TP}) \quad (2)$$

$$F1\text{-score} = \text{TP} / (\text{TP} + 1/2(\text{FP} + \text{FN})) \quad (3)$$

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN}) \quad (4)$$

where, TP refers to true positive, TN refers to true negative, FP represents to false positive, and FN represents false negative.

5 Results

In this section, the performance evaluation of proposed model is carried out, and the values obtained for various metrics, i.e., accuracy, precision, recall, and $F1$ -score are demonstrated. The graphical representation of accuracy vs. loss is also provided which illustrates the accuracy of our model. Afterward, the images of various positions of a person with and without mask are displayed. Here, we have also displayed the accuracy with which our proposed model is able to detect the mask on faces in different images.

5.1 Performance Metrics and Analysis from the Graph

The graph demonstrating training loss, validation loss, training accuracy, and validation accuracy values is shown in Fig. 5. The learning rate was taken as $1e - 4$. It should be as small as possible to create a face mask detection model which produces the least loss rate and produces the best accuracy value. After training the model with various epoch values, epoch values of 20 is considered so as to prevent the overfitting and underfitting of the model. The graph was plotted with epochs on the x-axis and loss and accuracy metrics on the y-axis. From the graph shown in Fig. 5, it is observed that training loss and validation loss are almost close to zero, whereas the training accuracy and validation accuracy are almost close to one. This indicates that our model is 99.9% accurate. The precision, recall, and $F1$ -score values are obtained as 0.99 as shown in Fig. 5

	Precision	Recall	$F1$ -score
With mask	0.99	0.99	0.99
Without mask	0.99	0.99	0.99

(continued)

(continued)

Accuracy			0.99
Macro-average	0.99	0.99	0.99
Weighted average	0.99	0.99	0.99

5.2 Face Mask Detection Accuracy and Servo Motor Output for Single Image

The outputs displaying the accuracy with which the proposed system is detecting the mask and no mask faces are demonstrated in Figs. 6, 7, 8, 9, 10 and 11.

The image showing a person wearing mask properly with 100% accuracy is displayed as Sample Image 1 in Fig. 6.

The second sample image in Fig. 7 shows the side view of the masked face, and it is giving a green boundary box with mask percentage as 100%.

The third sample image in Fig. 8 shows that a person is wearing a mask below the mouth which is not considered as wearing a mask, and it is giving a red boundary box with no mask percentage as 60.48%.

Figure 9 shows the fourth sample image where a person is not wearing mask properly; it is been hanging, and it is giving a red boundary box with mask percentage as 100%.

The output sample demonstrated in Fig. 10 shows that a person is wearing a mask properly, and it is giving a green boundary box with no mask percentage as 100%. For this result, the servo-motor will rotate at specified angle making the door open for entry of the person.

The result displayed in Fig. 11 shows that a person is not wearing a mask, and it is giving a red boundary box with no mask percentage as 100%. For such cases, the servo-motor will not rotate, hence, denying entry to the person.

5.3 Face Mask Detection Accuracy and Servo-Motor Output for Multiple Image

The proposed model works for input dataset with multiple images also. When more than one face is detected in a frame, it works accordingly. As shown in Fig. 12, there are two people, so it efficiently detects and gives results as access denied and access granted based on whether the person is wearing mask or not.


```
[INFO] evaluating network...
      precision    recall  f1-score   support
with_mask          0.99      0.99      0.99         575
without_mask       0.99      0.99      0.99         575

 accuracy          0.99
macro avg          0.99      0.99      0.99        1150
weighted avg       0.99      0.99      0.99        1150
```



	Precision	Recall	F1-score
With mask	0.99	0.99	0.99
Without mask	0.99	0.99	0.99
Accuracy			0.99
Macro average	0.99	0.99	0.99
Weighted average	0.99	0.99	0.99

(a)

(b)

Fig. 5 a Graph illustrating accuracy versus loss, b performance metrics values

Fig. 6 Mask worn properly (sample image 1)

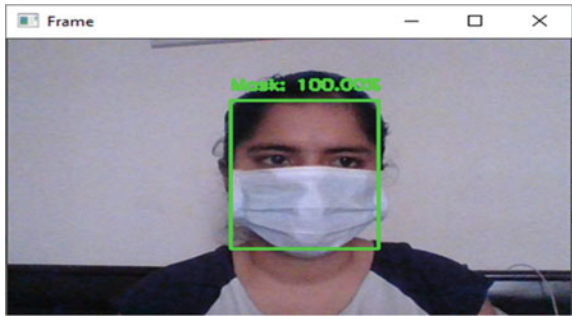
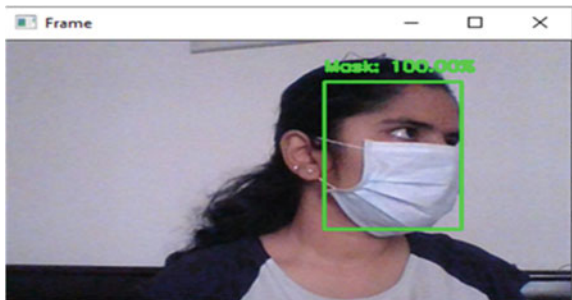


Fig. 7 Side view of masked face (sample image 2)



5.4 Performance Analysis of Proposed Model with Various Parameters

We have performed the analysis of a proposed system with different models. Here, firstly, we have taken epochs value as 40 and have trained the model which was

Fig. 8 Mask worn at neck (sample image 3)

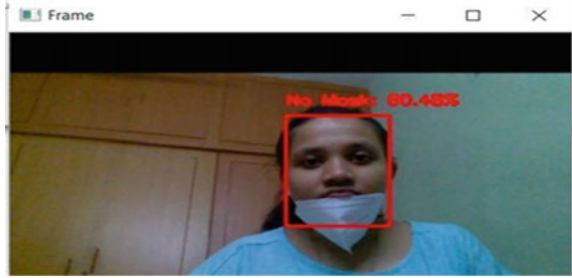


Fig. 9 Mask not worn properly (sample image 4)

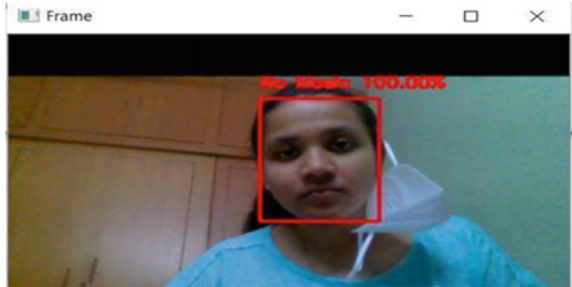


Fig. 10 With mask-access granted-motor rotated

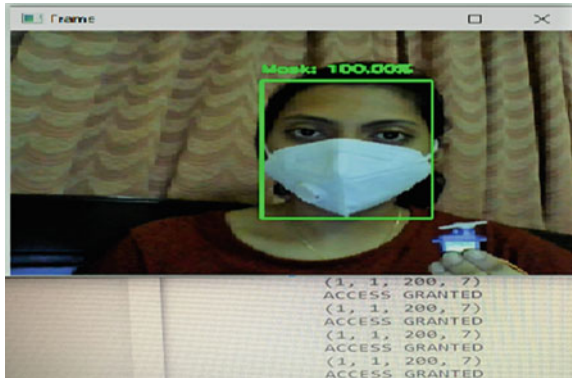


Fig. 11 No mask-access denied-motor does not rotate

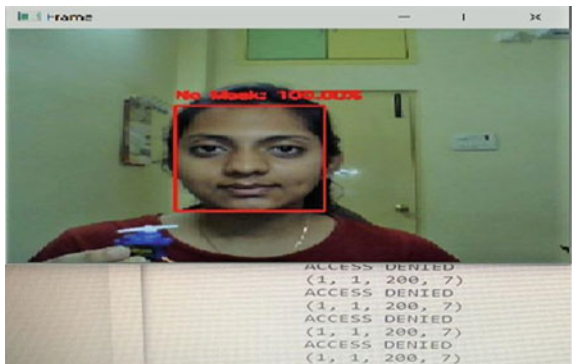
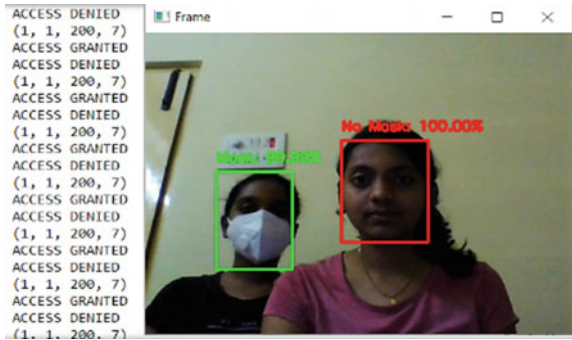


Fig. 12 Detecting multiple images at a time



built by using convolution neural network, and train and test split were taken as 90% and 10%, respectively. After training the model, we can observe from the graph that training loss is decreasing which is close to zero, but in contrast, the validation loss is increasing which is not a good sign. This results in the overfitting of the model which is shown in Fig. 13a.

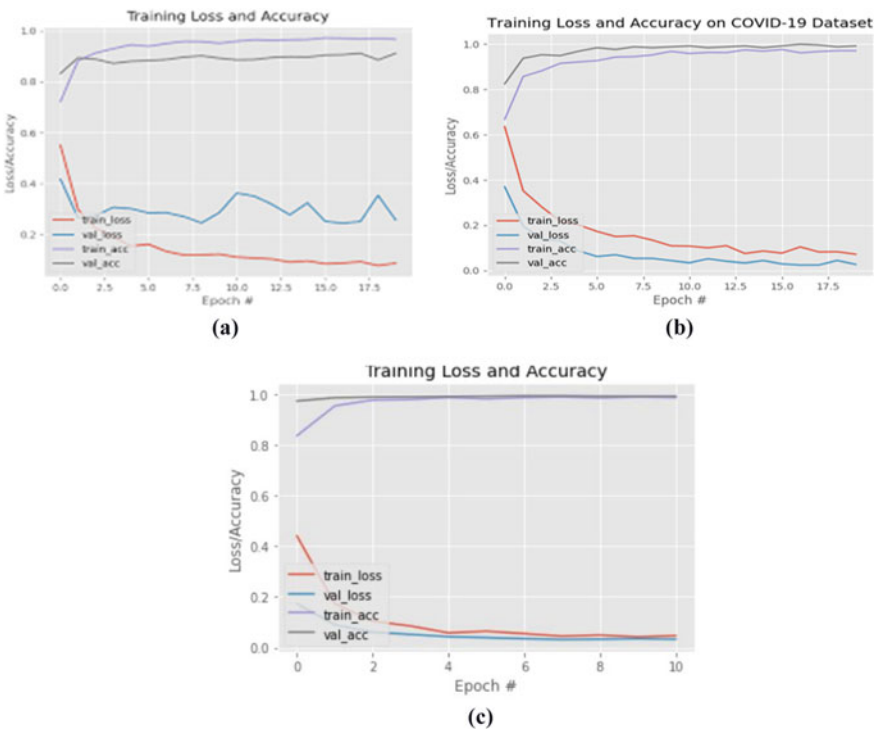


Fig. 13 a Model resulted as overfit, b model resulted as underfit, and c model resulted as best fit

In the next scenario, we have taken epoch value as 5 and have trained the model which was built by using a convolution neural network and YOLO. Train and test split were taken as 60% and 40%, respectively. After training the model, we can observe from the graph that validation loss is decreasing which is close to zero, but in contrast, the training loss is increasing which is also not a good sign. This results in the underfitting of the model which is shown in Fig. 13b. The observations in this situation are totally in contrast to the one we have done before.

Next, epochs value is considered between 5 and 20. After many trials and errors, we came to the conclusion that 11 is the best epoch value, and the same was done with train and text split which were finally taken as 70% and 30%, respectively. And, we have trained our mask detection model. It is observed that training loss and validation loss are close to zero, and the training and validation accuracies are close to one. And, our model accuracy is 99.9% as shown in Fig. 13c. This model gives the best accuracy and, hence, finalized for proposed design.

5.5 Comparison of Existing Model and Proposed Model

We have performed comparative analysis of proposed model and existing models for face mask detection as illustrated in Table 3. As observed, our proposed model achieved the highest accuracy of 99% in comparison to these existing methods. Moreover, in proposed study, an access control mechanism is also developed which provides entry to places only to those people who are wearing the mask properly. This feature is not present in existing systems. The face detection model with access control developed in this study reduces the manpower and makes model more reliable and accurate.

Table 3 Comparison of accuracy for existing and proposed model

S. No.	Models	Accuracy (%)
1	Rahman et al. [7]	98
2	Redmon et al. [8]	63
3	Das et al. [9]	94
4	Sanjaya et al. [10]	96
5	Islam et al. [12]	98
6	Meenpal et al. [4]	93
7	Isunuri et al. [13]	99
8	Gupta et al.[14]	95
9	Jiang et al. [15]	94
10	Batagelj et al [16]	97
11	Proposed model	99

6 Conclusion

In the proposed work, a face mask detector to prevent COVID-19 is developed using deep learning algorithms. This face mask detector detects people without masks and restricts entry. Thereby, it reduces the workforce in monitoring people. This model is developed using Python code and OpenCV. The servo-motor is connected to the Arduino to give access control. This automatic sensor door present at various public places will be opened or closed based on the output value generated in the main program. The proposed system can be used in many public places like shopping malls, hospitals, airports, educational institutions. The main objective of this work is to control the spread of coronavirus and make people to wear masks properly.

In the future, this work can be extended to impose a fine or challan on people who are not wearing masks properly in public places by capturing their pictures and notifying them on their mobile phones.

References

1. L. Martinelli, V. Kopilaš, M. Vidmar, C. Heavin, H. Machado, Z. Todorović, N. Buzas, M. Pot, B. Prainsack, S. Gajović, Face masks during the COVID-19 pandemic: a simple protection tool with many meanings. *Front. Publ. Health* **8** (2021)
2. The novel coronavirus outbreak in Wuhan, China. <https://ghrp.biomedcentral.com>. Last accessed 14/7/2021
3. WHO, Coronavirus disease 2019 (COVID-19): situation report, 205 (2020)
4. T. Meenpal, A. Balakrishnan, A. Verma, Facial mask detection using semantic segmentation, in *2019 4th International Conference on Computing, Communications and Security (ICCCS)* (2019)
5. D. Matthias, C. Managwu, Face mask detection application and dataset preprint available. <https://doi.org/10.13140/RG.2.2.18493.59368> (2021)
6. M. Sandeler, A. Howard, M. Zhu, A. Zhmoginov, MobileNetV2: inverted residuals and linear bottlenecks, in *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2018)
7. M.M. Rahman, M.M. Manik, M.M. Islam, S. Mahmud, J.-H. Kim, An automated system to limit COVID-19 using facial mask detection in smart city network, in *2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS)* (2020)
8. J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: unified, real-time object detection, in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, pp. 779–788 (2016)
9. A. Das, M. Wasif Ansari, R. Basak, COVID-19 face mask detection using TensorFlow, Keras and OpenCV, in *2020 IEEE 17th India Council International Conference (INDICON)* (2020)
10. S.A. Sanjaya, S. AdiRakhmawan, Face mask detection using MobileNetV2 in the era of COVID-19 pandemic, in *2020 International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy (ICDABI)* (2020)
11. R. Jaiswal, A. Agarwal, R. Negi, Smart solution for reducing the COVID-19 risk using smart city technology. *IET Smart Cities* **2**, 82–88 (2020)
12. M.S. Islam, E. Haque Moon, M.A. Shaikat, M. Jahangir Alam, A novel approach to detect face mask using CNN, in *2020 3rd International Conference on Intelligent Sustainable Systems (ICISS)* (2020)
13. B. Isunuri, B. Venkateswarlu, J. Kakarla, S. Prakash, Face mask detection using MobileNet and global pooling block, India, in *2020 IEEE 4th Conference on Information and Communication Technology (CICT)* (2020)

14. C. Gupta, N.S. Gill, Coronamask: a face mask detector for real-time data. *Int. J. Adv. Trends Comput. Sci. Eng.* (2020)
15. M. Jiang, X. Fan, H. Yan, *Retinafacemask: A Face Mask Detector* (2020)
16. B. Batagelj, P. Peer, V. Štruc, S. Dobrisek, *How to Correctly Detect Face-Masks for COVID-19 from Visual Information*, *MDPI* (2021)
17. WHO, *Advice on the Use of Masks in the Context of COVID-19: Interim Guidance* (2020)

Hierarchical Language Modeling for Dense Video Captioning



Jaivik Dave and S. Padmavathi

Abstract The objective of video description or dense video captioning task is to generate a description of the video content. The task consists of identifying and describing distinct temporal segments called events. Existing methods utilize relative context to obtain better sentences. In this paper, we propose a hierarchical captioning model which follows encoder-decoder scheme and consists of two LSTMs for sentence generation. The visual and language information are encoded as context using bi-directional alteration of single-stream temporal action proposal network and is utilized in the next stage to produce coherent and contextually aware sentences. The proposed system is tested on ActivityNet captioning dataset and performed relatively better when compared with other existing approaches.

Keywords Video description · Dense video captioning · Computer vision · Natural language processing

1 Introduction

Videos have become an integral part of information interchanging on online internet platforms such as YouTube. On YouTube alone, five hundred hours of video data are being uploaded every minute, and over a billion hours of video data being watched every day. Handling of these profuse amounts of video requires generation of short textual description using automatic analysis of the videos. Video description generation is beneficial to applications involving video retrieval [1, 2], video understanding [3], video recommendation [4], video summarization [5], etc. It can also be used in surveillance and security field for identifying drones, objects or activities and

J. Dave (✉) · S. Padmavathi
Department of Computer Science and Engineering, Amrita School of Engineering, Amrita
Vishwa Vidyapeetham, Coimbatore, India
e-mail: cb.en.p2aid19007@cb.students.amrita.edu

S. Padmavathi
e-mail: s_padmavathi@cb.amrita.edu

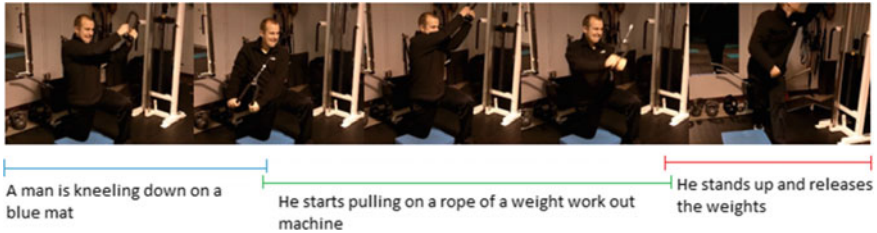


Fig. 1 Example of dense video captioning from ActivityNet captions [6]

report to the concerned authorities. This would also allow visually impaired people to absorb video content with ease.

Prior works in video description problem targeted generating a single caption for the whole video (video captioning). But, it is possible that a single sentence would not be sufficient to describe all the information in a video. Furthermore, longer and real-life videos generally contain numerous activities and objects which require a more detailed description. Hence, the aim of the dense video captioning problem is first to localize the unique activities of the video and automatically generate a natural language description of the same. Few sample frames and the relevant description are shown in Fig. 1. The problem comprises of several challenges like identifying unique and distinct events in the video, capturing the temporal and motion information of the objects and contents of the video and converting it into textual knowledge.

The task is usually formulated as a combination of three modules: encoding a video, temporal segmentation of the video, and finally generating sentences for the identified segments. We introduce an end-to-end approach that first encodes video frame sequence using spatio-temporal convolution neural networks. The system then predicts the possible event, based on past and future events using bi-directional alteration of single-stream temporal (SST) [7] action proposal. Finally, it uses a hierarchical scheme-based captioner to encode the context and decode it along with possible event features to produce a textual description. Hierarchical models have been used in natural language processing for successful sentence generation. In this paper, the hierarchical models are combined with SST action proposal for dense video captioning.

2 Related Works

Over the past decade, prolific research has been done on describing images and videos after the successful advances in natural language processing and computer vision. The early works started with image and video captioning problems to develop models that depict contents in images and videos in a single captioning sentence. Further several works proposed describing the videos in paragraphs to overcome the information loss in the video captioning methods. But even so, the distinct events in the videos were not

addressed or identified explicitly, lacking in providing accurate video descriptions. Nevertheless, as mentioned in section I, the dense video captioning is very beneficial to numerous video analytics task as well as a real-life application like tracking the attention level of students [8], analyzing underwater video [9], identifying animal and humans from surveillance camera [10, 11].

Early works in video captioning were based on template method or rule-based models (e.g., SVO [12], SVOP [13]). Their approaches predict the required content (Subject, Verb, Object, Place, etc.) and then put them into the template to produce a sentence. Modern works use deep learning models for better captioning results especially using recurrent neural network-based sentence generation (e.g., RNN, LSTM, and GRU). These approaches follow the encoder-decoder scheme where the encoder processes video features, and the decoder uses encoder output to produce sentences. Further improvement of video captioning results was proposed by using spatio-temporal attention [14], reinforcement learning [15], and paragraph generation instead of a single sentence [16, 17].

The video paragraph generation problem aims to overcome the limitations of video captioning by providing a detailed explanation of the video content in multiple sentences. Although it offers an elaborate explanation, it does not produce temporal segmentation of the video, which is addressed in the dense captioning task. In the paper [18], authors employed hierarchical RNN models to include sentence history in the decoding process but did not consider the visual context. The paper [19] also utilizes a hierarchical model to produce more coherent paragraphs. We also adopt the idea of incorporating sentence history along with a hierarchical model to get more coherency and meaning in the captions.

The dense video captioning task was introduced by [6] along with the ActivityNet captions dataset. They used Deep Action Proposals (DAPs) [20] for event localization and LSTM network to generate sentences by incorporating past and future context which fails to accommodate highly overlapping events due to usage of fixed strides. The paper [7] extended the idea of context-awareness by utilizing a bi-directional alteration of single-stream temporal (SST) action proposal [21] and employed an LSTM network with ‘context gating’ to generate contextual sentences. In [22], a hierarchical captioning module is utilized that uses controller LSTM to add context to sentence generator LSTM. The former produces better temporal segments and the later employs effective captioning model, but both lack in the other aspect of the system, i.e., sentence generation and accurate segment proposal, respectively. In view of the recent success of transformers in NLP tasks over RNN-based models, [23] employed a masked transformer-based captioning model to address the task in question.

Several approaches have attempted to include audio [24], speech or both [25] features of the video along with visual features for dense video captioning. The paper [24] proposed a multi-model approach that encodes the video’s audio features along with visual features and decoded them with a bi-modal transformer. In [25], importance of speech modality was showed along with video and audio to enhance the performance of the system. However, the event proposal only utilizes visual information and combination of the features is inefficient. The paper [26] captures

contextual relations between the events and stores into ‘common-sense’ knowledge base but building the database and fetching the vectors corresponding to particular events makes the training and process lengthy and computationally expensive. The paper [27] focus on pre-training strategies for multi-modal pipeline. In order to achieve superior results, some approaches [28] have employed multiple modules resulting in very complex pipeline architecture.

As discussed earlier, majority of the works faces limitations with event proposal or opt for complex approach for captioning. The goal of this work is to develop comparatively less complex and interpretable approach for dense video captioning task. This work focuses on processing the visual information with a 3D CNN; combining it with a bi-directional alteration of single-stream temporal (SST) for action proposal and uses hierarchical language model for sentence generation.

3 Proposed Architecture

This section discusses the architecture framework of the proposed method for the dense video captioning problem. Figure 2 shows an overview of the proposed framework, which comprises of three modules: video encoding, temporal video segmentation or event proposal, and captioning module. The system first encodes the given video to get feature stream and then identifies the unique events (temporal segment) using these feature stream. Finally, description for each event is generated as a natural language sentence.

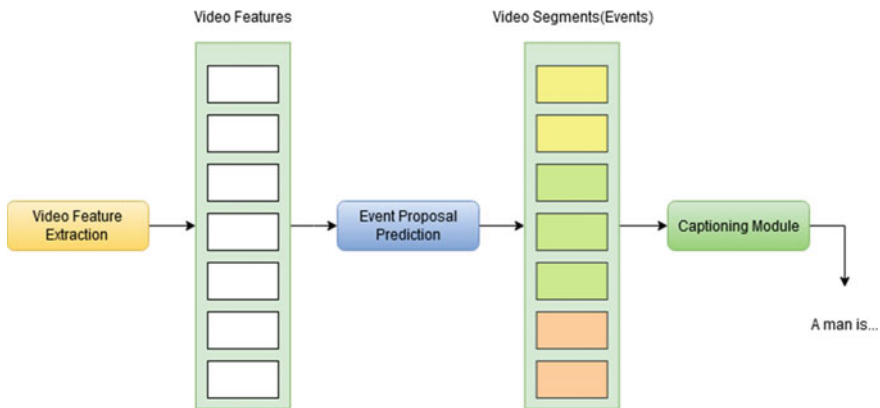


Fig. 2 Complete architecture of the proposed system

3.1 Video Feature Extraction

To obtain the sequence of features from the input video frames, we use the C3D [29] network, a three-dimensional convolutional network. The C3D model used is pre-trained on Sports-1 M [30] dataset and publically available with temporal resolution (δ) of 16 frames. For a video with n frames $v = \{v_1, v_2, v_3, \dots, v_n\}$, the network produces feature stream of length N where $N = n/\delta$. Principal component analysis (PCA) [31] is applied to reduce the dimensionality of the feature streams to $N \times L$, where L specifies the number of significant principal components. This module produces an output of feature stream $f = \{f_1, f_2, f_3, \dots, f_L\}$.

3.2 Event Proposal Prediction

The obtained feature stream is fed to the event proposal module which identifies possible temporal segments (event) of the video. We employ bi-directional alteration of single-stream temporal SST [7] which incorporates future events along with past events to predict better localization of the possible event. It encodes visual features using LSTM such that hidden state of the LSTM will contain the information till the current timestep t . These are processed to predict possible proposal events with certain scores. Next, the feature stream is encoded in the reverse order and processed similarly to get the proposal events with scores. The scores of the same predicted proposals are combined using an adapt combination strategy and proposal with scores higher than decided threshold are selected for further processing.

3.3 Captioning Module

The final module of the proposed architecture describes each identified event. The visual information of the video can be processed naively by handling each individual event separately and generating the corresponding caption. However, events are linked and can even influence or trigger one another. To model such situation successfully, we propose a hierarchical captioner as shown in Fig. 3. Here initially, the visual information along with language history is encoded. These are then combined with input proposal features for decoding. Input will be a short feature stream. It is necessary to modal the sequential information from these streams to generate accurate description. Since the video can contain any length of events (temporal segments), a simple recurrent neural network will not cater the need. Here we use LSTM cell to produce textual information. It can modal short-term as well as long-term dependencies in contrary to basic recurrent neural networks.

We model the context through encoder LSTM which is used later for decoding. The context is represented as visual information and language history. The context

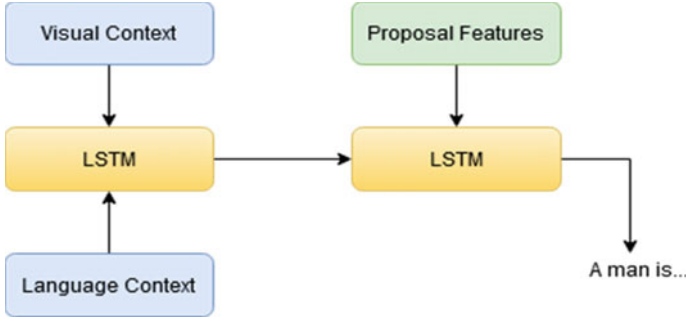


Fig. 3 Captioning module framework

involving past and future visual information; information of previously generated sentences in language history are encoded in hidden state of encoder LSTM (Eqs. 1–4). Different gate vectors of LSTM are computed using Eq. (1), Memory cells are computed using Eqs. (2) and (3), current hidden state of LSTM is computed using Eq. (4).

$$\begin{bmatrix} i_t \\ f_t \\ o_t \end{bmatrix} = \sigma \left(\begin{bmatrix} W_i \\ W_f \\ W_o \end{bmatrix} \begin{bmatrix} V_t \\ D_t \\ h_{t-1} \end{bmatrix} + \begin{bmatrix} b_i \\ b_f \\ b_o \end{bmatrix} \right) \quad (1)$$

$$\hat{c}_t = \tanh(W_c [V_t \ D_t \ h_{t-1}] + b_c) \quad (2)$$

$$c_t = i_t \cdot \hat{c}_t + f_t \cdot c_{t-1} \quad (3)$$

$$h_t = o_t \cdot \tanh(c_t) \quad (4)$$

where i_t , f_t , and o_t are gates of the LSTM input, forget, and output gates, respectively. V_t is visual information, D_t is past language history, and W and b are corresponding weights and biases. h_t and c_t represent hidden state and memory cell of the LSTM cell and \cdot represents element-wise multiplication.

Next, the decoder LSTM network processes hidden state h_t of the encoder LSTM along with event features to generate contextually aware coherent sentences by following equations (Eqs. 5–8):

$$\begin{bmatrix} i_{t,m} \\ f_{t,m} \\ o_{t,m} \end{bmatrix} = \sigma \left(\begin{bmatrix} W_i \\ W_f \\ W_o \end{bmatrix} \begin{bmatrix} h_{t,m}^{(1)} \\ V_t^p \\ h_{t,m-1}^{(2)} \end{bmatrix} + \begin{bmatrix} b_i \\ b_f \\ b_o \end{bmatrix} \right) \quad (5)$$

$$\hat{c}_{t,m} = \tanh(W_c [h_{t,m}^{(1)} \ V_t^p \ h_{t,m-1}^{(2)}] + b_c) \quad (6)$$

$$c_{t,m} = i_{t,m} \cdot \widehat{c_{t,m}} + f_{t,m} \cdot c_{t,m-1} \quad (7)$$

$$h_{t,m}^{(2)} = o_{t,m} \cdot \tanh(c_{t,m}) \quad (8)$$

where $i_{t,m}$, $f_{t,m}$, and $o_{t,m}$ are input, forget, and output gates of the decoder LSTM, respectively. V_t^p is visual information of proposal features, $h_{t,m}^{(1)}$ is the hidden state of encoder LSTM and context vector, W and b are corresponding weights and biases. $h_{t,m}^{(2)}$ and $c_{t,m}$ are represents hidden state and memory cell of the decoder LSTM cell and represents element-wise multiplication. The next m th word is predicted based on the hidden state $h_{t,m}^{(2)}$.

The whole model is trained in an end-to-end fashion with a learning rate of 0.0001 with Adam optimizer. We combine loss from the event proposal module and sentence generation module to get the total loss of overall architecture. Loss of event proposal module (\mathcal{L}_p) is calculated based on weighted cross-entropy, and only the proposal having higher IoU (Intersect-over-Union) than ground truths are sent to the language model. Analogous to prior work in language models, we compute language model loss (\mathcal{L}_c) as a sum of the negative log-likelihood of the right words in the sentences.

$$\mathcal{L}_T = \lambda_p \mathcal{L}_p + \lambda_c \mathcal{L}_c \quad (9)$$

where \mathcal{L}_T is total loss computed as a function of \mathcal{L}_p and \mathcal{L}_c weighted by λ_p and λ_c deciding contribution of each loss which are set to 0.5 and 1, respectively.

4 Experiments

4.1 Dataset

ActivityNet Captions dataset is based on ActivityNet [32] dataset and was introduced by [6] along with the dense video captioning task. The dataset contains videos annotated with temporal segments and sentences corresponding to each of the segments in the video. Thus, dataset links each unique event in the video with corresponding description. The temporal video segments do not have any constraints in terms of length and can occur simultaneously and overlap with each other. The dataset contains 20 k untrimmed YouTube videos and 100 k sentences. On average, each video is about two minutes long and has four events. Each sentence contains around 13 words on average. The videos are split into 50/25/25 percentage of training, validation, and testing, respectively.

4.2 Results and Discussion

To assess the performance of the proposed architecture, we use METEOR [33] score to determine the degree of similarity between the phrases. METEOR score has been shown to be closely associated and consistent with human assessments, especially when number of reference sentences are limited. However, slight incongruency in other available evaluation metrics (Bleu [34], CIDEr [35]) is noticed by [6, 7, 36], which is due to word sequence misalignment. Also, both the scores are designed based on correlation at corpus level especially Bleu, whereas METEOR is based on correlation with human judgements which is desired for this problem. Hence, the METEOR score is selected for the performance comparison.

The performance comparison of the proposed architecture with existing approaches on the ActivityNet captions dataset is shown in Table 1. The first method [6] introduced the dense video captioning problem and the ActivityNet dataset. It predicts proposals in a single pass with contextually aware sentences. But the technique used predefined strides for action proposal, thus limiting the model for event detection. JEDDI-Net [22] employed controller LSTM to encode language and visual context to enhance the performance, whereas [7] utilized a bi-directional encoder for proposal prediction to identify the endpoints of events more precisely. The proposed method takes language context into consideration which is not the case for [6, 7]. While [8] modals the visual and language history, the event proposal module cannot match the performance of Bi-SST employed in proposed method. For [7], we consider our implementation of variant without ranking proposed in the paper. As it can be seen, our method performs well in comparison to the baseline approaches taken into consideration.

For qualitative analysis, we show a sample example of dense captioning. Figure 4 shows the screenshots of video frames and sentences for the first three events. Even though architecture is able to identify details from the video like ‘snowboard,’ ‘flip,’ ‘slope,’ etc., it still fails to generate more in depth description provided in ground truth.

Table 1 Performance comparison with other existing methods

Method	Meteor
DAPs + LSTM [6]	4.82
JEDDI-Net [20]	8.58
Bi-SST + LSTM [7]	9.17
Ours	9.25

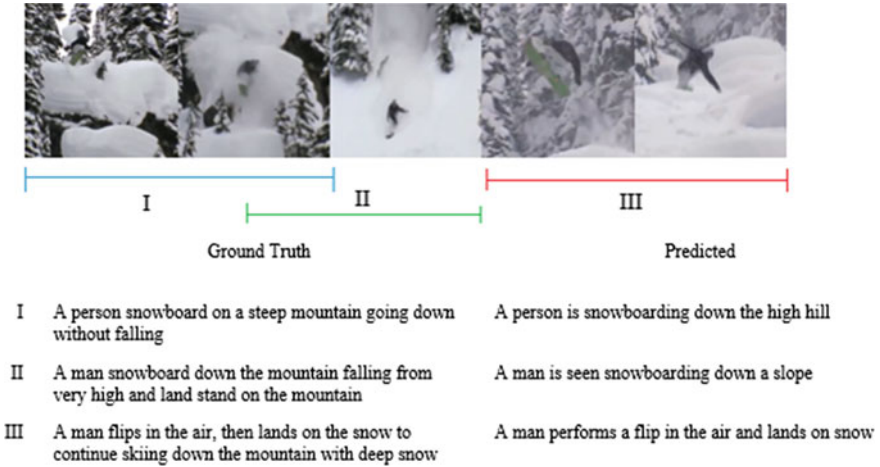


Fig. 4 Qualitative result on dense video captioning task

5 Conclusions

The dense video captioning problem was addressed in this paper, and an end-to-end pipeline, architecture was presented. The proposed architecture employs a bi-directional action proposal module with hierarchical captioning module which incorporates language and visual context into the decoding process. The architecture can produce accurate action proposal along with more coherent and consistent sentences throughout the video, and it performs well compared to existing methods. Transformers have been shown to outperform recurrent neural networks in a variety of natural language processing tasks; therefore, they could be employed as a sentence generator for the entire pipeline in the future.

References

1. V. Gabeur, C. Sun, K. Alahari, C. Schmid, Multi-modal transformer for video retrieval, in *European Conference on Computer Vision (ECCV)*, pp. 214–229 (2020)
2. R. Dhaya, Analysis of adaptive image retrieval by transition Kalman filter approach based on intensity parameter. *J. Innov. Image Process.* **3**(1), 7–20 (2021)
3. G. Bertasius, H. Wang, L. Torresani, Is space-time attention all you need for video understanding? [arXiv:2102.05095](https://arxiv.org/abs/2102.05095) (2021)
4. D. Yao, S. Zhang, Z. Zhao, W. Fan, J. Zhu, X. He, F. Wu, Modeling high-order interactions across multi-interests for micro-video recommendation. *AAAI* (2021)
5. T. Hussain, K. Muhammad, W. Ding, J. Lloret, S.W. Baik, V.H.C. de Albuquerque, A comprehensive survey of multi-view summarization. *Pattern Recogn.* (2020)
6. R. Krishna, K. Hata, F. Ren, L. Fei-Fei, J.C. Niebles, Dense-captioning events in videos, in *International Conference on Computer Vision* (2017)

7. J. Wang, W. Jiang, L. Ma, W. Liu, Y. Xu, Bidirectional attentive fusion with context gating for dense video captioning, in *Conference on Computer Vision and Pattern Recognition* (2018)
8. N. Krishnnan, S. Ahmed, T. Ganta, G. Jeyakumar, A video analytic based solution for detecting the attention level of the students in class rooms, in *International Conference on Cloud Computing, Data Science Engineering* (2020)
9. D.G. Lakshmi, K.R. Krishnan, Analyzing underwater videos for fish detection, counting and classification, in *International Conference on Computational Vision and Bio Inspired Computing* (2019)
10. S. Ravikumar, D. Vinod, G. Ramesh, S.R. Pulari, S. Mathi, A layered approach to detect elephants in live surveillance video streams using convolution neural networks. *J. Intell. Fuzzy Syst.* **38**, 6291–6298 (2020)
11. K. Mondal, S. Padmavathi, Wild animal detection and recognition from aerial videos using computer vision technique. *Int. J. Emerging Trends Eng. Res.* **7**(5), 21–24 (2019)
12. A. Barbu, A. Bridge, Z. Burchill, D. Coroian, S. Dickinson, S. Fidler, A. Michaux, S. Mussman, S. Narayanswamy, D. Salvi, L. Schmidt, J. Shangguan, J.M. Siskind, J. Waggoner, S. Wang, J. Wei, Y. Yin, Z. Zhang, Video in sentences out, in *Conference on Uncertainty in Artificial Intelligence* (2012)
13. J. Thomsan, S. Venugopalan, S. Guadarrama, K. Saenko, R. Mooney, Integrating language and vision to generate natural language descriptions of videos in the wild, in *International Conference on Computational Linguistics* (2014)
14. C. Yan, Y. Tu, X. Wang, Y. Zhang, X. Hao, Y. Zhang, Q. Dai, STAT: spatial-temporal attention mechanism for video captioning. *Trans. Multimedia* (2020)
15. X. Wang, W. Chen, J. Wu, Y. Wang, W.Y. Wang, Video captioning via hierarchical reinforcement learning, in *Conference on Computer Vision and Pattern Recognition* (2018)
16. A. Rohrbach, M. Rohrbach, W. Qiu, A. Friedrich, M. Pinkal, B. Schiele, Coherent multi-sentence video description with variable level of detail, in *German Conference on Pattern Recognition* (2014)
17. H. Yu, J. Wang, Z. Huang, Y. Yang, W. Xu, Video paragraph captioning using hierarchical recurrent neural networks, in *Conference on Computer Vision and Pattern Recognition* (2016)
18. H. Yu, J. Wang, Z. Huang, Y. Yang, W. Xu, Video paragraph captioning using hierarchical recurrent neural networks, in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4584–4593 (2016)
19. R. Lin, S. Liu, M. Yang, M. Li, M. Zhou, S. Li, Hierarchical recurrent neural network for document modeling, in *Conference on Empirical Methods in Natural Language Processing*, pp. 899–907 (2015)
20. V. Escorcia, F.C. Heilbron, J.C. Niebles, B. Ghanem, DAPs: deep action proposals for action understanding, in *European Conference on Computer Vision*, pp. 768–784 (2016)
21. S. Buch, V. Escorcia, C. Shen, B. Ghanem, J.C. Niebles, SST: single stream temporal action proposals, in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2911–2920 (2017)
22. H. Xu, B. Li, V. Ramanishka, L. Sigal, K. Saenko, *Joint Event Detection and Description in Continuous Video Streams* (IEEE, 2018)
23. L. Zhou, Y. Zhou, J.J. Corso, R. Socher, C. Xiong, End-to-end dense video captioning with masked transformer, in *Conference on Computer Vision and Pattern Recognition* (2018)
24. V. Iashin, A better use of audio-visual cues: dense video captioning with bi-modal transformer, in *British Machine Vision Conference* (2020)
25. V. Iashin, E. Rahtu, Multi-modal dense video captioning, in *Conference of Computer Vision and Pattern Recognition*, pp. 958–959 (2020)
26. A. Chadha, G. Arora, N. Kaloty, iPerceive: Applying common-sense reasoning to multi-modal dense video captioning and video question answering. *WACV* (2020)
27. G. Huang, B. Pang, Z. Zhu, C. Rivera, R. Soricut, Multimodal pretraining for dense video captioning, in *Proceeding of 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and 10th International Joint Conference on Natural Language Processing*, Association for Computational Linguistics, pp. 470–490. Suzhou, China (2020)

28. T. Wang, H. Zheng, M. Yu, *Dense Captioning Events in Videos: SYSU Submission to ActivityNet Challenge 2020*. [arXiv:2006.11693](https://arxiv.org/abs/2006.11693) (2020)
29. D. Tran, L. Bourdev, R. Fergus, T. Torresani, M. Paluri, Learning spatiotemporal features with 3D convolutional networks, in *International Conference on Computer Vision* (2015)
30. A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, L. Fei-Fei, Large-scale video classification with convolutional neural networks, in *Conference on Computer Vision and Pattern Recognition* (2014)
31. I.T. Jolliffe, Principal component analysis, *Springer Series in Statistics* (2002)
32. F.C. Heilbron, V. Escorcia, B. Ghanem, J.C. Niebles, Activitynet: A large-scale video benchmark for human activity understanding, in *Conference on Computer Vision and Pattern Recognition* (2015)
33. S. Banerjee, A. Lavie, METEOR: An automatic metric for MT evaluation with improved correlation with human judgements, in *Intrinsic and Extrinsic Evaluation Measures for MT and/or Summarization* (2005)
34. K. Papineni, S. Roukos, T. Ward, W.J. Zhu, Blue: a method for automatic evaluation of machine translation. *ACL* (2002)
35. R. Vedantam, C.L. Zitnik, D. Parikh, Cider: Consensus-based image description evaluation, in *Conference on Computer Vision and Pattern Recognition (CVPR)* (2015)
36. L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, A. Courville, Describing videos by exploiting temporal structure. *ICCV* (2015)

Resource Provisioning in Fog-Based IoT



Daneshwari I. Hatti and Ashok V. Sutagundar

Abstract The devices in the Internet of Things (IoT) communicate through the Internet without human intervention. An enormous number of devices and their generated data leads to several challenges such as data processing at appropriate devices, resource discovery, mapping, and provisioning. The proposed work addresses the management of the workload of devices by offering resources through the fog computing paradigm with less cost and energy consumption. Distributed provision solves the problem of multiple requests having similar response time requirements. It categorizes such requests into different swarms and provides the resources through various fog devices existing in several fog colonies. Each swarm gets mapped to one or more fog colonies considering response time, total resource capacity, and distance between them. Fitness value for all the tasks in a swarm is calculated for binding to fog colony using Multi-Objective Particle Swarm Optimization (MOPSO). In each swarm, the existing requests are mapped to suitable fog devices for processing and avoid overloading and under-provision of fog devices. The performance of the proposed model is evaluated in the CloudSim-Plus framework by the varying capacity of fog instances in terms of small, medium, high, and mixed resources set, tasks/cloudlet length, and response time of requests.

Keywords Clustering · Swarm · Fitness function · Optimization · PSO · Resource provision · CloudSim-plus · Cost · Energy consumption · And resource utilization

1 Introduction

Internet of Things (IoT) is paradigm that helps to communicate among device through Internet and serves [1–3] many applications in today’s world with increase in demand.

D. I. Hatti (✉)

Department of Electronics and Communication, BEC, Bagalkot, BLDEA’s V.P. Dr. P.G.H. College of Engineering and Technology, Vijayapur, Karnataka, India

A. V. Sutagundar

Department of Electronics and Communication, Basaveshwar Engineering College, Bagalkot, Karnataka, India

The sensed data is preprocessed at the cloud layer due to constraint resources at the fog devices. The performance of a system with more devices decreases, causing communication delay and data traffic. With the optimal resource provisioning strategy [3], cloud resources are pooled and reassigned to meet user demand. The applications are of soft real time and hard real time that are sensitive to latency. Hence, offloading the tasks to cloud is overcome by employing fog computing [4].

Fog computing [5] acts as an intermediate layer for computation, processing, provision the resources [6] and deciding the action at the edge. Management of fog devices resources at fog layer has become difficult due to resource constraints and the unpredictable behavior of fog devices. Prior to allocation, resource discovery, mapping, and provision is conducted to ensure that computing and communication resources are used effectively.

The number of IoT devices is growing every day, and these devices are time-sensitive and demands immediate responses. These devices connected to a cloud environment for processing are delayed for computation. To address this issue, the fog environment is retained at the IoT device layer's edge to perform fast computations for delay-sensitive applications. Due to heterogeneity of fog devices behavior and workload the choosing of appropriate set of resources is challenging and hence motivated to address this problem. The heterogeneity at the device layer, and fog layer is overcome, and provision is performed through swarm intelligence. After getting the perfect match between fog device and workload still, the resources of devices remain without utilization because of different demand. Hence, to avoid the issue of underutilization, bestfit algorithm is applied.

In this work, authors have focused on optimizing the throughput, resource utilization, and minimizing cost and energy consumption. The heterogeneous request handling and scheduling of requests, provision of resources from the resource pool are major problems solved by the proposed algorithm. The request demanding various configured resources requires provisioning for reducing latency and ensuring proper utilization of resources. The work mainly concentrates on the type of requests, classification, and provision with appropriate fog devices considering the task's response time, bandwidth, and CPU. Based on response time and requirement of the tasks, the fitness function is estimated to search for appropriate fog devices for allocating the tasks. The best matching between resource provider and user with requirement ensures proper provisioning by avoiding overloading and underloading of fog devices.

The objective of this work is to apply intelligence in discovering fog colony for each swarm and then in each swarm to search appropriate fog device in a selected fog colony through MOPSO. Firstly, provision the resources from fog device to all type of workload (short and huge length) with reduced waiting time. This is simulated by configuring fog colonies with varied and same capacity and by clustering requests with heterogeneous type. Provision is done for multiple swarms by multiple fog colony ensuring parallelism in processing the requests. The under and over provisioning is avoided by applying proposed modified best fit algorithm. The fog residing between IoT and cloud reduces the traffic towards cloud and manages the requests from IoT devices with reduced cost and latency.

The organization of this paper is as follows. Section 2 discusses the proposed model. In Sect. 3, results are discussed. Section 4 concludes the paper.

1.1 *Related Work*

Due to billions of devices/things, managing with fewer service failures is difficult due to the billions of devices/things [7] linked across the network. Machines must have intelligence comparable to human intelligence in order to manage finite resources. Because of the similarities between the behavior of devices/things in an IoT network and the behavior of particles in a swarm, SI's working concept has been applied to a variety of IoT scenarios. Several previous studies looked into agent-based provisioning [8], scheduling the clustered devices [9], authentication, and SI-based provision taking into account of cost and other considerations [10]. It is critical to manage resources for the fulfillment of user requests, while adhering to the service level agreement (SLA) [11].

In [12], cloud provider maximized the profit without exceeding the cloud provider's upper bound of energy usage. The goal of SaaS customers is to get the best possible QoS in order to complete their tasks within a set budget and time frame. In [13], authors have proposed optimization problem by taking into account of fog/IoT resources and reduced delay. Authors in [14] QoS aware resource provisioning in cloud computing is performed for analyzing the cloud workloads with reduction in execution time and execution cost. Authors [15] have addressed provisioning scheme for cloud data centers having heterogeneity workload and machines of different energy consumption. They have achieved proper matching between workload and virtual machine with reduced delay in cloud computing. In [16] fog resource provisioning is addressed with a tradeoff between reliability and the system cost. In [17] QoS aware fog resource provisioning is done to minimize system cost considering power of IoT devices. In [18], authors proposed distributed resource provision to minimize the latency of different IoT applications. They have placed IoT applications at edge devices and parts of the application are handled without violating service level agreement (SLA). In [19], authors proposed scheduling mechanism to select the most adequate host to achieve the minimum response time for a given IoT service using fog computing. In [20], authors have used multi-agents for allocating resources with reduced response time and makespan time. The workload is clustered for allocating the resources. The work addressed by several authors has minimized the latency, cost, and energy consumption. The modification over the existing method is performed by using swarm intelligence to handle heterogeneity demand and fog devices.

Swarm is a single network made up of populations such as bees, ants, birds, and other insects that work together for decision making is correlated to the nature of devices in an IoT layer. Each swarm constituting varied characteristics are mapped to fog colonies, and in each fog colony, the fog devices meeting the resource required is discovered for provisioning. Resources are provisioned based on the decision of

Particle Swarm Optimization (PSO) [21, 22] to achieve better QoS. Provisioning may be of reactive, proactive [23], and hybrid.

Proactive methods pertain to workload prediction and provisioning, whereas reactive mechanisms entail providing of resources depending on current workload.

The following are some of the current limitations in resource management in IoTs: intelligence in preprocessing activities, distributed in resource provision, frequent resource update, and scalability of resources to meet the demands of multiple swarms.

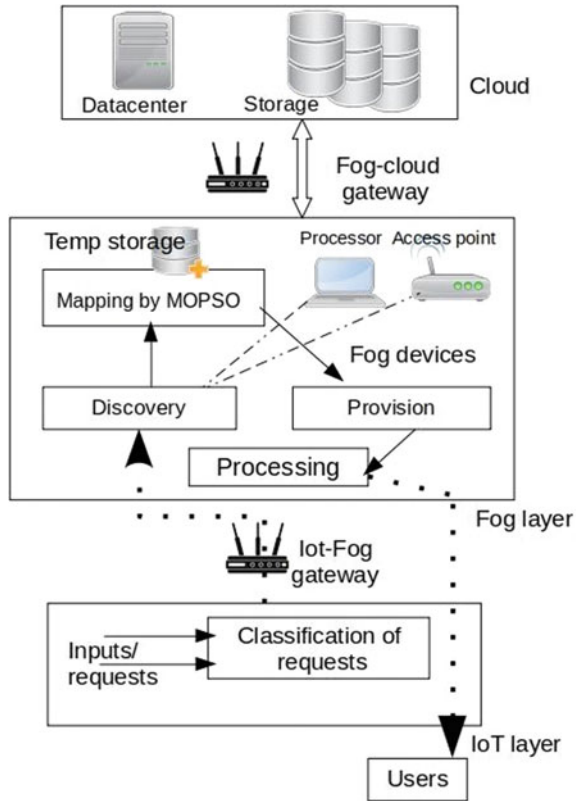
2 Proposed Work

Resource provisioning enables users to access particular resources based on their needs. Provision may be reactive, proactive, and hybrid modes based on the type of applications. In this work, resource provisioning in reactive way is employed for provisioning specific resources for existing and newly arrived tasks. The discovering and mapping of resources to the existing workload is done by MOPSO. The swarms are created initially at the device level and fog level. Fog computing leverages the deployment of IoT devices close to the user's location minimizes latency. The SI method is used to manage resources in a distributed manner, such as processing numerous swarms and fog colonies while paralleling resource provisioning depending on cost, energy consumption, resource use, and response time.

2.1 Network Scenario

Network model is illustrated in Fig. 1. It constitutes of three layers. At the bottom IoT or edge devices exists and in between cloud and IoT, the fog devices in a fog colony exists. Figure 1 shows the proposed model in which resource provision through fog devices is shown. The requests of different response time are clustered to form a swarm at IoT layer. To manage the heterogeneous resources at fog layer, the provisioning is optimized by discovering, mapping and then allocating. The requests are processed by fog resources at fog layer and then communicated back to users. Fog colony is framed possessing different CPU, BW, and storage capacity. The process of discovering a perfect fog instances in fog layer increases the time for provisioning and processing; hence to overcome this problem before arrival of requests, the fog instances are discovered and clustered as *vmsmall*, *vmmmed*, *vmhigh*, and *vmmix*. After discovering, the perfect match for the workload or request is performed by evaluating fitness value between request and fog device. The fitness value is obtained after searching in each fog colony. The fog colony is a search space for searching the appropriate fog devices existing in that area. After finding perfect match, the fog device is provisioned to request in each swarm resulting in less energy consumption and cost. The cost is directly related to the usage of resources at the devices. The

Fig. 1 Proposed model



process is iterated at fog layer till the optimal solution is obtained. According to the matching factor, the fog devices are provided and allocated for processing the request. At the scheduled interval, the periodic update of resource availability at fog colony is done to avoid under and overprovisioning of fog devices. If the fog device is less utilized, then the new incoming request is allocated. The matching criteria of fog device and request is done before allocating the request. This enhances the usage of resources, execution of more requests and reduces the cost. If the resources are not available at fog layer, it is forwarded to cloud and then after processing it is reverted back to user.

The fog devices are grouped together in a fog colony, and the requests with different response time are clustered [24] in to a swarm. The number of fog devices is varied from 1 to 10, and the capacity is in terms of CPU (MIPS), Memory (Mb), and Bandwidth. The fog devices with small capacity are clustered and named as vmsmall, fog devices having medium capacity is vmmed, high capacity fog devices is vmhigh, and fog devices of small, medium, high capacity is vmmix fog colony. The heterogeneous capacity of fog instances in a fog colony (vmmix) are mapped to swarm. Each requests are provisioned with best and nearest fog instance of a fog

colony. MOPSO is applied to discover fog resources and estimate the best fitness value resulting in appropriate match between requests and fog device. If the same priority requests approaches the same fog instance, then it is tedious to decide for which request the resources have to be provisioned; hence, this problem is overcome by applying proposed model for each swarm. The mechanism is shown in Fig. 2. The local pbest is estimated for every iteration between swarm_i and FC_i. The global best is selected among the obtained local pbest and ids of FC, and swarm having optimal gbest is chosen for provisioning.

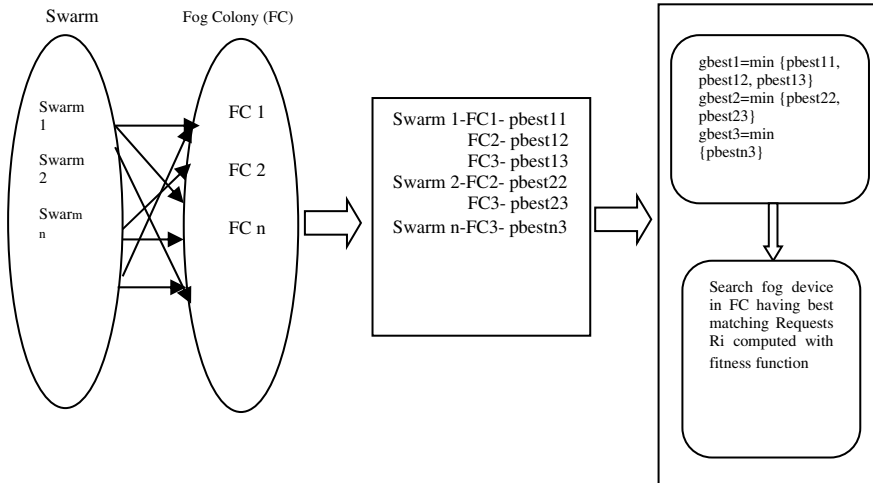


Fig. 2 Stages involved in provisioning

2.2 Algorithm—Proposed Work

Input: requests from devices $R = \{r_1, r_2, \dots, r_n\}$; Swarms $S_i = \{S_1, S_2, \dots, S_n\}$;
 Resour_request = $\{r_{MI}, r_{BW}, r_{RAM}\}$; Fog devices in a colonies $F_i = \{F_1, F_2, \dots, F_n\}$; Fog_dev_i = $\{f_1, f_2, \dots, f_i\}$;
 Fog device (f₁) = $\{MIPS, BW, RAM\}$
Output: p_best, g_best, fitness value

Begin

Step 1: Creation of cloudlets(requests) from the devices with heterogeneous capacity and requirement deployed in area of $x \times x \text{ m}^2$.

Step 2: Formation of swarms and fog colony as $\{S_i, F_i\}$

Initialize cluster head

Explore the search space and cluster in to fog colony

Step 3: For each Swarms(S_i), i 1 to n

Find distance using equation 7 for every S_i to F_i and check

$\text{Res_req_}S_i \leq \text{Res_ava_fog_col}$

Step 4: Matching the requests with fog devices is done using equations 3, 4, 5 and 6

Step 5: Fetch the id of swarm and fog colony

Step 6: Create a matrix

matr_ S_i _ F_i = {Swarms as rows and Fog colonies as Columns}

Step 7: for each element in matr_ S_i _ F_i i 1 to n

Step 8: for each element in matr_ S_i _ S_i j 1 to n

Step 9: If resour_requi < res_avaibility

Step 10: for S_i { T_i } i 1 to n

Find p_best_i using equation 8 for all fog devices

Find global best as g_best =

minimum{p_best_1, p_best_2, ..., p_best_n}

if T_i _res < ava_res(F_i)

Allocate new requests T_i with available resources at F_i

end

end for

end if

end for

end for

End for

Step 11: Find the fitness value by applying MOPSO and map each fog instances to request of devices

Step 12: Evaluate energy consumption

Step 13: Set the costperbw, costperproc, costperram

Step 14: Get resource utilization summary of each set of cloudlets, EC, RC and ET.

END

The pbest is calculated by the particle velocity and position using Eqs. 1 and 2 [12].

$$Pp^{i+1} = Pp^i + vi^{t+1}; \forall Pp^i \in R\{\min, \max\} \quad (1)$$

where Pp^{i+1} is a new position obtained by Eq. 1 using current position and its velocity vi^{t+1} at $t + 1$ looking at the particle in range of a particular swarm.

$$vi_j^{t+1} = vi_j^t + a_1r_{1j}^t(p_best_i^t - Pp_{ij}^t) + a_2r_{2j}^t(g_best - Pp_{ij}^t) \quad (2)$$

where

vi_j^{t+1}	new velocity of each particles in j th dimension at time $t + 1$.
vi_j^t	current velocity in j th dimension at current time t .
$a_1r_{1j}^t, a_2r_{2j}^t$	product of acceleration coefficients of particle and random distribution in range [0 1].
$p_best_i^t$	particle best output.
Pp_{ij}^t	position of particle at time 't'.
g_best	global best output of a swarm

$$R_reqi = \{\alpha, \beta, \gamma\} \quad (3)$$

where $R_reqi = \{\text{response time, Cloudlet length, Bandwidth}\}$ for every input request of a device in a swarm.

$$\text{fogdev}_i = \{\delta, \varepsilon, \lambda\} \quad (4)$$

$\text{Fogdev}_i = \{\text{processing time, CPU, Bandwidth}\}$ of fog devices in each fog colony.

$$\delta = \frac{\beta}{\varepsilon} \quad (5)$$

Processing time = R_reqi (MI)/ fogdev_i (MIPS)

$$\text{laten} = \delta + \text{dist}_{ij} \quad (6)$$

'laten' is delay calculated by summing the processing time and propagation delay which is calculated by applying Euclidean distance measure represented in Eq. 7.

$$\text{dist}_{ij} = \sqrt{(fd_ix - r_ix)^2 + (fd_iy - r_iy)^2} \quad (7)$$

dist_{ij} is actual distance between fog colonies and swarms, (fd_ix, fd_iy) is coordinates of i th fog device and (r_ix, r_iy) is coordinates of requesting i th devices in j th dimension.

The fitness function shown in Eq. 8 computes best match between fog device and swarm considering the requirement and type of request. Multi parameters are used in fitness function to optimize resource usage and reduce energy consumption, resource cost.

$$\text{Map_fitfun} = X_{\text{res}} * \left[\frac{\text{laten}}{\alpha} \right] + Y_{\text{proc}} * \left[\frac{\beta/\alpha}{\varepsilon} \right] + Z_{\text{bw}} * \left[\frac{\gamma}{\lambda} \right] \quad (8)$$

‘X_res’, ‘Y_proc’ and ‘Z_bw’ are the assigned weights based on the priority of the tasks. ‘X_res’ value is set to 0.6 for tasks prioritized based on response time, ‘Y_proc’ for the processing factor and is set to 0.2 and ‘Z_bw’ decides bandwidth and is set to 0.2. Based on the type of request the weights are changed.

The proposed algorithm is simulated using Cloudsim plus for different configuration of fog colony and by changing the workload.

3 Results and Discussion

3.1 Simulation

The proposed model is simulated using CloudSim Plus [25]. It is a simulation framework in which the cloudlets and virtual machines are created dynamically during runtime. It is simulated for various characteristics and evaluated with performance parameters such as energy consumption, cost, and resource utilization.

Simulation steps

Begin
1. Create the proposed network environment in Cloudsim-plus
2. Configure, set parameters and inputs according to the specification shown in Tables 1 and 2
3. Cluster the requests and fog instances
4. Apply the MOPSO method
5. Compute and analyse the performance parameters of the network
End

Table 1 Characteristics of fog layer

Parameter	Fog	Cloud
Fog devices	4–15	4
MIPS of CPU	450–2600	5000–11,000
BW cost per unit (\$)	0.04	1.0
Processing cost per unit (\$)	0.1	1.0

Table 2 Cloudlet characteristics

Number of cloudlets/requests	30–120
MI of cloudlets	400–1100
Memory	256–1024 MB
Input and output file size	600 MB

The simulation is done using Cloudsim Plus, and the editors are Eclipse, PyCharm. Tables 1 and 2 show the parameters used for simulating the proposed algorithm.

3.2 Results

The performance of the system is evaluated for different characteristics. Figure 3 displays the number of cloudlets that have been executed versus the number of cloudlets that have been submitted. Fog colonies in this area have small, medium, high, and mix-capacity fog machines. This is the first iteration’s outcome. Cloudlet execution varies by fog colony capacity: ‘small, medium, large, and mix.’ The graph demonstrates that heterogeneous fog colonies outperform others. The execution of cloudlets is high in mix fog colony compared to others as it processes all types of requests. The other configuration is suitable for the requests requiring same set of resources from the pool. It is not suitable for heterogeneous requests because it does not have sufficient resources to process.

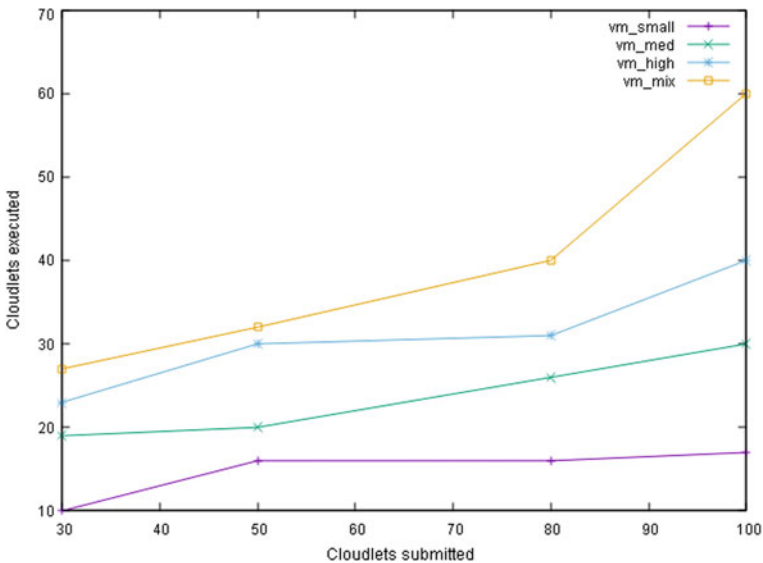


Fig. 3 Cloudlet submitted versus no. of executed

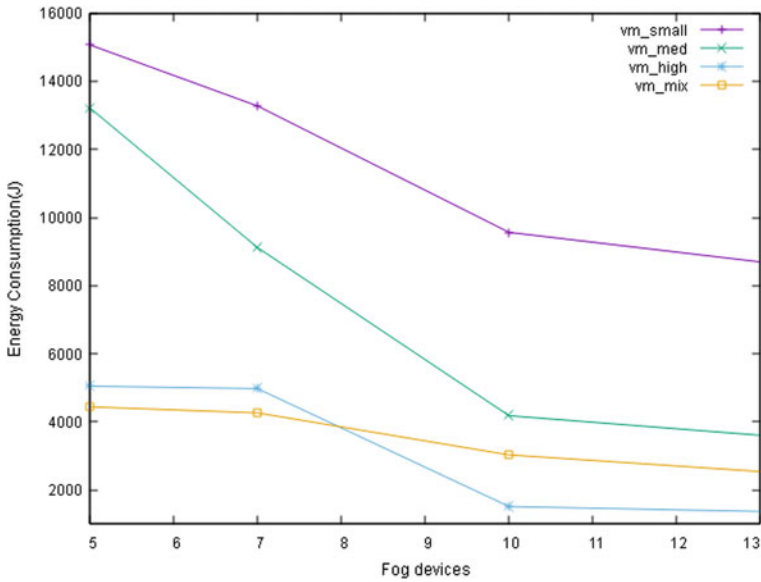


Fig. 4 Fog devices v/s energy consumption with 30 cloudlets/tasks

Figure 4 represents the energy consumption (EC) in Joules obtained by varying fog devices with incoming cloudlets of 30 and cloudlet length equal to 1000MI. Fog layer has 5, 7, 10, and 15 fog devices, and for simulation, fog colony is set to 3. The fog colony possess devices of diverse resource capacity, few colony are having small, medium and high capacity fog devices. When compared to fog colonies with different capabilities, fog colonies with diversified resources provide the resources with reduced energy usage. Few requests that are not handled by fog colonies are forwarded to the cloud, resulting in significant energy use.

Figure 5 depicts the resource cost of various fog instances in fog colonies v/s their capabilities as vmsmall, medium, high, and mix (heterogeneous). Because of optimal utilization of resources, the cost attained in vmmix is lower than other types of fog devices in fog colony.

Figure 6 depicts the variation in resource utilization of fog devices in first iteration for 50 heterogeneous requests. The utilization of each fog colony is obtained with 70, 85 and 93% for 3, 6, 14 devices in each fog colony. Utilization of fog devices is varying due to the characteristic of incoming request and number of devices. In a swarm of heterogeneous requests is mapped with last available pool of resources because there is no option of choosing the best solution. Hence, the available resources are allocated to newly arrived tasks to increase the resource utilization. Figure 6 shows the available resources at each fog device that can be used for further provisioning. Based on this, the best fit algorithm is applied for allocating the requests to fog device having high resources. The fog devices are uniformly distributed among requests based on fitness function avoiding the overprovisioning and under provisioning of fog resources.

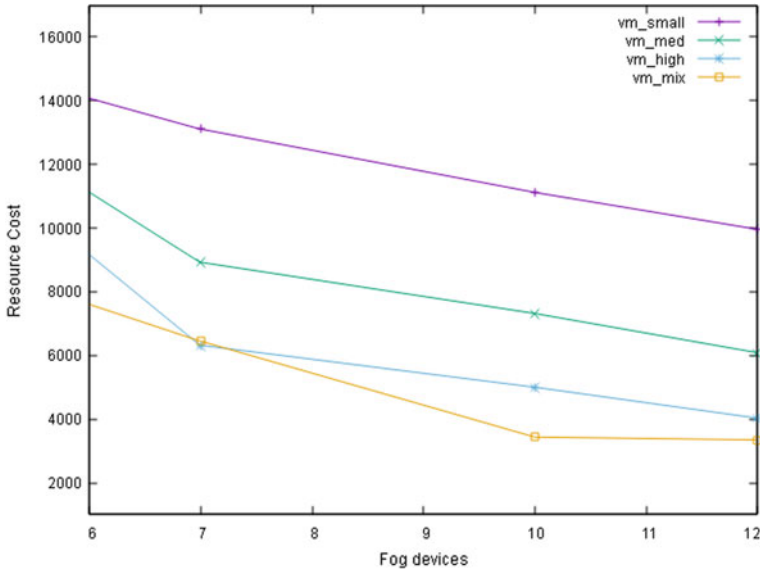


Fig. 5 Fog devices versus resource cost

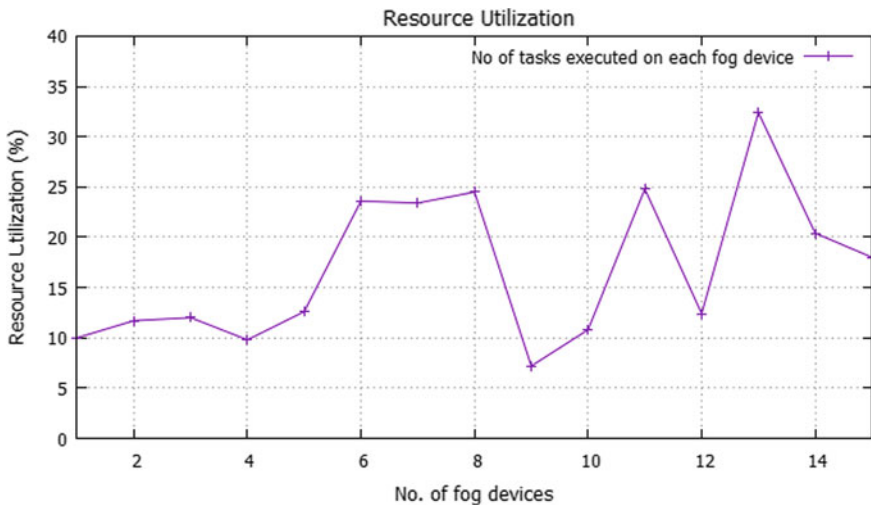


Fig. 6 No. of fog devices versus resource utilization

The fog devices are uniformly distributed among requests based on fitness function avoiding the overprovisioning and under provisioning of fog resources. Figure 7 shows the results obtained for proposed algorithm and bestfit, firstfit algorithms. The proposed algorithm performs better than bestfit, first fit algorithms in terms of energy consumption, execution of cloudlets (requests), execution time, and cost.

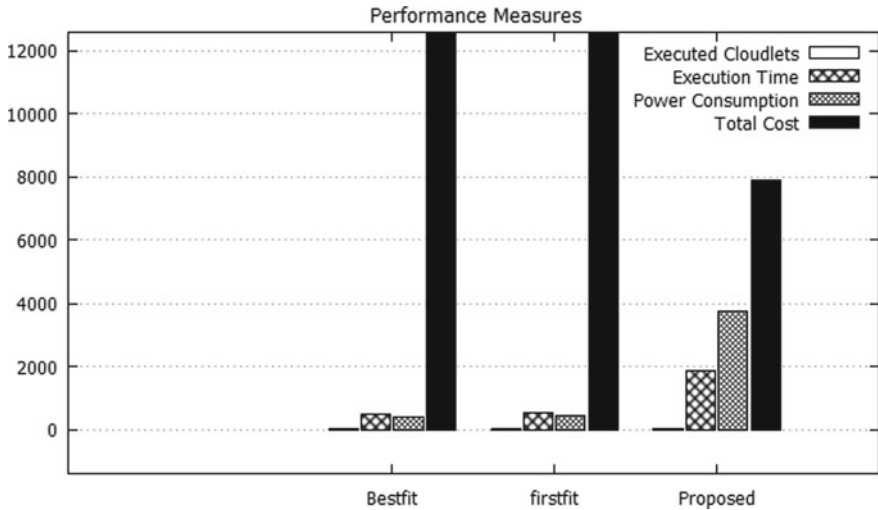


Fig. 7 Comparison of proposed work with existing algorithm

4 Conclusion

The proposed work addressed the challenge in a centralized strategy for provisioning the fog resources. Due to growing user demand, finding appropriate services with reduced waiting times and allocating available resources for processing requests in IoT and fog settings is a challenge. Due to insufficient resources at edge devices, the requests require proper management. To address this problem, the fog environment offers edge devices with resources like as CPU, bandwidth, storage, and memory. The suggested technique uses fog devices to distribute resources to swarms in parallel, without breaking SLA, and performs better for diverse requests. This method ensures that all types of requests are handled differently, with varying response times. In this work, the reactive provisioning technique is used to satisfy demand for requests and ensure higher QoS. Future work in MOPSO will include prediction prior to provision and dynamic updating of inertia, acceleration, and cognitive variables.

Acknowledgements The authors thank Basaveshwar Engineering College, Bagalkot and BLDEA’s V.P. Dr. P. G. Halakatti College of Engineering and Technology, Vijayapur for providing the facilities and support in doing the work.

References

1. F. Khodadadi, R.N. Calheiros, R. Buyya, A data-centric framework for development and deployment of Internet of Things applications in clouds, in 2015 IEEE 10th International Conference on Intelligent Sensors, Sensor Networks and Information Processing ISSNIP 2015 7–9. <https://doi.org/10.1109/ISSNIP.2015.7106952>

2. M. Tropmann-Frick, Internet of things: trends, challenges and opportunities. *Commun. Comput. Inf. Sci.* **909**, 254–261 (2018). https://doi.org/10.1007/978-3-030-00063-9_24
3. S. Zhao, L. Yu, B. Cheng, An event-driven service provisioning mechanism for IoT (Internet of Things) system interaction. *IEEE Access* **4**, 5038–5051 (2016). <https://doi.org/10.1109/ACCESS.2016.2606407>
4. A.V. Dastjerdi, R. Buyya, Fog computing: helping the Internet of Things realize its potential. *Computer* **49**, 112–116 (2016). <https://doi.org/10.1109/MC.2016.245>
5. N. Wang, B. Varghese, M. Matthaiou, D.S. Nikolopoulos, ENORM: a framework for edge node resource management. *IEEE Trans. Serv. Comput.* 1–1 (2017). <https://doi.org/10.1109/TSC.2017.2753775>
6. M. Aazam, E.N. Huh, Dynamic resource provisioning through fog micro datacenter. *IEEE Int. Conf. Pervasive Comput. Commun. Workshop PerCom Workshop* **2015**, 105–110 (2015). <https://doi.org/10.1109/PERCOMW.2015.7134002>
7. M. Ketel, Fog-cloud services for IoT, in *Proceedings of the SouthEast Conference* (ACM, New York, NY, USA, 2017), pp. 262–264
8. A. Singh, D. Juneja, M. Malhotra, A novel agent based autonomous and service composition framework for cost optimization of resource provisioning in cloud computing. *J. King Saud. Univ. Comput. Inf. Sci.* **29**, 19–28 (2017). <https://doi.org/10.1016/j.jksuci.2015.09.001>
9. S.K. Sharma, N. Kumar, A modified particle swarm optimization for task scheduling in cloud computing *SSRN Electron. J.* 1–6 (2019). <https://doi.org/10.2139/ssrn.3368722>
10. O. Skarlat, S. Schulte, M. Borkowski, P. Leitner, Resource provisioning for IoT services in the fog, in *Proceedings of 2016 IEEE 9th International Conference on Service-Oriented Computing Application SOCA*, 2016, pp. 32–39. <https://doi.org/10.1109/SOCA.2016.10>
11. S.S. Aote, M.M. Raghuvanshi, R. Latesh Malik, A brief review on particle swarm optimization: limitations and future directions. *Int. J. Comput. Sci. Eng.* **2**, 2319–7323 (2013)
12. C. Li, L.Y. Li, Optimal resource provisioning for cloud computing environment. *J. Supercomput.* **62**, 989–1022 (2012). <https://doi.org/10.1007/s11227-012-0775-9>
13. O. Skarlat, S. Schulte, M. Borkowski, P. Leitner, Resource provisioning for IoT services in the fog, in *2016 IEEE 9th International Conference on Service-Oriented Computing and Applications (SOCA)*, pp. 32–39 (2016)
14. S. Singh, I. Chana, Q-aware: quality of service based cloud resource provisioning. *Comput. Electr. Eng.* **47**, 138–160 (2015). <https://doi.org/10.1016/j.compeleceng.2015.02.003>
15. Q. Zhang, M.F. Zhani, R. Boutaba, J.L. Hellerstein, Dynamic heterogeneity-aware resource provisioning in the cloud. *IEEE Trans. Cloud Comput.* **2**, 14–28 (2014). <https://doi.org/10.1109/TCC.2014.2306427>
16. J. Yao, N. Ansari, Fog resource provisioning in reliability-aware IoT networks. *IEEE Internet Things J.* **6**, 8262–8269 (2019). <https://doi.org/10.1109/JIOT.2019.2922585>
17. J. Yao, N. Ansari, QoS-aware fog resource provisioning and mobile device power control in IoT networks. *IEEE Trans. Netw. Serv. Manage.* **16**, 167–175 (2019). <https://doi.org/10.1109/TNSM.2018.2888481>
18. C. Avasalcai, S. Dustdar, Latency-aware distributed resource provisioning for deploying IoT applications at the edge of the network, in *Advances in Information and Communication*. ed. by K. Arai, R. Bhatia (Springer International Publishing, Cham, 2020), pp. 377–391
19. H.M. Fard, R. Prodan, F. Wolf, A container-driven approach for resource provisioning in edge-fog cloud, in *Algorithmic Aspects of Cloud Computing*. ed. by I. Brandic, T.A.L. Genez, I. Pietri, R. Sakellariou (Springer International Publishing, Cham, 2020), pp. 59–76
20. A.V. Chandak, N.K. Ray, Multi agent based resource provisioning in fog computing, in *Trends in Computational Intelligence, Security and Internet of Things*. ed. by N. Kar, A. Saha, S. Deb (Springer International Publishing, Cham, 2020), pp. 317–327
21. D. Kumar, Z. Raza, A PSO based VM resource scheduling model for cloud computing, in *Proceedings of 2015 IEEE International Conference on Computing Intelligent Communication Technology CICT*, 2015, pp. 213–219 (2015). <https://doi.org/10.1109/CICT.2015.35>
22. P. Civicioglu, E. Besdok, A conceptual comparison of the Cuckoo-search, particle swarm optimization, differential evolution and artificial bee colony algorithms (2013)

23. H.N. Pham-Nguyen, Q. Tran-Minh, Dynamic resource provisioning on fog landscapes. *Secur. Commun. Netw.* **2019**<https://doi.org/10.1155/2019/1798391>
24. I. Ullah, H.Y. Youn, Task classification and scheduling based on K-means clustering for edge computing. *Wirel. Pers. Commun.* **113**, 2611–2624 (2020). <https://doi.org/10.1007/s11277-020-07343-w>
25. M.C. Silva Filho, R.L. Oliveira, C.C. Monteiro, P.R. Inácio, M.M. Freire, CloudSim plus: a cloud computing simulation framework pursuing software engineering principles for improved modularity, extensibility and correctness, in *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)* (IEEE, 2017), pp. 400–406

DOMAIN-Based Intelligent Network Intrusion Detection System



Nithil Jose and J. Govindarajan

Abstract The state-of-the-art presently in the network intrusion detection, both in the network-level intrusion detection system and the host-level intrusion detection system, is completely based on the black box model which learns the pattern from knowledge database or from the dataset to the model. Proposed model is to combine the machine learning-based IDS approach and the domain knowledge incorporating method to build efficient and intelligent IDS which can be employed to detect typical intrusion and future intrusion which is not known. The idea behind is to make some data assimilation process in the features of the dataset such that a reduced and a meaningful feature set representation can be fed in to the model so as to construct intelligent generalized model which will be capable of handling unforeseen attack and new different kind of large data within in limited time period. May be with some compromise in the accuracy of the model but with increased generalizability.

Keywords Black box model · Domain feature model · Feature aggregation · Feature mapping

1 Introduction

The present-day network flow traffic makes a drastic change in the network infrastructure system and in the corresponding technologies. And, this makes the network intruders to use most advanced techniques and algorithm to make the intrusion possible. Hence, even the sophisticated malware detection tools become helpless before such kind of evolving type of malware algorithms and techniques. The recent days when artificial intelligence and machine learning became popular, the advancement and research had also undergone in the network security domain. As a result,

N. Jose · J. Govindarajan (✉)

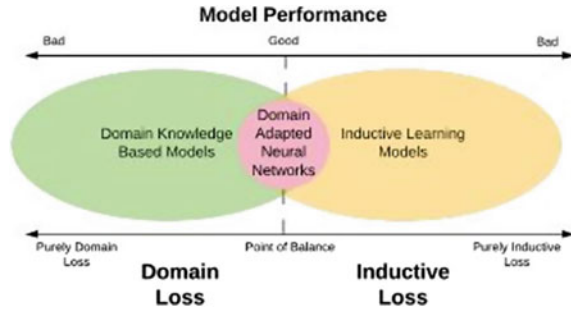
Department of Computer Science Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Coimbatore, India

e-mail: j_govindarajan@cb.amrita.edu

N. Jose

e-mail: cb.en.p2aid19030@cb.students.amrita.edu

Fig. 1 Balanced domain-adapted model [1]



there are number of intrusion detection models has been put forward, and it further leads to efficient analysis and detection of malwares, and intrusions became possible. The drastic revolutions happened in the Internet infrastructures after the introductions of IoTs, cloud platforms, distributed systems, and big data; the need for the most relevant domain-based intrusion detection models which would be capable of handling the unknown kind of data and attacks has been put forward.

Domain knowledge-adapted models have always the advantage of finding a balance between the domain loss and the inductive loss as shown in Fig. 1, in which the domain loss represents the loss which happens when a model completely depends on the features and input, whereas inductive loss happens when model completely stick on to the domain constraints.

Novelty of the proposed domain-based intrusion detections system (IDS) which makes the conventional machine learning and deep learning a explainable type rather than a black box model. The black box model is the model which is completely based on the data feed to the model where the system administrator or the model builder will not be aware of the particular domain, and he or she has to completely depend on the data feed to the model.

The proposed work inhibits domain knowledge from the global cyber security principles known as confidentiality, integrity, availability (CIA) by some data assimilation and feature mapping process; thus, the features will become more meaningful and generalizable can be fed to AI models. The proposed work was verified by comparing with conventional machine learning models and also with different datasets. The operation had been performed on three different AI models neural network (NN), random forest (RF), and logistic regression (LR) [2].

2 Related Works

Abstractive domain knowledge incorporating in the neural network is a reemerging field where the prominent works in the new era by Kumar and Soman et al. [3] focused on the dynamic nature of the malware. Flexible and effective algorithm for unforeseen and unpredictable attacks the deep learning model which can make

abstract and high-dimension feature representation of data by passing them through many hidden layers

Muralidhar et al. [4] defined a method to incorporate domain knowledge in many scenario where data is limited and of poor-quality domain knowledge can be integrated into model training for deep networks. Physical models, constraints, dependencies relationships, and knowledge of valid ranges of features are taken as the major domain knowledge factors. Approximation constraints (valid range) and monotonicity constraint (relationship between variables) are the two main constraints taken as domain knowledge-infusing techniques. Incorporating loss term as the knowledge available is type of monotonicity constraint

First, a good feature representation is learnt from a large collection of unlabeled data, termed as unsupervised feature learning (UFL). In the second stage, this learnt representation is applied to labeled data, x_l and used for the classification task was a another important research field [5]. A novel approach to select important features by utilizing two selected feature selection algorithms is utilizing filter approach. The selected features were further validated by domain experts where extra features were added into the final proposed feature set.

Li [6] used genetic algorithm for network intrusion. One network connection and its related behavior can be translated to represent a rule to judge whether or not a real-time connection is considered as an intrusion. The population evolves until the evaluation criteria are met. The generated rule set after the evolution will be the best fitted configuration and can be used as the knowledge to the model.

Yang and Shafto [7] utilize Bayesian teaching, where a more modest subset of models are utilized to prepare the model rather than the entire dataset. The subset of models is picked by area specialists as the models are generally pertinent to the issue of interest. Notwithstanding, for this reason, picking the right subset of models in reality is testing.

Niyaz [8] Sun proposed two channel-based element choice strategies were used at the element choice stage to deliver two elements correlation-based feature selection subset evaluator (CFSE) and consistency subset evaluator (CSE) used by these two-component determination techniques. CFSE utilizes a calculation that cooperates with an assessment equation, wherein the thoughts depend on test hypothesis. Great highlights are then chosen with a fitting relationship measure and a heuristic hunt technique. The calculation has the points of interest in recognizing insignificant, excess, and uproarious highlights quick. Important highlights can be recognized as long as their pertinence does not firmly rely upon different highlights.

A transfer learning approach for network intrusion detection by Wu et al. [9]. Their solution to this problem is first gaining basic knowledge from some existing base dataset and learning the target dataset based on the acquired knowledge

Zhang and Zulkernine [10] and A. Haque talked about arbitrary backwoods strategies in anomaly identification by learning examples of interruptions, abnormality identification with exception location system, and half-recognition by consolidating both the anomaly and inconsistency discovery. They announced that the abuse approach worked in a way that is better than winning sections of KDDCup 99

test results, and in expansion, oddity location worked better contrasted with other distributed solo oddity location strategies.

Harini [11] given the idea proxy-based signature system which enables the authentication process to be happened in the IoT itself. Signature system based on the ECC provides better scheme of authentication and helps in the unauthorized access and data assimilations.

Govindarajan and Kousalya [12] proposed a methodology or a protocol for satellite network which will be capable of handling both real-time and non-real-time application based on the new discriminative algorithm. The proposed method was capable of reducing the high error and high congestions rate in the network traffic.

Hu and Maybank [11], an ID calculation utilizing AdaBoost procedure was recommended that used decision stumps as powerless classifiers. Their framework performed in a way that is better than other distributed outcomes with a lower bogus alert rate, a higher recognition rate, and a computationally quicker calculation. Nonetheless, the downside is that it neglected to embrace the gradual learning approach

In the other work, Govindarajan and Anusuya Devi in [13] raised a new method to find the unfairness in the TCP protocol data by analyzing the MAC-level information, and other simulation tools were an effective work to detect the unfairness.

Daweri and Salam in [14] make analysis study between the KDDCup 99 dataset and the UNSW NB 15 dataset so that the feature is related, what are the kind of attack category in both datasets, and how they can be related. They had used mainly three methods for the analysis: the rough set theory, back propagation neural network, and a discrete variant of Cuttle fish algorithm.

Using the knowledge model and the hierarchical machine learning model, another intrusion detection system was proposed by Sarnovsky and Paralic [15]. Combination of different set of hierarchical machine learning models and the ontology-based knowledge model is used for intrusion detection.

A Pyspark-based distributed type of anomaly detection system was proposed by Ranganathan G. which was a real time-based system. Clustering method was used by observing various parameters like CPU load and memory usage.

From this discussion, however, we can find different efficient works capable of dealing with the anomalies, feature selection, algorithmic models, but these systems are not capable to handle today's huge network traffic flow and mostly to handle the unknown new data, connections, and intrusion. Since there is no balance between the two-loss mentioned, these systems will perform mostly as the supervised learning method

3 Problem Formulation

Research problems to be addressed:

- Real-time implementation to reduce the time for intrusion detection or the model training.
- Add the domain knowledge to the system (domain knowledge + model)
- Considering both payload and packet in the network connection for the model
- For unfamiliar data, like different protocol, services, versions.

4 Dataset

Because the data packets of the UNSW NB 15, informational index is made by the IXIA perfect storm device in the Cyber Range Laboratory of the Australian Centre for Cyber Security (ACCS) for producing a mixture of genuine present-day typical exercises and contemporary assault practices. Tcp dump device is used to catch 100 GB of the crude traffic (e.g., Pcap documents).

This informational index has nine groups of assaults, in particular, fuzzers, analysis, backdoors, DoS, exploits, generic, reconnaissance, shell code, and worms. The Argus and Bro-IDS apparatuses are used, and twelve calculations are created to produce absolutely 49 highlights with the class label. The number of records in the training set is 175,341 precedents and the testing set is 82,332 precedents from various sorts of assault and type.

The attack classes were of nine types in which fuzzers normally scan the system for the security loopholes by inputting huge amount of random data and makes the system busy. Backdoor makes unauthorized access to the system from the remote location, whereas the exploit is set of instruction which makes use of the bugs in the system by unsuspected hosts or network.

Analysis is a web application-based attack. Dos makes network resources unavailable. Reconnaissance gathers the information about the system to invade them. Worms are of malicious replicating-type code, and generic is a method which works against the block ciphers [16].

5 Our Model

The workflow starts with the black box neural network model that is with feeding all available features to the hidden layers of the network. And, considered as the base line implementation for both UNSW NB-15 dataset and observed the confusion metrics. Proposed model implementation starts with domain feature sets-infusing techniques, where the most correlated features or the important feature are identified from the table for UNSW NB-15 dataset which was constructed by using the correlation coefficient calculated from the Pearson correlation coefficient. Then, each feature is mapped to the type of attack from the Table 1. According to the compromise in the security principle, features are again mapped to the CIA (Fig. 3).

Table 1 Types of attacks

Attack type	ID
Fuzzer	1
Backdoor	2
Exploit	3
Analysis	4
Dos	5
Reconnaissance	6
Shell code	7
Generic	8
Worm	9
Normal	0

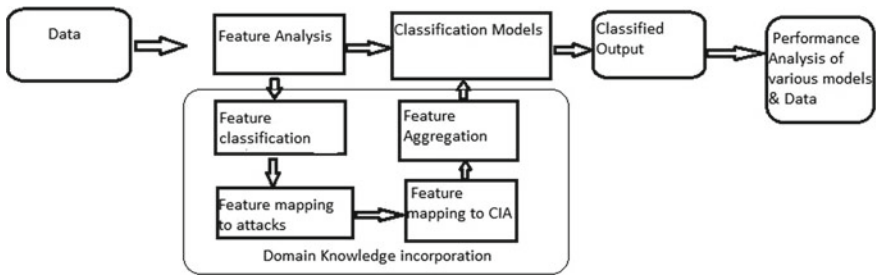


Fig. 2 Domain-based intrusion detection architecture

Fig. 3 CIA triad [18]



The feature analysis is the detailed study of the dataset feature by feature for their importance and correlation to the target variable, and this includes the preprocessing operations like cleansing operation and filling missing data. Feature classification was an important process since it helps the feature mapping based on whether each feature belongs to connection, flow, content, time type. Based on this classification, feature mapping (explained in Sect. 5.3) to attack was performed in which how each attack class is affecting the each feature value is studied with the help of reference [14] based on this mapping by relating the corresponding compromises in CIA (explained in Sect. 5.2) Feature mapping to CIA was performed. Results of all these mapping were concluded in the constructed reference table [17].

After the feature mapping, the relationship coefficient vector of the features to comprehend by feature aggregation (explained in Sect. 5.4) whether the expansion or increase in the value of an element has a positive or negative effect on the objective variable. At that point, convert the relationship coefficient vector V into a 1 or -1 dependent on whether the correlation coefficient is positive or negative appropriately.

According to the correlation coefficient (explained in Sect. 5.1) calculated, the features are transformed into a new three groups or feature CIA using the feature aggregation. If the correlation coefficient vector (V_i) for a feature (C_i) of that group has a negative value, then the product of the feature value and the correlation coefficient for that feature is deducted, and vice versa if positive, thus, domain incorporated reduced feature set is formed.

After data assimilation process, implemented interesting classification models. First one was domain-adapted neural network (DNN) for network-level intrusion detection system (NIDS) and host-level IDS (HIDS), made out with one input layer, one hidden layer, and a output layer. The hidden layers in the DNN encourage separating exceptionally complex features and improving design recognition or pattern recognition abilities in IDS information. Each layer handles nonlinear features that are passed to the following layer, and the last layer in the DNN plays out the classification followed by random forest and logistic regression implementation.

5.1 Pearson Correlation Coefficient

$$r = \frac{\sum(xi - x^1)(yi - y^1)}{\sqrt{\sum(xi - x^1)(yi - y^1)}} \tag{1}$$

- r correlation coefficient.
- xi value of the x variable
- x^1 mean of the x
- yi value of the y variable
- y^1 mean of the y

Pearson correlation coefficient shows the linear dependency between two variable, and it is covariance between two variable. It gives how each feature in the dataset is

correlated to the target variable so that it can be used as one of the feature selection criteria.

5.2 *CIA Cyber Security Principle*

The CIA triads [18] are the most acceptable cyber security principle that can be adopted in intrusion detection scenario. This triad gives the idea of balance between the confidentiality integrity and availability of a network resource.

Confidentiality can be explained as the control over the data so that the only authorized person will be allowed to access the data and resource network. Thus, to avoid the misuse of resource, it is the most obvious one among the three triads.

The integrity mainly focuses on the purity of the data resource in which sender or receiver has to ensure no unwanted changes or adulteration should happen in the resource. Miscommunication or the data loss is the main problems happens due to the compromise in the integrity of the data.

The availability ensures the authorized persons to have easy access to the resource without any delay or the hindrances. Meanwhile, without compromising the security, for example, a block on the payroll or the email sent by the clients etc.

5.3 *Features Mapping*

Feature mapping was done manually from reference [14] and from other data analysis from previous work, and each feature was analyzed carefully and categorized based on whether it comes under flow, time, and content, and additional kind of feature from this, we can further map which kind of attacks will mostly affect particular category. If the attack category was known, we can easily map to the corresponding CIA

5.4 *CIA Aggregation*

According to the Pearson correlation coefficient features that are more correlated and according to their importance, the mapping table is formed based on the compromises under CIA principles. Creating additional feature which transforms the feature dataset in to a meaningful one. The coefficient is multiplied with the feature variable and add or subtract to the corresponding newly formed feature based on the given below Equations 2–4. C_i stands for the mapped feature vector, and V_i stands for correlation coefficient.

$$C = \sum_{k=0}^n C_i V_i \quad (2)$$

$$I = \sum_{k=0}^n I_i V_i \quad (3)$$

$$A = \sum_{k=0}^n A_i V_i \quad (4)$$

5.5 Preprocessing

The data cleansing was performed to avoid the unwanted strings in the integer-type features and followed data filling process to fill the missing fields in the features by appropriate method like filling with mean, forward fill. Label encoding and one hot encoding are used to encode the label and attack type class features. Standard scaling was applied to the entire dataset to reduce the values in between +1 and -1 in order to feed into machine learning models. Upsampling using SMOTE was done since many of the attack type classes are small in size which will affect the balance of dataset. After feature aggregation, Minmax scalar is to keep the values of the features in between +1 and -1.

5.6 Experiments

Altogether, three experiments were carried out based on the three models neural network, random forest, and logistic regression. Proposed model was fed with both the All-feature data (dataset with every features) and the domain feature data (data after data assimilation process). The neural network model implemented with one hidden layer and ReLU as the activation function for input and hidden layer, sigmoid for the output layer. We selected Adam optimizer as the optimizer for the neural network and run the training for 200 epochs with 1000 as the batch size. Similarly, for the random forest, n_job is given as 4 and estimators = 10, rest as the default parameters. In the logistic regression, random state is taken as 0 and n_jobs = 4.

5.7 Evaluation Methods

We exploit confusion matrix metric to evaluate our model. Also used query function to detect the number of correctly detected cases for a particular attack.

5.8 Comparison Methods

The model performance has been verified with the UNSW dataset. The metrics used for the purpose were confusion metrics. The proposed model was compared with conventional all-feature model where the entire dataset was taken after proper data preprocessing. The entire work says the performance of three different machine learning models, i.e., deep neural network, random forest, logistic regression. In order to verify whether performance of the model on the unforeseen attack, particular class of attack records has been deleted from the entire training dataset and verified by including the corresponding records in the test data. As further additional step, a function was used to verify the actual number of correctly detected cases for the unforeseen kind of data.

6 Results

The metrics used for the evaluation of various model or feature settings are accuracy and also the false positive from confusion matrix (CM) rate since the false positive case was the worst case in the intrusion detection scenario where the intrusion is detected as the normal data

$$\text{Accuracy} = \frac{\text{All positive cases}}{\text{All cases}} \quad (5)$$

There are mainly three outcomes for the overall experiment, and the first outcome was obtained by taking the CM of the predicted model shown in Sect. 6.1. The results are compared with that of the CM of conventional model (all-feature model) to understand the performance of the proposed model. In Sect. 6.2, we can see the same comparison in the case of unknown attack case, i.e., records belong to one particular kind of attack were deleted during data training from the data. Third outcome in Sect. 6.4 shows the number of correctly detected cases in the predicted data by the model in the unknown attack case, and this was performed by using query function in the predicted and original dataset.

6.1 Evaluation of Confusion Matrix of Known Attack Cases

The all-feature model has the advantage of maximum number of features that will help to find the pattern very easily. Tables 2 and 3 show the same, but act as an overfitted model completely based on the dataset in which model has no role with the unknown attack cases. Here, in this case, we can see the accuracy for the all-feature model is comparatively higher than that of the domain-based model

Table 2 Comparison of CM and FP of UNSW NB 15 normal dataset

S. No.	Dataset	Model	Accuracy	Precision	False positives
1	UNSW (domain feature)	Neural network	80	92	662
		Random forest	82	81	2607
		Logistic regression	76	84	1728
2	UNSW (all feature)	Neural network	90	95	2420
		Random forest	92	94	2942
		Logistic regression	85	90	1944

Table 3 Comparison of CM and FP of UNSW NB 15 unknown attack dataset

Unknown attack cases					
S. No.	Dataset	Model	Accuracy	Precision	False positives
1	UNSW (domain feature)	Neural network	80	95	223
		Random forest	82	82	2517
		Logistic regression	76	84	1714
2	UNSW (all feature)	Neural network	90	92	2254
		Random forest	92	95	2896
		Logistic regression	90	96	1826

6.2 Evaluation of Confusion Matrix of Unknown Attack Cases

6.3 Analysis Based on CM

The analysis was made by comparing the proposed model as shown in Figs. 4 and 5 with that of the conventional model in which all features have been taken into consideration. From the confusion matrix, the accuracy and false positive rate had taken for analysis. The analysis was done on three different models neural network, random forest, and logistic regression.

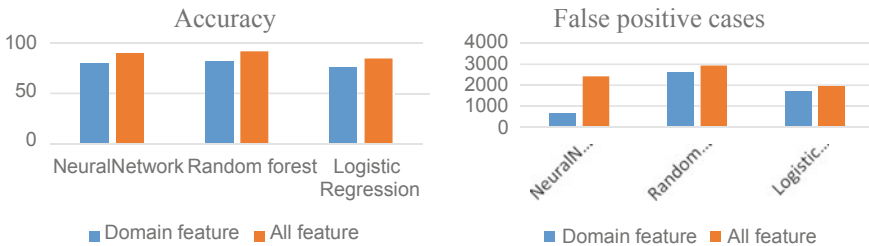


Fig. 4 Analysis based on CM for known attack

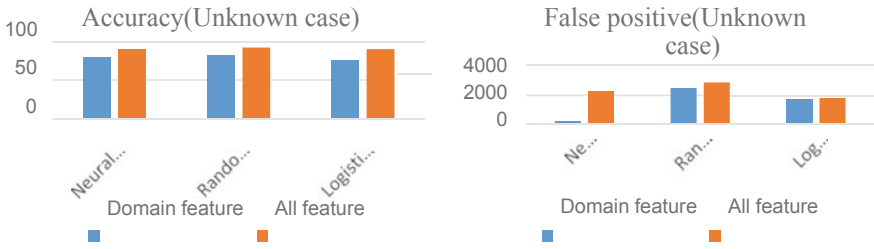


Fig. 5 Analysis based on CM for unknown attack

6.4 Analysis Based on Correctly Detected Cases

Another analysis was performed using the query function as shown in Fig. 6, and the main objective is to find the correctly detected cases for a particular kind of attack which was not included at the time of data training that means the data records belong to one particular kind of attack have been deleted from the dataset during the training process. Figure 6 shows the pie chart for the correctly detected cases for attack id (1, 3, 5, 8) see Table 1.



Fig. 6 Correctly detected attack case

7 Conclusion

The absence of domain knowledge or explainable nature of the AI models is important especially in the most critical area like medical and defense sectors. The black box models performing better are failed to reduce the false prediction rate which cannot be afforded in this kind of sectors. To moderate this issue, mix the CIA guideline (i.e., generalized security information) in the AI-based discovery model for better reasonableness and generalizability of the model.

Our test results show feasible accomplishments in better reasonableness with a far reaching, cutting-edge, and real-world network interruption dataset. Also, the infused domain information helps in distinguishing an obscure intrusion as it sums up the issue, which eventually makes the way for reducing the false predictions.

7.1 Future Enhancement

The main drawback of the proposed work is that they are not generalizable for every dataset since the feature mapping work was done by manually and not automated since due to technical limits. The automation of this part will make this model a generalizable and most efficient one which will be capable of implementing in the real-time scenario.

References

1. <https://ieeexplore.ieee.org/abstract/document/8621955/>
2. <https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/>
3. R. Vinayakumar et al., Deep learning approach for intelligent intrusion detection system. *IEEE Access* **7**, 41525–41550 (2019)
4. N. Muralidhar, et al., Incorporating prior domain knowledge into deep neural networks, in *2018 IEEE International Conference on Big Data (Big Data)* (IEEE, 2018)
5. A. Javaid, et al., A deep learning approach for network intrusion detection system, in *Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS)* (2016)
6. W. Li, Using genetic algorithm for network intrusion detection. *Proc. United States Dept. Energy Cyber Secur. Group* **1**, 1–8 (2004)
7. S.C.-H. Yang, P. Shafto, Explainable artificial intelligence via bayesian teaching, in *NIPS 2017 Workshop on Teaching Machines, Robots, and Humans*, 2017
8. Q. Niyaz, W. Sun, A.Y. Javaid, *A Deep Learning Based DDoS Detection System in Software-Defined Networking (SDN)*. [arXiv:1611.07400](https://arxiv.org/abs/1611.07400) (2016)
9. P. Wu, H. Guo, R. Buckland, A transfer learning approach for network intrusion detection, in *2019 IEEE 4th International Conference on Big Data Analytics (ICBDA)* (IEEE, 2019)
10. J. Zhang, M. Zulkernine, A. Haque, Random-forests-based network intrusion detection systems. *IEEE Trans. Syst. Man Cybern. C Appl. Rev.* **38**(5), 649–659 (2008)
11. N. Harini, D. Kamakshi, D. Aruna, Secure proxy blind ECDS algorithm for IoT. *Int. J. Pure Appl. Math.* **118**(7), 437–445 (2018)

12. J. Govindarajan, G. Kousalya, Cooperative flow regulation protocol for real-time and non-real-time applications over satellite network. *J. Ambient Intell. Human. Comput.* **12**, 979–990 (2021); Z. Zhu, Y.-S. Ong, M. Dash, Wrapper-filter feature selection algorithm using a memetic framework. *IEEE Trans. Syst. Man Cybern. Part B (Cybern.)* **37**(1), 70–76 (2007)
13. J. Govindarajan, G. Anusuya Devi, G. Kousalya, Analysis of TCP-unfairness from MAC layer perspective in wireless ad-hoc networks. *Indian J. Sci. Technol.* **8**(19) (2015)
14. A. Daweri, M. Salam, K.A.Z. Ariffin, S. Abdullah, An Analysis of the KDD99 and UNSW-NB15 datasets for the intrusion detection system. *Symmetry* **12**(10), 1666 (2020)
15. M. Sarnovsky, J. Paralic, Hierarchical intrusion detection using machine learning and knowledge model. *Symmetry* **12**(2), 203 (2020)
16. https://www.researchgate.net/publication/335639462_Machine_learning_approach_for_information_security
17. <https://drive.google.com/file/d/1P99M-7whaFzeNvkFd2jaD1jaqUVpUakc/view?usp=sharing>
18. <https://www.deepwatch.com/blog/cia-in-cybersecurity/>

Movie Recommendation System Using Color Psychology Based on Emotions



G. R. Ramya and Priyadarshini Bhatnagar

Abstract Emotions are the undeniable reliance of information in bridging human and machine intercommunication. Machines are able to recommend better when they can comprehend an individual's emotions. Producing emotions in the users is conventionally recognised as the fundamental goal of movies. Hence, movie recommendations based on one's emotional trajectory is key as it allows them to map movie recommendations based on their emotional stage. This paper uses color psychology in capturing user emotions. Two criteria including collaborative filtering (CF) and content-based filtering (CBF) are applied in comparing the apparent user interest profile and identifying like minded users with their cross-recommended items. This is achieved using algorithms with the ability of computing the recommended hybrid and detecting prevalent the emotions. As a result, the system seeks efficiency improvement and enhancement of quality of recommendations.

Keywords Content-based filtering · Collaborative filtering · Movie recommendation · Color psychology · K-nearest neighbours · Singular value decomposition · Hybridisation

1 Introduction

Human reviews which are mainly impersonal and generic were the most widely used recommendation sources. Though the concept of recommendation systems were first introduced in the mid-1990s, it is only until recently that they have come into regular usage. Notably, the recommendation concepts were formulated in 1990s, but only been applied regularly in the recent years.

G. R. Ramya (✉) · P. Bhatnagar
Department of Computer Science and Engineering, Amrita School of Engineering, Amrita
Vishwa Vidyapeetham, Coimbatore, India
e-mail: gr_ramya@cb.amrita.edu

P. Bhatnagar
e-mail: cb.en.u4cse17242@cb.students.amrita.edu

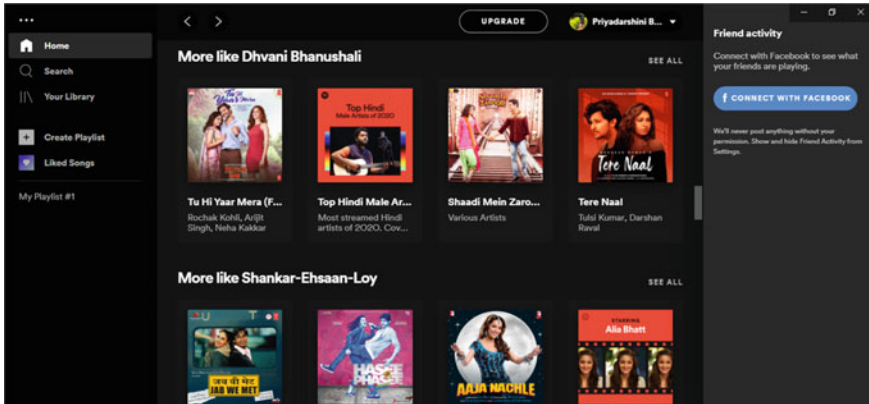


Fig. 1 Spotify recommending similar songs based on singers

Recommendation system is a platform that offers its users with numerous innards depending on their tastes and partialities. As such one can therefore go through a range of desired items that are available over music streaming platforms like Spotify or buying items on an e-commerce site as in Fig. 1. Sequentially, it is easier to determine user interest and the identification of specific user interest and needs from larger sets of objects through the recommendation systems.

From recommending courses on e-learning sites like Coursera, EdX, etc. to recommending books based on your previous reads to recommending potential mates on dating sites, recommendation systems can be used for many applications. We use recommendations systems in order to help the users find the right products of their particular interest. It also helps the providers to bring their products to the correct set of target audience. They help identify the most similar products to a particular user, showcase personalised content to each user and suggest top discounts and notifications to improve the user consumption, thereby, the business revenue.

In this paper, our proposed movie recommendation system uses Collaborative Filtering and Content Based Filtering methods to provide definitive recommendations matching the needs of the user by the input of multiple colors which reflect the emotions of the user. Emotions play a critical role in prediction process. Usually, emotions are not taken into account for recommendation in e-commerce sites like Flipkart, Amazon etc. since the objects sold are not of that great importance to the user's emotional wellbeing whereas this can play a vital role in mood upliftment in the case of music, movies, web-series etc. Our system will take multiple color inputs and will recommend movies which has majority of colors given in the input. So if the user has given four colors example: red, yellow, blue, green, the algorithm will filter out movies which may represent these colors and will recommend based on highest similarities. The accuracy of the algorithm is checked with the help of algorithms like KNN in the case of CBF and other algorithms for CF, ensuring the final hybridised results are precise.

The issue with Recommender systems are like that its complex and filtering and retrieving becomes a tough job when there is a huge dataset. We have used Movie Lens 1 M dataset containing 1 Million movie information along with the ratings dataset that has multiple user ratings for the movies present in the dataset. It will help us as we will know the genre of the movie each user likes. Recommendation process will ease up if we have some user information along with the color input. We have also worked on the new user registration and tried proposing movies in that case as well. The system gives effective recommendations and enables users to co-relate with their emotional welfare too.

This might be of great importance in the present times since the mental and emotional health of the people having been affected adversely due to pandemic. Also further improvement might involve connecting through various people globally and expanding the horizon of movies one watches.

2 Literature Survey and Related Works

In film making, producers hire designers to design the background tones of a scene in order to set a mood to the scene even before the actors have uttered a single word [1]. Color grading has opened a lot of various possibilities to colors in films. We have used the same concept of color psychology to retrieve genres that trigger the user feelings based on their choice of colors.

In terms of methods used, collaborative filtering can be described as the process of automatically predicting user interests by collecting information about the tastes or preferences of many users [2]. Collaborative filtering applications involved very large datasets. These methods have been applied to various kinds of data like sensing and monitoring data.

CBF is a technique of selecting items on the basis of the correlation between movie content and the user's preferences as opposed to CF that chooses movies based on correlation between users. It usually works by taking data from the users, which can include be either explicit, like ratings or implicit, like user clicks [3].

Hybrid recommendation systems combine CF and CBF systems in order to reduce the drawbacks caused by each of them [4]. Based on the movie (genre) watched by the user, his emotions are triggered. In the same way, feelings of a person might cause the want to watch particular type of movies. Collaborative and content filtering algorithm used with a mixed form [5]. User specified data was clubbed to make a cluster. To predict the difficulty and uniqueness level of each individual's prediction, they proposed a method called content boosted collaborative filtering (CBCF).

With all the research papers we referred, we came to a conclusion that basic topic modelling and filtering techniques were just not enough for product recommendation systems. There is a need to combine two or more techniques to achieve more accuracy. A need to improve the latest MRS emerges for more benefits of the users. Colors will be more relatable to the movies and will also be attractive and convenient for human

searches. Hence, our goal is to perform a hybrid recommendation system based on emotions use both CF, CBF and color psychology.

The approach in [6] uses building the recommendation system by content-based filtering. The system aims at providing movie recommendation based on the genres of the movies. If a user highly rates a movie of a particular genre, movies containing similar genres will be recommended to him. The dataset is subdivided into two sections. One section contains the list of movies along with the genres that they have been categorized. The other part of the dataset contains a list of ratings of movies that have been rated by the user. A combined dataset of movies, genres and their ratings is constructed. The ratings have been converted to binary values. If the rating given by a particular user is greater than 3, it receives a value of 1, otherwise it receives a value of -1 . The genres are also segregated in a binary format.

Out of all genres that are present in total, if a movie has a certain genre, it receives the value of 1 else 0. The user profile matrix is constructed by computing the dot product of the genre and the ratings matrix. If the dot product is a negative value, 0 is assigned to it else 1 is assigned. For a positive value, 1 is assigned to it. After obtaining a dot product matrix of all the movies, a similarity measure is calculated by computing the least distance between the user under consideration and the others. The values which have the least deviation with respect to the current user's preferences are the ones that are recommended by the system.

The paper [7] presented a movie recommender system based on collaborative filtering using Apache Spark. The parameter selection is seen to be affecting the performance of the system. RMSE is used for evaluating the performance. Best cases are selected for selecting the parameters and the model which gives the lowest RMSE is selected. But in this case no proper algorithm explanation was given and user profile, emotions was not taken into consideration.

3 Proposed Methodology

Given colors as input to define the emotional state of the user and recommend movies according to the emotional state of the user. Relate the color according to emotions like sad, happy, angry etc. With the movie dataset given, relate each movie with a color and return the movies which best match the input color.

Figure 2 shows the proposed model work. Relate the genres of the movies to specific colors based on color psychology.

- Identified the unique genres in the movies data frame and have assigned colors (multivalued) since one movie can have multiple numbers of genres.
- Taken user input of colors and filtered only those movies which contain those colors. Later we have merged it with the ratings given by the users.
- Cosine similarity was found.
- Performed collaborative filtering memory based on item-item and user-user and stored the results.

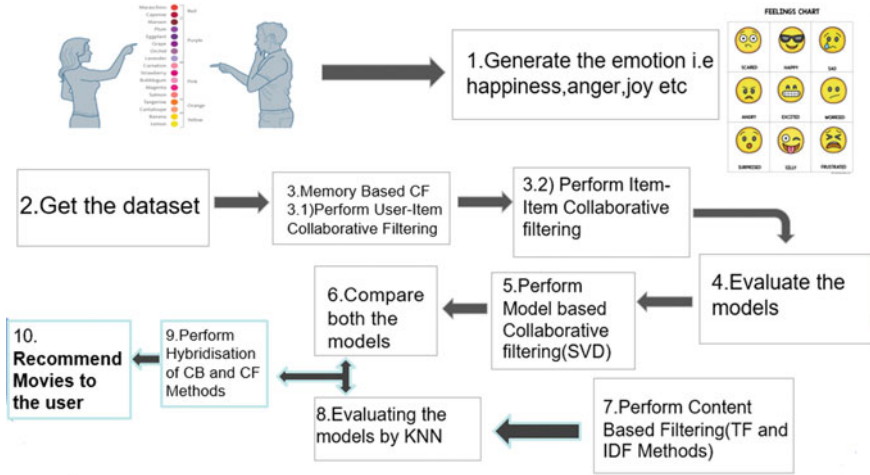


Fig. 2 Proposed model

- Performed content based filtering using TF-IDF method. Here the movies recommended by the results of CF is taken as input and passed onto the function and movies with similar genres are recommended. Movies already watched are dropped.
- The results of both CF and CBF is combined, the movies with SVD predicted estimates or SVD similarity between 2 movies is less than 2, then those movies are dropped.

The project is divided into 4 phases. The first phase being the registration of new user, generation of emotions based on the color input. For users already existing in the dataset, their User_ID and color input is taken. For new users, the user the supposed to register with User_ID and password. If the ID is already existing in the database, the user is asked to create a new ID.

To know more about new user, we have asked to rate some random movies from 1 to 5 if they have watched any. This will help in better recommendation in the final results. The rated movies by the new users are added to the Ratings dataset. Figure 3 shows the sign up page of the user and Fig. 4 shows the ratings movies if new users have watched.

Since, we do not have the color input for the movies dataset, so we have used the following algorithm in order to give different colors based on the genres to the movies. We have assigned multiple colors to multiple genres. This is done on the raw dataset as its the integral part of the system.

Figure 5 shows the color assignment code for movies. Here df is the data frame name which stores the movies dataset. np is Numpy library imported in python. We have created a new column storing all the values. The assignment of colors is based on color psychology. Figure 6 shows the resultant movies and ratings dataset.

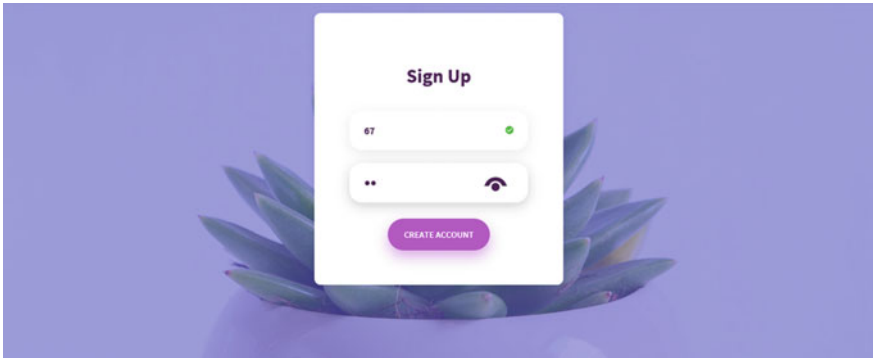


Fig. 3 Sign-up page for new users

Re-Enter your ID: 676767

Sno	Movie Name	1	2	3	4	5
1	Toy Story (1995)	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
2	Jumanji (1995)	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3	Grumpier Old Men (1995)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
4	Waiting to Exhale (1995)	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Fig. 4 Rating movies if new users have watched

3.1 Color Psychology

The movies we watch in our daily lives, color plays a major role. It is used to set the mood of a scene before the actors utter a single word or perform a scene. According to the researchers, a person who is happy finds bright colors more attractive, where as a person who is sad or anxious relates their current mood with dull colors. For example, when we watch the color red our mind relates to danger, power or passion. Similarly, pink denotes innocence, beauty and delicacy. Yellow is related to happiness, naivety, natural and insanity. Blue symbolizes passivity, outlying and calmness. Each color carrying a specific meaning will help in understanding different moods of the user. Using similar analogies, we have assigned different colors to the movies in

```
[ ] conditions = [
    (df['tag'] == "Animation"),
    (df['tag'] == "Action"),
    (df['tag'] == "Adventure"),
    (df['tag'] == "Children's"),
    (df['tag'] == "Comedy")
]

# create a list of the values we want to assign for each condition
values = ['violet,pink,blue,orange,yellow', 'red,orange,blue', 'orange,blue', 'yellow,blue,pink', 'yellow,orange',]

# create a new column and use np.select to assign values to it using our lists as arguments
df['colour'] = np.select(conditions, values)
```

Fig. 5 Color assignment code for movies

accordance with the genres. The input colors to formulate the recommender system algorithm is listed below.

- Action:** red, orange, blue
- Animation:** violet, pink, blue, orange, yellow
- Documentary:** orange, yellow, green, blue, brown
- Drama:** blue, pink, white, red
- Musical:** green, blue, purple, cyan
- War:** yellow, orange, red, blue, black

Figure 7 shows the input colors to formulate the recommended system. We then take the input from the user. A user can enter multiple colors and the input will be used to filter out a movie which has these colors assigned in the newly added column. This will help in further reducing the size of the dataset.

Phase 2 deals with Collaborative Filtering algorithm.

3.2 Collaborative Filtering

Recommendations are made possible after the resemblances between the elements and the users are tackled through collaborative filtering. Recommendations are made after the algorithm determines the relation between items and the users. Its aim is to find the similar users to our target user in the large set of users. Such is possible by finding a lesser set of users searched from a large set of users. A ranked list of suggestions is determined by looking and combining liked items. Resultantly, movie recommendation between users will be used to determine similarities between items and users.

movie_id	movie_name	tag	colour
0	1 Toy Story (1995)	Animation Children's Comedy	violet,pink,blue,orange,yellow
1	2 Jumanji (1995)	Adventure Children's Fantasy	orange,blue,pink,yellow,purple
2	3 Grumpier Old Men (1995)	Comedy Romance	red,pink,green,black,yellow,orange
3	4 Waiting to Exhale (1995)	Comedy Drama	red,orange,blue,pink,yellow,white
4	5 Father of the Bride Part II (1995)	Comedy	yellow,orange
...
3878	3948 Meet the Parents (2000)	Comedy	yellow,orange
3879	3949 Requiem for a Dream (2000)	Drama	blue,pink,white,red
3880	3950 Tigerland (2000)	Drama	blue,pink,white,red
3881	3951 Two Family House (2000)	Drama	blue,pink,white,red
3882	3952 Contender, The (2000)	Drama Thriller	blue,pink,white,red,black

UserID	MovieID	Ratings	Timestamp
0	1	1193	5 978300760
1	1	661	3 978302109
2	1	914	3 978301968
3	1	3408	4 978300275
4	1	2355	5 978824291
...
1000204	6040	1091	1 956716541
1000205	6040	1094	5 956704887
1000206	6040	562	5 956704746
1000207	6040	1096	4 956715648
1000208	6040	1097	4 956715569

1000209 rows x 4 columns

Fig. 6 Resultant Movies and Ratings Dataset

for $i = 1$ to $i = n$:

for $j = 1$ to $j = n$:

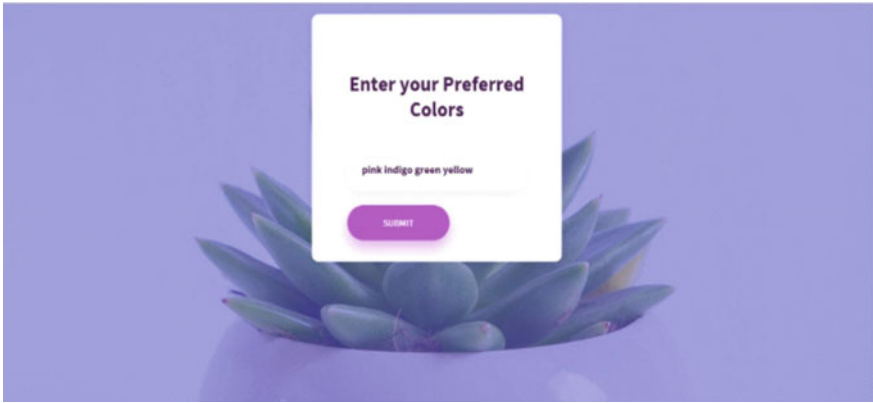


Fig. 7 Input colors to formulate the recommender system algorithm

$$\begin{aligned} & \text{if } i == j : \text{user}[i][j] = 1 \\ & \text{else : } \text{user}[i][j] = ((\text{Number of similarity rated items} * 5) \\ & \quad - (\text{sum}(\text{abs}(\text{user}[i] - \text{user}[j]))) \\ & \quad / (\text{Number of similarly rated movies} * 5)) * 100 \quad (1) \end{aligned}$$

There are two types of collaborative filtering methods we have incorporated in our system, Memory based and Model based methods:

3.2.1 Memory Based Methods

Item Based Filtering

The system identifies one’s taste of product and recommends it to another user. In measuring the similarities, similarity metrics are used. We are finding the similarity between the users using cosine similarity. Cosine similarity calculates the cosine of the angle formed by two vectors in their dot product space to determine their similarity.

The User_Id is passed into the algorithm and Collaborative filtering method forms a similarity matrix between the movies the user has watched with respect to the rest of the movies present in the dataset based on filtering with the help of colors. The movies with higher similarity to the movie already watched by the user are recommended to the user. Figure 8 shows the item similarity matrix, each movie is represented in a row and the different movies are presented in the column. Figure 9 shows the item based collaborative filtering code.

	0	1	2	3	4	5	6	7	8	9	...	256
0	0.000000	0.291487	0.290466	0.168894	0.274611	0.236920	0.091567	0.209844	0.147476	0.258509	...	0.035396
1	0.291487	0.000000	0.442444	0.136846	0.361104	0.207971	0.104685	0.115306	0.183647	0.216945	...	0.033209
2	0.290466	0.442444	0.000000	0.136884	0.474758	0.166875	0.096061	0.129562	0.173472	0.232111	...	0.064153
3	0.168894	0.136846	0.136884	0.000000	0.142239	0.081117	0.033503	0.269776	0.041963	0.132644	...	0.042092
4	0.274611	0.361104	0.474758	0.142239	0.000000	0.185550	0.139823	0.141356	0.195404	0.251207	...	0.092230
...
261	0.064399	0.050154	0.068641	0.126848	0.091437	0.086438	0.000000	0.104038	0.038304	0.056881	...	0.029596
262	0.067336	0.111165	0.083104	0.048766	0.093331	0.017453	0.054877	0.065858	0.037381	0.010794	...	0.072206
263	0.041631	0.095400	0.056986	0.024352	0.064600	0.064743	0.033929	0.010179	0.104002	0.045760	...	0.022321
264	0.071479	0.113779	0.076154	0.028862	0.128538	0.114818	0.021417	0.009179	0.081280	0.063960	...	0.022544
265	0.173265	0.104357	0.129652	0.155952	0.115659	0.054440	0.051914	0.140601	0.033690	0.097457	...	0.064247

Fig. 8 Item similarity matrix

```

def item_similarity(movieName):
    """
    recommends similar movies
    :param data: name of the movie
    """
    try:
        user_inp=movieName
        inp=rslt_df[rslt_df['movie_name']==user_inp].index.tolist()
        inp=inp[0]
        rslt_df['similarity'] = ratings_matrix_items.iloc[inp]

def recommendedMoviesAsperItemSimilarity(user_id):
    #Recommending movie which user hasn't watched as per Item Similarity
    user_movie= merged[(merged.UserID==user_id) & merged.Ratings.isin([5,4])][['movie_name']]
    user_movie=user_movie.iloc[0,0]
    item_similarity(user_movie)
    sorted_movies_as_per_userChoice=rslt_df.sort_values( ["similarity"], ascending = False )
    sorted_movies_as_per_userChoice=sorted_movies_as_per_userChoice[sorted_movies_as_per_userChoice['similarity'] >=0.5]
    ['movie_id']
    #Taking movies with similarity greater than 0.5
    recommended_movies=list()
    df_recommended_item=pd.DataFrame()
    user2Movies= df4[df4['UserID']==user_id][['movie_id']]
    for movie_id in sorted_movies_as_per_userChoice:
        if movie_id not in user2Movies:
            df_new= df4[(df4.movie_id==movie_id)]
            df_recommended_item=pd.concat([df_recommended_item,df_new])
            best10=df_recommended_item.sort_values(["Ratings"], ascending = False )[1:10]
    return best10['movie_id']
    
```

Fig. 9 Item-based collaborative filtering code

From the matrix, the movies which have not watched by the user and has high similarity with the other movie is recommended. Figure 10 shows the results of the Item CF.


```

- Recommended movies,:
  [1271  Indiana Jones and the Last Crusade (1989)
   Name: movie_name, dtype: object, 1284  Butch Cassidy and the Sundance Kid (1969)
   Name: movie_name, dtype: object, 900   Casablanca (1942)
   Name: movie_name, dtype: object, 1108  On Golden Pond (1981)
   Name: movie_name, dtype: object, 1284  Butch Cassidy and the Sundance Kid (1969)
   Name: movie_name, dtype: object, 1284  Butch Cassidy and the Sundance Kid (1969)
   Name: movie_name, dtype: object, 900   Casablanca (1942)
   Name: movie_name, dtype: object, 900   Casablanca (1942)
   Name: movie_name, dtype: object, 900   Casablanca (1942)
   Name: movie_name, dtype: object]
    
```

Fig. 10 Item based collaborative filtering recommended movies

	0	1	2	3	4	5	6	7	8	9	...	5830
0	0.000000	0.000000	0.635001	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.164488	...	0.204124
1	0.000000	0.000000	0.233646	0.334497	0.081731	0.215694	0.418121	0.447076	0.258573	0.316369	...	0.085349
2	0.635001	0.233646	0.000000	0.508001	0.000000	0.208457	0.000000	0.346144	0.336596	0.296638	...	0.129619
3	0.000000	0.334497	0.508001	0.000000	0.000000	0.175863	0.000000	0.419314	0.441726	0.131590	...	0.000000
4	0.000000	0.081731	0.000000	0.000000	0.000000	0.000000	0.000000	0.115262	0.000000	0.124591	...	0.099751
...
5835	0.000000	0.470157	0.317345	0.624695	0.248036	0.109861	0.000000	0.261943	0.275944	0.246611	...	0.127515
5836	0.218218	0.323907	0.318708	0.272772	0.000000	0.227061	0.000000	0.343132	0.192785	0.367916	...	0.137343
5837	0.000000	0.352260	0.114129	0.112331	0.150958	0.161332	0.000000	0.306163	0.272908	0.439756	...	0.370694
5838	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	...	0.000000

Fig. 11 User similarity matrix

User Based Filtering

User based filtering method forms similarity matrix between users. The users which have greater similarity with our user is taken and those movies watched by these users are recommended. Figure 11 shows the item similarity matrix, each movie is represented in a row and the different movies are presented in the column. Figure 13 shows the results of the User CF.

Figure 12 shows user based collaborative filtering code. Figure 13 shows user based collaborative filtering recommended movies list.

3.2.2 Model Based Collaborative Filtering

Singular Value Decomposition

Model based evaluation in collaborative filtering uses the SVD algorithm. The algorithm is applied after the formulation of a distinct data frame with required columns.

```

movieId_recommended=list()
def getRecommendedMoviesAsperUserSimilarity(userId):
    """
    Recommending movies which user hasn't watched as per User Similarity
    :param user_id: user_id to whom movie needs to be recommended
    :return: movieIds to user
    """
    user2Movies= df4[df4['UserID']== userId]['movie_id']
    sim_user=df_similar_user.iloc[0,0]
    df_recommended=pd.DataFrame(columns=['movie_id','movie_name','tag','colour','UserID','Ratings','Timestamp'])
    for movieId in df4[df4['UserID']== sim_user]['movie_id']:
        if movieId not in user2Movies:
            df_new= merged[(merged.UserID==sim_user) & (merged.movie_id==movieId)]
            df_recommended=pd.concat([df_recommended,df_new])
    best10=df_recommended.sort_values(['Ratings'], ascending = False )[1:10]
    return best10['movie_id']

```

Fig. 12 User based collaborative filtering code

```

[1287  When Harry Met Sally... (1989)
Name: movie_name, dtype: object, 2327  Shakespeare in Love (1998)
Name: movie_name, dtype: object, 2039  L.A. Story (1991)
Name: movie_name, dtype: object, 3466  American Psycho (2000)
Name: movie_name, dtype: object, 1192  Star Wars: Episode VI - Return of the Jedi (1983)
Name: movie_name, dtype: object, 2647  Ghostbusters (1984)
Name: movie_name, dtype: object, 221   Don Juan DeMarco (1995)
Name: movie_name, dtype: object, 3289  Defending Your Life (1991)
Name: movie_name, dtype: object, 352   Forrest Gump (1994)
Name: movie_name, dtype: object]

```

Fig. 13 User based collaborative filtering recommended movies

SVD is a dimensionality reduction algorithm. For our purpose, we are creating a data-frame with user_id, movie_id and user_ratings. A User-Movie ratings matrix is formed and Matrix factorization is done [8].

The matrix is demeaned by removing user ratings mean value from the user_rating matrix to remove bias.

SVD decomposes the matrix into three other matrices:

```

U, sigma, Vt = svds(R_demeaned, k = 50)
sigma = np.diag(sigma)
all_user_predicted_ratings = np.dot(np.dot(U, sigma), Vt) +
user_ratings_mean.reshape(-1, 1)

```

These matrices are further used in the prediction of movies which can be watched by the user.

3.2.3 Model Evaluation

The model is evaluated by passing the user_id to both SVD and both memory based CF models. The total numbers of hits are calculated by checking how many movies are being recommended in common to the both memory and model based CF'S by the total number of the recommendations. Figure 14 shows the hit ratio percentage of User-SVD recommended movies.

Figure 15 shows the evaluation of collaborative filtering model. Figure 16 shows the collaborative filtering model evaluation results.

```
In [47]: predictions
```

```
Out[47]:
```

	movie_id	movie_name	tag	MovieID
412	3363	American Graffiti (1973)	Comedy Drama	3363.0
163	1259	Stand by Me (1986)	Adventure Comedy Drama	1259.0
149	1208	Apocalypse Now (1979)	Drama War	1208.0
106	912	Casablanca (1942)	Drama Romance War	912.0
174	1304	Butch Cassidy and the Sundance Kid (1969)	Action Comedy Western	1304.0

Fig. 14 SVD results

```
user_id=user
def evaluation_collaborative_svd_model(userId,userOrItem):
    movieIdsList= list()
    movieRatingList=list()
    movieIdRating= pd.DataFrame(columns=['MovieID','Ratings'])
    movieIdsList=recommendedMoviesAsperItemSimilarity(user_id)
    for movieId in movieIdsList:
        predict = svd.predict(userId, movieId)
        movieRatingList.append([movieId,predict.est])
    movieIdRating = pd.DataFrame(np.array(movieRatingList), columns=['MovieID','Ratings'])
    count=movieIdRating[(movieIdRating['Ratings']>=3)]['MovieID'].count()
    total=movieIdRating.shape[0]
    hits= count/total
    hey=hits
    return hey
```

Fig. 15 Collaborative filtering model evaluation

```
In [54]: print("Hit ratio of User-user collaborative filtering")
print(evaluation_collaborative_svd_model(user_id,True))
print("Hit ratio of Item-Item collaborative filtering")
print(evaluation_collaborative_svd_model(user_id,False))
```

```
Hit ratio of User-user collaborative filtering
0.7777777777777778
```

Fig. 16 Collaborative filtering model evaluation results

3.3 Content Based Filtering

Content based filtering endures to speculate the conduct or structure of the users when the features of the items are provided. The user preferred feature vectors and the item feature vectors from the previous results are used to calculate the similarity matrix at recommendations. Thus, recommendations are made for the top few. Content based filtering is not dependent on user information prior interest in making recommendations. The recommended top few movies from the basis for content-based recommendation.

for $i = 1$ to $i = n$:

for $j = 1$ to $j = n$:

if $i = j$: item[i][j] = 1

else :

$$\begin{aligned} \text{item}[i][j] = & 1/n * (\text{distinct genre})/(\text{total count of genre}) \\ & + 1/n * (\text{distinct title})/(\text{total count of title}) \\ & + 1/n * (\text{distinct emotion})/(\text{total count of emotion}) \end{aligned} \quad (2)$$

In our system, we have taken the genres as the feature for performing content based filtering. TF-IDF is performed on these genres and is divided into various groups. Since we are not taking any movie input from the user, we have taken the output of the collaborative filtering model as the input. Here the genre of each input movie is passed into the function and those movies are recommended which are in the same group as that of the input movie genre and have greater dot product of similarities.

Figure 17 shows the content based filtering code and Fig. 18 shows the content based filtering recommended movies.

Following are the movies recommended in CBF

3.3.1 Model Evaluation with K-nearest Neighbours

Content based model evaluation uses KNN to find important insight into the similarities between what the user has watched and the particular movies predicted. Figure 19 shows the KNN code. The KNN algorithm is used for clustering movies after predicting them through the content-based filtering process. The function returned by genre. If the movie passed matches the genre that is returned by a function, then

```

tfidf_movies_genres = TfidfVectorizer(token_pattern = '[a-zA-Z0-9\-\_]+')
df['tag'] = df['tag'].replace(to_replace="no genres listed", value="")
tfidf_movies_genres_matrix = tfidf_movies_genres.fit_transform(df['tag'])
cosine_sim_movies = linear_kernel(tfidf_movies_genres_matrix, tfidf_movies_genres_matrix)

[ ] def get_recommendations_based_on_genres(movie_title, cosine_sim_movies=cosine_sim_movies):
    # Get the index of the movie that matches the title
    idx_movie = df.loc[df['movie_name'].isin([movie_title])]
    idx_movie = idx_movie.index
    sim_scores_movies = list(enumerate(cosine_sim_movies[idx_movie][0]))
    # Sort the movies based on the similarity scores
    sim_scores_movies = sorted(sim_scores_movies, key=lambda x: x[1], reverse=True)
    sim_scores_movies = sim_scores_movies[1:6]
    movie_indices = [i[0] for i in sim_scores_movies]
    # Return the top most similar movies
    return df['movie_name'].iloc[movie_indices]

```

Fig. 17 Content based filtering code

```

{'Aladdin and the King of Thieves (1996)',
 'American Tail, An (1986)',
 'American Tail: Fievel Goes West, An (1991)',
 'Bad Boys (1995)',
 'Big Bully (1996)',
 'Bio-Dome (1996)',
 "Bug's Life, A (1998)",
 'Carrington (1995)',
 'Corrina, Corrina (1994)',
 'Don Juan DeMarco (1995)',
 "Don't Be a Menace to South Central While Drinking Your Juice in the Hood (1996)",
 'Drop Zone (1994)',

```

Fig. 18 Content based filtering recommended movies

it is regarded as a hit and vice versa (counted a fault). Our model shows 93.3% of the hit count and provides a fault count of 6.6%. This shows that most of the movies have been clearly classified to correct genres as shown in the Fig. 20.

3.4 Hybridization

Different multiple recommendation techniques are applied to determine desired output in the hybrid recommendation system. Compared to individual content based or collaborative filtering, the accuracy is high in hybrid recommendation systems. In our work we have taken the results obtained by collaborative filtering, given those results as input for content-based filtering. The results obtained from content based filtering are taken and each movie of the output is given as the input for the SVD model of collaborative filtering. The movies which has higher predicted values from SVD

```
[ ] def get_movie_label(movie_id):
    classifier = KNeighborsClassifier(n_neighbors=5)
    x= tfidf_movies_genres_matrix
    y = df.iloc[:, -1]
    classifier.fit(x, y)
    y_pred = classifier.predict(tfidf_movies_genres_matrix[movie_id])
    return y_pred
```

```
▶ true_count = 0
false_count = 0
def evaluate_content_based_model():
    for key, columns in df.iterrows():
        movies_recommended_by_model = get_recommendations_based_on_genres(columns["movie_name"])
        predicted_genres = get_movie_label(movies_recommended_by_model.index)
        for predicted_genre in predicted_genres:
            global true_count, false_count
            if predicted_genre == columns["tag"]:
                true_count = true_count+1
            else:
                false_count = false_count +1
    evaluate_content_based_model()
total = true_count + false_count
```

Fig. 19 K-nearest neighbours code


 Hit:0.9338655678599022
Fault:0.06613443214009786

Fig. 20 Accuracy rate of the model using content-based filtering

model is recommended to the user. Figure 21 shows remmended movies after the hybridization process. Our propose work uses the deduced and perceived similarities to recommend movies.

Figure 22 shows the recommended movies after the hybridization process.

```
df_movies=df2
def hybrid_content_svd_model(userId):
    recommended_movies_by_content_model = get_recommendation_content_model(userId)
    recommended_movies_by_content_model = df_movies[df_movies.apply(lambda movie: movie["movie_name"]
                                                                    in recommended_movies_by_content_model, axis=1)]
    for key, columns in recommended_movies_by_content_model.iterrows():
        predict = svd.predict(userId, columns["movie_id"])
        recommended_movies_by_content_model.loc[key, "svd_rating"] = predict.est
    rec=recommended_movies_by_content_model.sort_values("svd_rating", ascending=False).iloc[0:11]
    hybrid_content_svd_model(userId)
```

Fig. 21 Hybridisation code

Fig. 22 Recommended movies after the hybridization process

Movie Name
Sunset Blvd. (a.k.a. Sunset Boulevard) (1950)
Dr. Strangelove or: How I Learned to Stop Worrying and Love the Bomb (1963)
Postino, Il (The Postman) (1994)
In the Heat of the Night (1967)
Star Wars: Episode V - The Empire Strikes Back (1980)
Boat, The (Das Boot) (1981)
Godfather: Part II, The (1974)
Spellbound (1945)
Cyrano de Bergerac (1990)
Adventures of Robin Hood, The (1938)
Diva (1981)

4 Conclusion and Future Work

The emotion-based movie recommender system is a useful concept in coming up with recommendable movies for target users. It is particularly instrumental in finding out the various triggers to a person taste in movies including the motivation to choose specific genre of a movie. This tactic is also successful in determining specific movies that corelate with the user’s emotional states and defining user’s emotional states when variables including color are provided as input. Emotions relate to the colors including happiness, angry, sad. Emotion-based movie recommender system applies Collaborative filtering to automatically predict user interests by gathering preferences and tastes from several users. Also, it applies the user-based filtration to identify one’s taste of product and recommend it to another user, while the Content-Based recommender system endures to speculate the conduct or structures of the users when the features of the items are provided; positive reactions. Sequentially, movie recommendations based on one’s emotional trajectory is key as it allows them to map movie recommendation based on their emotional states. In the future, we aim on including a wider range of colors for the users to choose from during the color input phase. This enables our system to provide even better recommendations.

References

1. K. Topal, G. Ozsoyoglu, Movie review analysis: emotion analysis of IMDb movie reviews, in *MSNDS 2016: The 7th International Workshop on Mining and Analyzing Social Networks for Decision Support* (2016)
2. K.N. Jain, V. Kumar, P. Kumar, T. Choudhary, Movie recommendation system: hybrid information filtering system. *Adv. Intell. Syst. Comput.* (2018)
3. P. Sharma, L. Yadav, Movie recommendation system using item based collaborative filtering. *Int. J. Innov. Res. Comput. Sci. Technol. (IJRCST)* **8**(4) (2020)
4. V. Subramaniaswamy, R. Logesh, M. Chandrashekar, A. Challa, V. Varadarajan, A personalised movie recommendation system based on collaborative filtering. *Int. J. High Perform. Comput. Network.* (2017)
5. D. Roy, A. Kundu, Design of movie recommendation system by means of collaborative filtering. *Int. J. Emerg. Technol. Adv. Eng.* **3**(4) (2013)
6. S.R.S. Reddy, S. Nalluri, S. Kuniseti, S. Ashok, B. Venkatesh, Content-based movie recommendation system using genre correlation, in *Smart Innovation, Systems and Technologies*, Singapore (2018)
7. M.F. Aljunid, D.H. Manjaiah, Movie recommender system based on collaborative filtering using apache spark. *Adv. Intell. Syst. Comput.* (2018)
8. S.T. Soman, V.J. Soumya, K.P. Soman, Singular value decomposition a classroom approach. *Int. J. Recent Trends Eng.* **1**(2) (2009)
9. G.R. Ramya, P.B. Sivakumar, An incremental learning temporal influence model for identifying topical influencers on twitter dataset. *Soc. Netw. Anal. Min.* **11**(1) (2020)
10. S.R. Mugunthan, T. Vijayakumar, Design of improved version of sigmoidal function with biases for classification task in ELM domain. *J. Soft Comput. Paradigm (JSCP)* **3**(02), 70–82 (2021)
11. S. Manoharan, Early diagnosis of lung cancer with probability of malignancy calculation and automatic segmentation of lung CT scan images. *J. Innov. Image Process. (JIIP)* **2**(04), 175–186 (2020)

Global Positioning System (GPS) and Internet of Things (IOT) Based Vehicle Tracking System



V. Baby Shalini

Abstract Vehicle Tracking Technique is a practice that associates the utilization of automaton vehicle locality in singular vehicles with software that automatically determines the geographic location of the vehicle and then transmits the data to the end-user. Present-day, GPS technology was assimilated on vehicle tracking systems for discovering the vehicle locale, however different sorts of automatic vehicle locality innovation can likewise be utilized. In this paper, a vehicle tracking system using GPS and NodeMCU is undertaken to enable users to locate their vehicles at any time with ease and in a convenient manner. The proposed system utilizes GPS that practices satellite technology for navigation and will unremittingly provide statistics like longitude, latitude, velocity, distance traveled, etc. of a vehicle from a remote place to NodeMCU, user's phone as well as send this real-time data to IOT Cloud platform. When a user sends an SMS request, the system responds by sending an SMS to the user's mobile phone with the vehicle's location, and the same information is also stored on Thing Speak. Therefore, advances in technologies and the availability of economical open-source hardware systems are setting a new trend in system designing. The proposed system has been designed using both computer software and hardware, results have been obtained which showed accuracy in positioning and fast response to user commands.

Keywords Automaton · Geographic · Locale · Navigation · Remote

1 Introduction

Vehicle Tracking Scheme is an innovation primarily used to decide the location of a vehicle and about. Present-day, this scheme is using GPS technology [1] to oversee and discover the vehicle anyplace on earth. A GPS receiver [2] estimates the time requisite by radio signs to go from no less than atleast four satellites to its area, it then, at that point figures it computes the vehicle's longitude, latitude, and height based

V. Baby Shalini (✉)

Department of Information Technology, Kalasalingam Academy of Research and Education, Krishnankoil, India

e-mail: v.babyshalini@klu.ac.in

on its separation from each satellite. A NodeMCU is utilized for the conveyance of the locality of the vehicle from a far-off region to the user's mobile phone who has demanded it and likewise the same data is also stowed on cloud database for subsequent use.

Vehicle Tracking [3] can be classified into dual classes, based on data communique: passive and active tracking. Passive tracking otherwise called GPS loggers which accumulate a vehicle's locale information and then stockpile it on a hard drive or memory card, which we would then be capable of accessing sometime in the future. To usance a passive device, we just spot it in or on the vehicle. At the point, when you requisite to acquire the data procured by the tracker, just addon the device to a PC. This device incorporates auto-downloads that transference data through radio. After the information has been obtained, then it will be revealed as a summary statement over an interactive mapping software or with Google Earth, letting the user recognize every locale the vehicle passed on, how much time the vehicle stayed at each stop, and then some. With this tracking, users just neediness to spend money one time upon purchase (incorporate hardware, software, and database) and installation process of the device. These systems don't furnish real-time pursuing data, making it pointless amid crises like road accidents or burglary. In any case, this issue can be handily settled by introducing extra models on the vehicle for an extra outlay. These add-on tools permit the framework to keep watch over vehicles lively, which thus empowers the fleet managers to recognize the vehicle's speed and the location at a random time [4]. Active tracking amasses similar data as passive tracking systems however real-time transmission of observed data to a PC through cellular or satellite networks or data center for estimation. But this system is more complicated and expensive when contrasting to the passive device and has many benefits brought by its functions. When an active tracker is positioned on a vehicle, the user will be capable to scrutinize the vehicle locality, entire period the vehicle holds on a specific stop, speed, and more tracking details from our home or office[5] because an active GPS tracker emanates with a reliable interface, therefore, the user will be able to keep track of anything hastily and proficiently. GPS trackers that send real-time data consistently are known as active trackers, while those that don't make a record of real-time tracking are considered passive trackers. Occasionally, modern tracking devices integrate both tracking proficiencies [6].

Automatic Vehicle Location (AVL), Assisted Global Positioning System (AGPS), Radio Frequency Identification (RFID) are some sorts of GPS vehicle tracking which usages active device. AVL [7] system is a sophisticated tracking scheme in which GPS and Geographic Information System (GIS) are used to determine the vehicle's true geographic location. Every 60 s, vehicle locality, speed, direction, mileage, and vehicle status being transmissible to the base station and will be exhibited on a computerized map. This system furnishes the locale of the vehicle with a precision of about 5 m to 10 m. If the AVL system is used for tracing a vehicle, then at that point, the average expense per vehicle is \$1 to \$2 each day. The system has constraints like users can't get precise, whole, and sufficient satellite data from densely populated urban locations or indoor, and where communication is hampered by natural obstacles or numerous buildings.

To upgrade the performance of GPS Receivers, AGPS framework utilizes terrestrial RF network since it provides GPS receivers with information about the satellite constellation directly. Mobiles and cellular networks are employed by AGPS to discover precise positioning data. Here, GPS satellites intend to discern the vehicle locale incessantly. Four satellites are connected to a GPS receiver in a vehicle (latitude, longitude, and elevation are determined by 3 satellites and the fourth one gives an element of time). As a result, it never fails to show a vehicle's whereabouts. The vehicle's location is provided with a precision of 3 m to 8 m and a speed of 1 km. While the vehicle is in mobility, data such as the location of the vehicle, its average speed, way navigated and cautions are continuously consigned from GPS device to the base station after every 10 s and handed off as an SMS utilizing the cell phone of the client node to the base station. This system is providing uninterrupted renovation about the vehicle location updates, therefore, when contrasting to AVL system, this system is more overpriced. On the off chance that the user necessities keep informed after every 10 and 5 s, therefore the subscription is \$1.33, \$1.67 per day per vehicle respectively. The system has few restraints, GSM is employed for disseminating the recognized data from a vehicle to the base station, therefore the expenditure of conveyance SMS is the foremost worry to be thought of.

RFID [8] is an automaton identification scheme in which radio waves are utilized for acquiring vehicle location. RFID [9, 10] includes three parts: tag, RF reader, and software. Tag with microelectronic circuits propels the vehicle information to a far-flung RFID reader, a transceiver coordinated by a microprocessor or digital signal processor (DSP), antenna enclosed with RFID reader retrieves data from RFID tags through software. This system furnishes the locality of the vehicle with an exactness of 4 m to 6 m. Here, information such as the location of the vehicle, mileage, and speed are distributed every one minute to the base station.

Nowadays, notwithstanding, with innovation evolving at a high speed, to trace and exposition the vehicle locations in real-time, automated vehicle tracking system is being operated. This paper put forward a vehicle tracking system employing GPS, NodeMCU, smartphones, and IOT to afford better service and lucrative solutions for users. In these times, a GPS navigation system is a space-based scheme extensively espoused in vehicles today that confers vehicle position wherever on or near the earth where there is an unhindered range of view to four or more GPS satellites. Global Navigation Satellite System (GNSS) network is utilized for tracking the locale of the device. The network is made up of a range of satellites that utilizes microwave signals to transport the data which will be gotten by the GPS receiver module. The GPS satellite constellation comprises 24 earth orbiting GPS satellites which transference three pieces of information – 1) the satellite's number 2) its position in space 3) the time the data is thrown. With signals from at least three satellites, a GPS receiver can notice vehicle location on the ground (i.e., longitude and latitude). NodeMCU is an open-source programming and equipment development setting that is fostered around a reasonable System on a Chip (SoC) called the ESP8266 which is having all the decisive components of the modern computer like processor, memory unit, networking (Wi-Fi), and even a modern operating system. The Internet of Things [11, 12] is a fabulous technology where devices being allied to the internet and all sorts of

data exchange are also probable. In this paper, IOT based vehicle tracking system is exploited for tracing the vehicle location. Smartphone's impact on society continues to grow as people become more accustomed to them and utilise them in their daily lives. The straightforward and modest Short Message Service (SMS) permits users to frontward up to 160 characters. In this way, SMS is more adequate for propelling the location information to the user's mobile phone with high consistency.

The rest of the paper is organized into four different sections. Section 2 covers various previous vehicle detection systems and their drawbacks. Section 3 explains the proposed method and Sect. 4 includes results and discussion. Section 5 summarises the review of the work.

2 Related Work

Pham et al. [13] explained the proposed vehicle tracking system which consists of main hardware modules like U-BLOX NEO-6Q GPS receiver, U-BLOX LEON-G100 GSM Module, and ARDUINO UNO microcontroller which have been fixed in the vehicle. Here, to acquire a vehicle's coordinate, GPS is utilized and then this information is passed to a microcontroller which in turn process the information and sent it to the LEON-G100 GSM and then forwarded it to a mobile network. The communication between GPS/GSM module and microcontroller is done by UART (Universal Asynchronous Receiver and Transmitter). The pitfall here is the reliability of the system can be improved and additional features can also be implemented.

Muhammad et al. [14] explained about vehicle monitoring scheme to avoid theft. The GSM modem receives the location of the vehicle details from GPS and it transmits the received data to the user through SMS and can be displayed on Google Maps or lay on a site so it tends to be gotten too distantly through the web. This method is used in many applications and it can also be implemented by a transmitter that transmits the location data to the receiver and also can be integrated into a compact box for convenience. There are as yet a few enhancements that can be completed later on. For instance, the framework can be tried with a transmitter that can propel the location facet in real-time.

Seok et al. [15] elucidated about vehicle tracking system which uses GPS/GPRS, GSM technologies, server, and a smartphone application. GSM and GPS are responsible for connecting the vehicle and the server and the connection is done by TCP/IP. Likewise, the communication between the vehicle and server is done by HTTP. The server stores the data about the vehicle location in the database. To display the current locale of the vehicle, a smartphone application is developed which shows the location of the tracked vehicle and by updating at regular intervals with calculated distance and time. GPS continuously tracks the location of the vehicle and the data is passed from the GPS receiver to the microcontroller ARM7. The data is also collected by multiple sensors (eg) Temperature sensor, eye blink sensor, alcohol sensor. Due to RTOS programming, the sensor also detects the real-time events and sent the data to the server or owner, ex if the fuel level of the vehicle is low, then it gives the nearest

location of the petrol pump to the driver as well as to the owner and also in case of an accident, it sends the information to the nearest hospital. The data are displayed by using GUI.

Prashant et al. [16, 17] elaborated about a system for monitoring the school bus which uses an Embedded UNIX board with Raspberry PI, GSM, and GPS/GPRS. The GPS is responsible for tracking the locality of the vehicle. The GPRS is responsible to send the latitude and longitude value of the tracked vehicle location to the board. GSM is responsible to communicate the perceived data to the server and board. This method also monitors the current path and speed limit of the vehicle. When the location or speed data are mismatched with the specified stored data, then the warning alert will be sent to the owner as well as the driver. It also uses a temperature sensor and gas leakage sensor for safety purposes.

Fatin et al. [18] explained about vehicle tracking device (VTD) which consist of ARDUINO UNO and GSM Module which is fixed in the vehicle in a hidden place. The VTD receives the latitude and longitude information from GPS. When the user requests the VTD for the location information, the VTD sent the latitude and longitude details to the user through SMS with the help of the GSM module.

Omar et al. [19] elucidated a framework for tracking systems based on ARDUINO INTEL GALILEO Board. This ARDUINO Board is fixed in the vehicle, GPS gives the location of the vehicle to the ARDUINO Board. The GSM is being used to commune the data's from the ARDUINO board to the webserver or user. The web server displays the data like location, speed, time, data, etc. on the google maps application. It also sends the data to vehicle owners through SMS.

Rahul et al. [20] elaborated on the vehicle tracking process with IOT environment and RFID technology. RFID tracks the vehicle with help of radio waves, RFID tags are fixed in the vehicle and RFID readers are fixed at several nodes with equal distance inroads. The RFID emits a signal to activate the RFID tag for a particular vehicle, therefore it can read the ID of a particular vehicle and it collects the data and sent it to the Internet whereas it is directly connected with nodes using WAN connections. Therefore, using the internet, the user collects the data of a particular vehicle.

Mayuresh et al. [21] explained a system that uses IOT for data collection and analysis in order to track the vehicle location and parameter. The location of the vehicle is given by GPS and send to the ARDUINO controller and then the controller transmits the data to the web server by using GSM Module. The web server displays the data in google maps through the internet. In this method, vehicle parameters are also identified with the help of sensors, ex, speed, temperature, fuel sensors are used for data analysis of vehicle parameters. By various testing of parameters, this method can be developed in the future.

Bhavana et al. [22] elucidated that the proposed system is used to track vehicles like school buses, college buses, etc. In this method, embedded units, namely, Raspberry Pi technology is inserted into the device. GPS gives the location of a vehicle and sends it to the server. The vehicle owner will receive a message alert, as well as an audible alert for the driver, if vehicle deviates from its intended path. It is also used for a variety of features like speed limit, fuel level, etc. The data's are displayed in the google map application.

Edward et al. [23] explained about vehicle monitoring system with the help of RFID concepts. The RFID tag is fixed in the vehicle and RFID Reader is fixed in the particular entrance or exit. Now, when the vehicle enters a particular area, the RFID checks whether it is registered or not. If it is registered, the gate will open using a servomotor and the green LED glows with a buzzer sound and also captures the image of the vehicle and communes the data to the database or user using GSM. On the other hand, if the vehicle is not registered, the gate will not open but the image of the vehicle will be captured, still, if the vehicle wants to enter the particular area, it will be allowed through the bypass and the gate will be opened with RED LED light and buzzer sound. This data also forwarded to the database or server using GSM Module.

Abdullah et al. [24] explained a method done by two software like Thing Speak and freeboard. The location of the vehicle is tracked by using GPS. This information is sent to the ARDUINO UNO controller and also send to the Thing Speak through GSM. The Thing Speak website displays the location data in the form of a chart and forwards the data to the freeboard website, it analysis the data and displays the data in the form of a map. This is assessable to all users.

In all existing methods, ARDUINO UNO and Raspberry Pi microcontroller are used to collect the location of a vehicle in real-time but when compared to ARDUINO UNO microcontroller and Raspberry Pi microcontroller, NodeMCU microcontroller has additional features like low cost, Wi-Fi connectivity is attached. Therefore, in the proposed method instead of GSM modem and ARDUINO UNO, NodeMCU is utilized.

3 Methodology

The proposed method vehicle tracking system is shown in Fig. 1.

The components utilized for developing the proposed scheme are GPS Receiver, NodeMCU, ThingSpeak. Here, GPS is utilized to detect the location of vehicles anywhere on a road and then forward the perceived information to the NodeMCU module, and the same information is displayed on an LCD and then loaded to ThingSpeak for further reference. If an administrator wants to collect specific information about a location of a particular vehicle means, then the request is forwarded to NodeMCU module which in turn the data is retrieved from there and transfer to administrator mobile phone.

NEO6M GPS Receiver

GPS module depends on the u-blox NEO-6 M GPS engine. This module embraces a configurable UART interface for serial communicate and a $25 \times 25 \times 4$ mm ceramic antenna, which affords a robust satellite pursuit competence.

In this proposed method, the GPS module monitors the location of the vehicle continuously and delivers NMEA-formatted real-time tracking position data which starts with string, global positioning system fix data, comma, GMT Time, latitude,

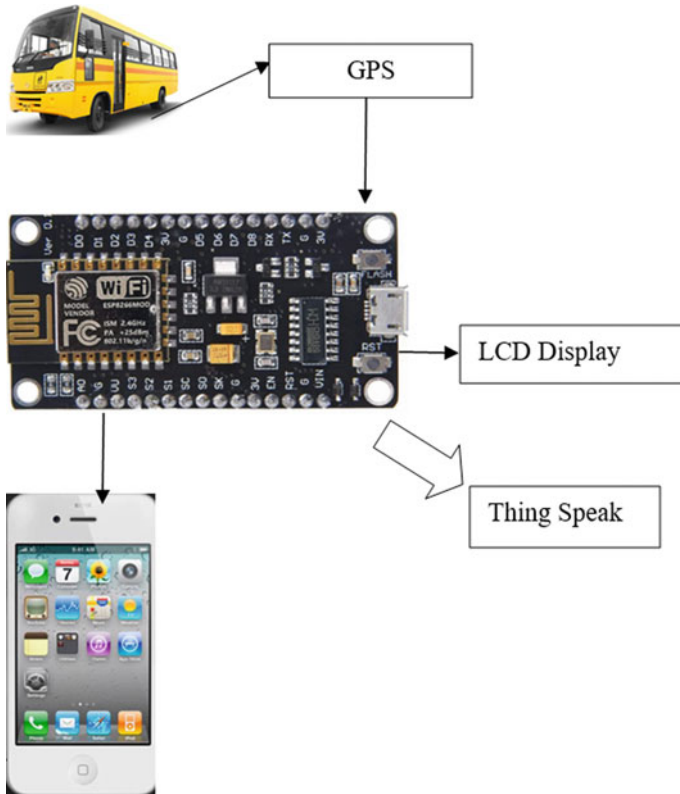


Fig. 1 System Architecture

longitude, fix quantity, number of satellites viewed, HDOP, Altitude, geoids height, checksum. Latitude and longitude values are extracted to identify the vehicle's location or, where it is located.

ESP8266 NodeMCU

The ESP8266 is a node cum microcontroller chip with WI-FI connections premeditatedly designed for IOT applications by Espressif systems. The Node MCU has 30 pins in total, Micro USB, 3.3 V, GND, VIN, EN, RST, Analog Pin A0,16 General Purpose Input Output Pin (GPIO 1-GPIO 16), SD1, SDO, CLK, CMD, 2 UART Interfaces, and one I2C Pin. Here, NodeMCU can connect with vehicles and let data transfer using the Wi-Fi protocol.

To interface GPS Receiver with NodeMCU, connect VCC pin of NEO6M GPS Receiver with 3.3 V pin of ESP8266 NodeMCU. TX and RX pin of NEO6M GPS Receiver is connected with D2 and D1 pin of ESP8266 Node M CU. GND pin of NEO6M GPS Receiver is connected with GND pin of ESP8266 Node MCU.

Liquid Crystal display (LCD)

LCD is a flat panel display those usages liquid crystals in their prime form of activity. Liquid crystals don't exude light straightforwardly, rather employing a backlight or reflector to create images in color or monochrome. LCDs are to exhibit random or immobile images, which can be exposed or concealed. All the information about the location of vehicles is displayed on LCD Display.

To connect NodeMCU with LCD display,

1. Connect the GND pin on the LCD display to one of the GND pins on the NodeMCU.
2. Connect the VCC pin on the LCD display to the VIN pin on the NodeMCU.
3. Connect the SDA and SDL pin on the LCD display to the D2 and D1 pin on the Node MCU.

Thing Speak

Things are either sensors or actuators. A sensor tells us something about the environment. Actuators are something that you want to control. The IOT brings everything together and allows us to interact with things. Thing Speak is an application platform for IOT. This permits us to assemble an application around statistics congregated by sensors. At the core of this platform is a Thing Speak Channel which is the place where we forward our statistics to be deposited. Each channel incorporates 8 fields for information, 3 location fields, and 1 status field. When we have a Thing Speak Channel, we can distribute information to the channel, have Thing Speak assess the information, and afterward have your application recover the information.

4 Result and Discussion

The proposed method does two things.

1. Track the real-time location of the vehicles
2. Location History (ie past location that the vehicles have traversed). Using the Thing Speak IOT Platform, the location history of a vehicle is monitored.

The results and discussion of a vehicle tracking system are presented in this section where Fig. 2. shows the experimental setup. The proposed system consists of GPS Receiver, NodeMCU connected with GPS Receiver, LCD Display, and Thing Speak.

GPS Receiver module with inbuilt ceramic container which receives the GPS coordinates (Latitude and Longitude) from the satellites and then these two data's are sent to NodeMCU. NodeMCU is a Wi-Fi module that helps us to connect with the Thing Speak cloud process the data's and gives it to Thing Speak Cloud. With the help of Thing Speak Cloud, monitoring is done by the end user. Here LCD is used to show the IP address with which the internet is connected.

Set up the IoT platform (ThingSpeak) to store GPS coordinates on the cloud and graphically visualise the GPS data after completing the hardware as shown in the

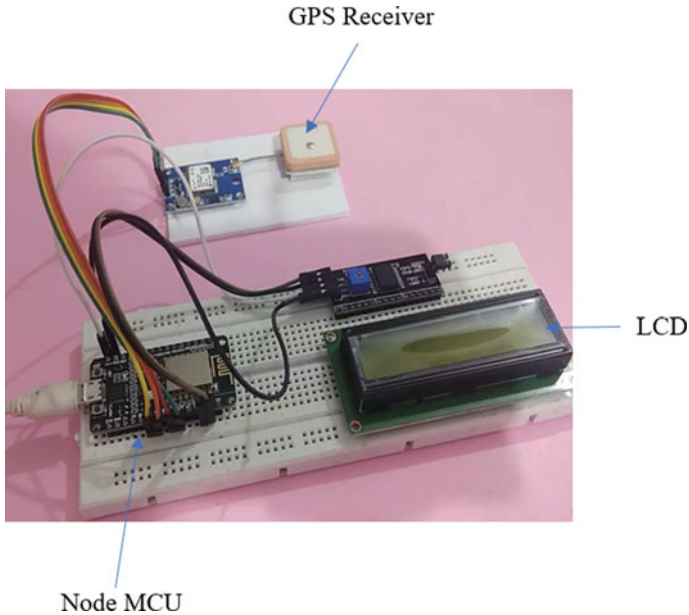


Fig. 2 Experimental Setup

above circuit diagram. To do so, go to the ThingSpeak website, establish a new account if you don't already have one, or check in with your existing credentials, and then click New Channel. Now fill up the details and create two field names such as latitude and longitude and then click on the Save channel. Select the created channel and then record the channel ID and API key. Once hardware connections and ThingSpeak setup are done, program the ESP8266 NodeMCU.

In ThingSpeak platform, two charts as shown in Fig. 3 are there: one chart as in Fig. 4 is for displaying latitude value and another chart as in Fig. 5 is for displaying longitude values. With the help of latitude and longitude values, real time location of a vehicle is monitored.

5 Conclusion

NodeMCU has been planned and tried effectively. A NodeMCU module is interfaced with a GPS receiver to acquire the latitude and longitude position of a vehicle. To be an efficient tracking scheme, the information endowed by this system is in real-time. Therefore, the NodeMCU module is connected with the Thing Speak IOT platform and the perceived data is also stored on Thing Speak via Wi-Fi module. By having GPS coordinates info, the locale of a vehicle can be discovered by the users anytime

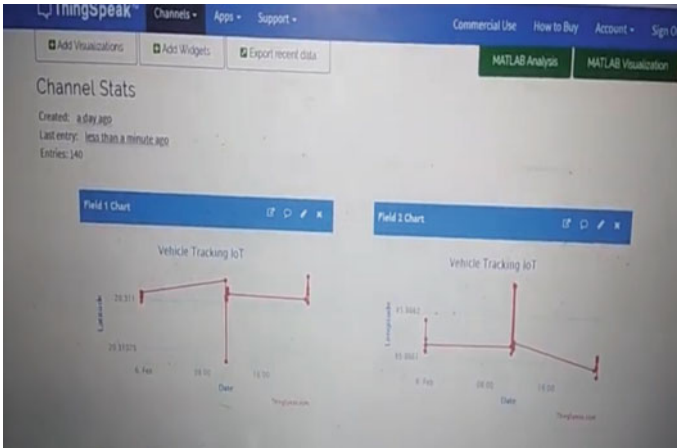


Fig. 3 Latitude and Longitude Charts

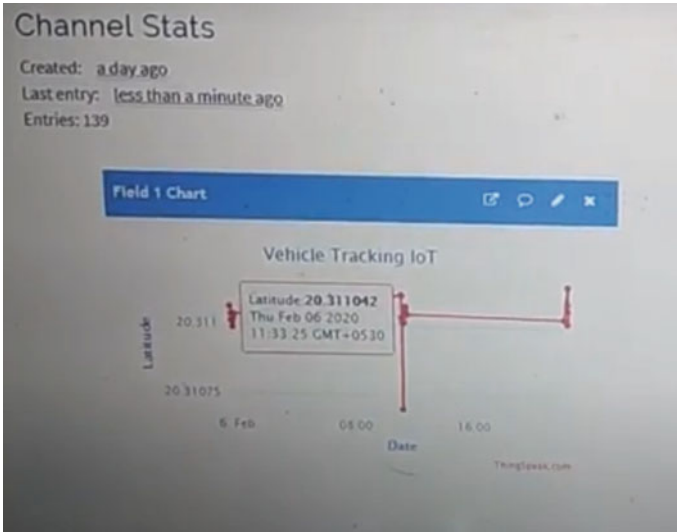


Fig.4 Latitude Chart

and anywhere. Therefore, the proposed strategy assumes a significant role in real-time observing of a vehicle and renovating real-time data on the server-side after a certain timespan. IoT an open-source platform makes this proposed method very dynamic, efficient, and cost-effective. The main problem here is GPS Receiver tracks the vehicle locale 24/7 as a result of which consumes more energy. The future scope is to detect accidents or theft of the vehicle by some sensors.

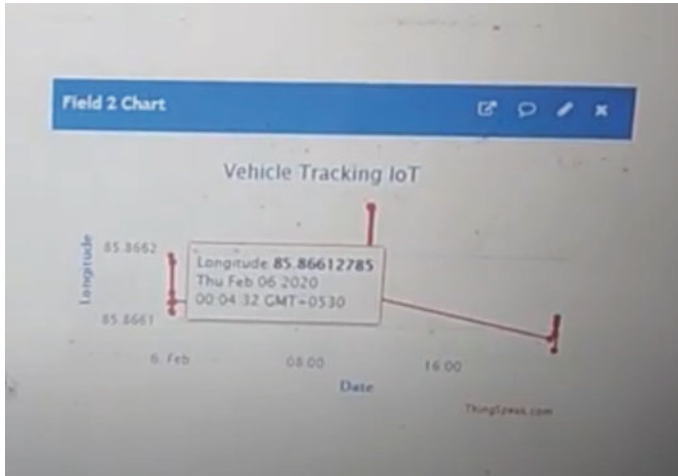


Fig. 5 Longitude Chart

References

1. A.S. Dinker, S.A. Shaikh, Design and implementation of vehicle tracking system using GPS. *J. Inf. Eng. Appl.* **1**(3) (2011)
2. I.M. Almomani, N.Y. Alkhalil, E.M. Ahmad, R.M. Jodeh, Ubiquitous GPS vehicle tracking and management system, in *IEEE Conference on Applied Electrical Engineering and Computing Technology (AEECT)* (2011)
3. T.M. Bojan, U.R. Kumar, V.M. Bojan, *Designing Vehicle Tracking System—An Open Source Approach* (IEEE, 2014)
4. A.A. Ahmed Abdelrahman, *Design and Implementation of Vehicle Tracking and Theft Control System* (IEEE, 2015)
5. S.S. Dukare, D.A. Patil, K.P. Rane, Vehicle tracking, monitoring and alerting system: a review. *Int. J. Comput. Appl.* **119**(10) (2015)
6. H. Jose, L. Harikrishnan, A review on vehicular monitoring and tracking. *IOSR J. Electron. Commun. Eng. (IOSR-JESE)* **9**(3), 20–23 (2014)
7. O. Aloquili, A. Elbanna, A. Al-Azizi, Automatic vehicle location tracking system based on GIS environment. *IET Software* **3**(4), 255–263 (2009)
8. E. Ilie Zudor, Z. Kemény, P. Egri, L. Monostori, The RFID technology and its current applications, in *Proceedings of the Modern Information Technology in the Innovation Processes of the Industrial Enterprises*, vol. 963 (2006), pp. 29–36
9. S.L. Ting, L.X. Wang, W.H. Ip, A study of RFID adoption for vehicle tracking in a container terminal. *J. Ind. Eng. Manage.* 22–52 (2012)
10. M.A. Hannan, A. Mustapha, A. Hussain, RFID and communication technologies for an intelligent bus monitoring and management system. *Turk. J. Electr. Eng. Comput. Sci.* 106–120 (2012)
11. L. Xiao, Z. Wang, Internet of Things: a new application for intelligent traffic monitoring system. *J. Networks* **6**(6) (2011)
12. R. Pendor, P. Tasgaonkar, *An IoT Framework for Intelligent Vehicle Monitoring System* (IEEE, 2016)
13. P.H. Oat, M. Drieberg, N.C. Cuong, *Development of Vehicle Tracking System Using GPS and GSM Modem* (IEEE, 2013)

14. M.R.A. Fuad, M. Drieberg, *Remote Vehicle Tracking System Using GSM Modem and Google Map* (IEEE, 2013)
15. S.J. Lee, G. Tewolde, J. Kwon, *Design and Implementation of Vehicle Tracking System Using GPS/GSM/GPRS Technology and Smartphone Application* (IEEE, 2014)
16. P.A. Shinde, Y.B. Mane, *Advanced Vehicle Monitoring and Tracking System Based on Raspberry Pi* (IEEE, 2015)
17. P.A. Shinde, Y.B. Mane, P.H. Tarange, *Real Time Vehicle Monitoring and Tracking System Based on Embedded Linux Board and Android Application* (IEEE, 2015)
18. F.B.B. Alzahri, M. Sabudin, *Vehicle Tracking Device* (IEEE, 2016)
19. O.A. Mohamad, R.T. Hameed, N. Tapuş, *Design and Implementation of Real Time Tracking System Based on Arduino Intel Galileo* (IEEE, 2016)
20. R.B. Pendor, P.P. Tasgaonkar, *An IoT Framework for Intelligent Vehicle Monitoring System* (IEEE, 2016)
21. M. Desai, A. Phadke, *Internet of Things based Vehicle Monitoring System* (IEEE, 2017)
22. B. Godavarthi, P. Nalajala, V. Ganapuram, Design and implementation of vehicle navigation system in urban environments using Internet of Things (IoT). IOP Conf. Ser. Mater. Sci. Eng. (2017)
23. E.B. Panganiban, J.C. Dela Cruz, *RFID Based Vehicle Monitoring System* (IEEE, 2017)
24. A.H. Alquhali, M. Roslee, M.Y. Alias, K.S. Mohamed, *IOT Based Real Time Vehicle Tracking System* (IEEE, 2019)

Machine Learning-Driven Algorithms for Network Anomaly Detection



Md. Sirajul Islam, Mohammad Abdur Rouf, A. H. M. Shahariar Parvez, and Prajoy Podder

Abstract Network security has grown more critical over the past few years as the Internet infrastructure evolves and new technologies become available. An intrusion detection system (IDS) is any kind of security software that can detect and alert administrators of unwanted network access attempts. As a consequence, implementing an intrusion detection system (IDS) to protect network systems is essential. It is a kind of defensive technology that is used to keep networks safe from hackers. IDS is beneficial for both identifying successful attacks and monitoring for suspicious behaviour. Intrusion detection systems provide a solid basis for network protection (IDS). The increased usage of social media networks and cloud computing results in the production of a huge amount of data. With the growth of data production, intrusion attacks may take on a variety of forms. This article focuses mostly on machine learning (ML) techniques for intrusion detection and security. Six different classification techniques are used: KNN, Logistic regression (LR), Naive Bayes (NB) classifier, XGB, DT, and Light Gradient Boosting Machine (LGBM).

Keywords Intrusion detection system (IDS) · Machine learning (ML) · Logistic regression · Naive Bayes · LGBM · Feature selection

Present Address:

Md. Sirajul Islam · M. A. Rouf · A. H. M. Shahariar Parvez
Department of Computer Science and Engineering, Dhaka University of Engineering & Technology, Gazipur 1707, Bangladesh
e-mail: mr.s.islam72@duet.ac.bd

M. A. Rouf
e-mail: marouf.cse@duet.ac.bd

A. H. M. Shahariar Parvez
e-mail: shahariar.parvez@bu.edu.bd

P. Podder (✉)
Institute of Information and Communication Technology (ICT), Bangladesh University of Engineering and Technology (BUET), Dhaka 1205, Bangladesh
e-mail: 0416312017@ict.buet.ac.bd

1 Introduction

The present age is the age of the internet, so the use of the internet is increasing. Now we can't go without the internet. We start using it after waking up and end up going to sleep. So the use of the internet is increasing. More and more bytes are having to increase data production due to increasing usage [1, 2]. Along with the advantages, there are many disadvantages. Because of the increase in data, hacking tools are also increasing. Different information is hacked. So the need to save this data through discussion is immense. In order to save data, the speed and amount of data have to be identified in different ways. Big data techniques are used to infiltrate and manage the data in the traditional way.

All this information is defined under 7v. Big data strategies are added together. It has some size, speed, value, authenticity, variability, invent. There are such data sizes. Data will be generated with speed. Different types of data and their values have value and variability. Invent allows data to be transferred very securely and easily. This makes the data completion rate difficult for traditional data handling systems [3, 4]. Managing this kind of data requires solid and robust advanced growing algorithms in the nature of data.

Now there are various demands for current cybersecurity. Because of increasing it a lot. As a result, cybersecurity is uncertain. With the use of computer networks, various applications are being used by some groups of people to face various problems as a result of cyberattacks. Larger networks have to bear more financial losses. Different companies and educational institutions are affected by these cyberattacks at different times, and it is financially detrimental. The big rich countries are suffering more as a result. Many countries have been demanding cyber protection from cyberattacks. To protect cyber, you have to take all kinds of measures starting from different computer networks. It identifies to be more capable. Cyberattacks in line with the needs of the world cybersecurity demands are growing. In large-scale networks, cyber-attacks such as denial-of-service (DoS) attacks [5, 6], computer malware [7], and illegal access caused irreversible harm and financial losses.

A cybersecurity system is a system that works on network security and computer security. Firewalls and encryption, the internet are designed to manage cyber attacks. We have to face many problems as a result of cyberattacks. So there is a lot of need for cybersecurity. So, we need to ensure cybersecurity for computer networks various privacy. Firewall monitors and manages the exact direction of most computer activities IDS is created for cyberattacks, and the necessary protection plays an important role in network security [8–10]. An IDS monitors and evaluates everyday network or computer system activities to detect security concerns or threats such as DoS attacks. Illegal system behavior, such as unauthorized access, modification, or destruction, may also be detected, determined, and identified using an intrusion detection system [11, 12].

The most important thing to avoid such cyberattacks is awareness and knowing how the attacks happen, how they will damage the system and how to protect against them. One type of cyberattack is phishing, in which case the perpetrator will say that

data has been stolen from the account. In this way, everyone has to be the victim of a cyber-attack [13, 14].

Awareness in the cyber world is essential to keep computer networks safe. Moving without computers and smart device is out of the question. Not just work, important information, including income and expenditure of different organizations, are stored in the computer in the form of digital files. For this, it is more important to ensure the security of the office computer. It is most important to keep the network that we usually use securely. We should be careful about data protection. The computer can learn anything without having to write a program on the subject beforehand; this is machine learning. A computer can do anything very easily because of its ability to learn on its own. In other words, if the number of computer games increases as well as its winning rate, then it must be understood that the computer is learning. This means that he is learning to play, and this ability to learn on his own is called machine learning.

Machine learning and artificial intelligence can be used to identify a subject. For example, predicting whether a person will repay a loan before giving a bank loan can be found with machine learning. Specific information is used to verify the feasibility. In this paper, network anomaly classification can be performed on a dataset collected from Kaggle using some ML models. The nobility of the proposed work is that ML models have been applied to the selected features of the dataset. The Top 15 Features have been chosen using RFE technique. Then ML models are applied after necessary parameter tuning in order to get good accuracy, precision and recall. Since LGBM provides the highest accuracy among the six ML models, tuning for LGBM hyperparameters are also performed.

This paper can be organized as follows: Sects. 2 and 3 describes the literature review and IDS applications. Dataset description and Data analysis are briefly described in Sects. 4 and 5, respectively. Experimental results are clearly elaborated in Sect. 6, where Sect. 7 concludes the paper.

2 Literature Reviews

Over the last decade, much work has been made to improve IDS detection and to prevent various malicious attempts from gaining access to computer data. This section considers various machine learning methods and algorithms applied in the intrusion detection classification process, including feature selection schemes, data preprocessing, metric evaluation, the number of selected features, and classification algorithms.

The authors of [15] described an intrusion detection technique in a wireless network based on machine learning techniques such as feature selection strategies and classification algorithms in a study published in 2018. A preprocessing step is done before the training phase, which involves data set conversion to integers, managing large data sets, and normalizing data in narrower ranges. Classifiers including Random Tree, AdaBoost, J48, Multi-Layer Perceptron, Random Forest,

ZeroR and Logit Boost were all used. The paper focuses on the efficiency of the classification algorithm's reduction function, which leads to improved detection and speed accuracy. The training model was built using four different feature sets: 5, 7, 10 and 32. The test results indicate that the random forest classifier's 32 chosen features are linked with the best outcomes. For the categorization techniques, precision is 99.64%, recall is 96.6%, and precision is 99.5%. The suggested method was tested on the wireless dataset of AWID. A comparison between the suggested method and current classification approaches is performed to validate the findings [15].

In [16], the authors conducted a study in 2018 on the performance of four algorithms: Random Forest, SVM, Decision Tree, and Naive Bayes. The technique categorizes network traffic intrusion detection using Apache Spark tools. The public data set of UNSW-NB15 for intrusion detection in the network is used to build a model with 42 features. The results in the experiment indicate that classifier of random forest is the most accurate classification method, with a sensitivity of 97.49% and a specificity of 97.75%.

Bhosale et al. [17] introduced the filter-based hybrid feature selection algorithm (HFSA) in 2018 for an acceptable selection feature process. HFSA optimized a selection of the most significant and top class features in order to generate classifications for each multi-class. This approach is based on real-time packet capture using the Jpcap package. The Naïve Bayes classifier technique is applied to categorize common malicious attacks. The preprocessing procedure is divided into two stages. To begin, data transformation changes symbolic information to a numerical value. Second, data normalization entails scaling features between the biggest and smallest ranges (0,1) and standardizing all records. The characteristics are then chosen to identify six common types of attacks using Naive Bayes: R2L, Normal, DoS, U2R, Brute Forces and Sample. HFSA is used to enhance the categorizing system for the purpose of upgrading. The model's total accuracy ratio was 92, 95, and 90%.

The work of [18] suggested classification using the Support Vector Machine (SVM) and Naive Bayes for IDS. Only 24 functions from 42 in the NSL-KDD data set were chosen using the selected feature type correlation subset. The experimental results indicate that the SVM is the superior classifier, with a total accuracy of 93.95%, when compared to the Naïve Bayes.

Kazi et al. [19] published a paper in 2019 describing a novel supervised method for evaluating and classifying network data for malignant attacks. The SVM and the ANN were utilized in this study to categorize data. Both feature selection methods were based on correlation using a filter-based Chi-Square approach and a wrapper-based feature selection method. The NSL-KDD dataset, which contains 25,191 items, is applied to train the model. The process employs a correlation-based wrapping method with 17 additional significant features selected from 41. On the other side, a chi-square-based filter is used to pick 35 features that are more informative and significant for the training model stage. The testing results indicate that ANNs outperforms all other approaches by 94.02% when wrapped around 17 features.

A research paper [20] published a novel classification method in 2020 that blends CART with RF feature selection. The Hybrid Intrusion Detection System for Anomalies is the name given to this technology (HAIDS). To improve model efficiency, the

hybrid approach is used rather than a single algorithm. Additionally, the method of removing superfluous features is utilized to compensate for the high dimension. The proposed model was applied to the UNSW-NB15 data set, and the top 13 features were chosen. The hybrid method achieved the highest performance and accuracy, with 11,86% false alert and 87,74% accuracy.

Iman et al. [21] presented a random forest classification technique in 2020, using the Boruta algorithm to construct IDS on the NSL-KDD dataset. Additionally, the entropy and Gini indexes are calculated as a z-score for the number of tree depth values. 34 out of 41 attributes are optimal. The accuracy, sensitivity, and specificity of the proposed model were all determined to be 0.99.

Latah et al. [22] suggested a hybrid network-defined hybrid intrusion detection system in 2020. The hybrid system is composed of the HELM, KNN and Extreme Learning Machine (ELM) method and the algorithm. The system results reported here demonstrated high accuracy of 84.29% when used with the KDD Cup 99 dataset. Additionally, the method has a detection rate of 77.18% for fresh attacks.

In 2020, the work of [23] proposed a novel technique based on the SVM and Naïve Bayes integrated feature for intrusion detection. The Naïve Bayes algorithm is used to modify characteristics in order to transform data. As a classifier, the SVM method is employed. We utilized a combination of data sets to identify different attack types, including CICIDS2017, UNSW-NB15, Kyoto 2006+ and NSL-KDD, by using multiple functions for each data source. The proposed method, comparing the results of the embedding system to those of a single SVM algorithm, discovered that the embedded Naive Bays with SVM provided the highest detection accuracy. This work got 99.36% accuracy for NSL-KDD data.

The authors of [24] developed hybrid algorithms for IDS classification and an enhanced profile to detect anomalous user behaviour in 2020. SVM and Naive Bayes classification algorithms are used in the hybrid techniques. Furthermore, it provided data preprocessing. The positive effect on model accuracy is data normalization, which scales features between 0 and 1 and selects suitable characteristics during real-time data gathering. This hybrid approach produces classifiers with a total accuracy of 93.1% and a precision of 95.8%. Also included are the Classifier Enhancement (CE) accuracy values of 95.3% and 95.8%.

Kumari et al. [25] suggested a mixed classification scheme for IDS in 2020. The hybrid is a combination of the SVM and Decision Tree J48. The SVM is capable of resolving the problem of high dimensions. Additionally, the Particle Swarm Optimization (PSO) technique selected nine of the 42 critical criteria for feature extraction. For both the training and testing stages of the document, the KDD99 dataset is used. Numerous ratios are included in the data collection. The results indicate that it is optimal to use 70% of the data set for testing and 30% for training since this increases accuracy and decreases the incidence of false alarms. In general, the hybrid model achieves a total accuracy ratio of 99.1%, a detection rate of 99.6%, and a FAR of 0.90%.

The authors of [26] created a system for intrusion detection using IoT technology in 2020 to upgrade the electrical grid to a smart grid (SG) and identify regular hazardous attacks. They developed a hybrid of three Decision Trees (HDT)

for detecting different types of attacks in this research. Additionally, the suggested hybrid approach's performance has been compared to that of K-Nearest Neighbors (KNN), SVM, and DT techniques. The results of the testing indicated that the (HDT) technique proposed in conjunction with the evaluation measurement was more efficient in terms of accuracy 83.1485%, precision 97.21%, with 72.45%, and F-Score 83.04% when NSLKDD was used.

In 2020, the works of [27] created a DDoS detection algorithm to improve network security using machine learning techniques. K-Nearest Neighbor and Naive Bayes methods were utilized for classification, whereas correlation was used for feature extraction. For NSL-KDD and KDD Cup 99, the proposed method was compared to conventional learning models. The testing findings indicated that utilizing its KNN algorithm with eight characteristics, Naive Bayes got the best results. These various metrics are used to evaluate algorithm performance. They include accuracy of 98.51%, accuracy of 98.9%, recall of 97.8%, F-measure of 1.01%, sensitivity of 97.8%, specificity of 99.12%, efficiency of 98.48%, the error rate of 1.5%, and ROC of 0.98%.

3 IDS Applications

IDS is critical for protecting humans from cyber-attacks. Every information and transaction processing takes place via the internet, which is extremely susceptible to a variety of hostile actions. As a result, a greater emphasis on information security is required. The following applications are possible:

3.1 *IoT*s in IDS

The IoT is a term that refers to an object or gadget that has a unique identity and is capable of detecting, collecting, and transmitting data over the internet without the intervention of a person or human interaction with a computer. We power low-power Internet of Things devices and develop lightweight protocols. Additionally, it is lightweight. The authors of [28] illustrated the purpose of IoT in smart grid. It is very vulnerable, and even attackers are capable of altering the data on the sensors. Physical attacks, environmental attacks, Sybil attacks, side channel attacks, black hole attacks, and cryptanalysis attacks are the most common types of attacks against IoT devices. The work of [29] proposed using supervised learning to detect light penetration. They developed SVM classification to assist in identifying attacks (target DDoS). The authors of [30] addressed anomaly and assault detection. They have been executing their work using machine learning algorithms including SVM, LR, RF, ANN and Decision Tree.

3.2 *Smart City with IDS*

The authors of [31] presented detection of intrusion in smart cities. The author made use of data collected from the intelligent water distribution system. The objective is to detect DDoS attacks on intelligent city applications. The paper proposed a two-part approach: A Boltzmann Machine Restricted model and a classification model. This approach was divided into two sections. This RBM model was used to learn high-level characteristics in the absence of human supervision. The classification was intended to differentiate between common and varied distributed denial-of-service attacks. They classified using four different techniques: SVM, FFNN, RF and Automated FFNN. For the high-level features, RBM processed the K-Means methods and created up to five layers, each of which contained five sub-versions with varying K values from the clustering algorithms. Four classification types were employed for each of the five data sets generated via clustering, and a total of twenty trials were conducted.

3.3 *Big Data in IDS Environment*

The term “big data” refers to very large quantities of disparately ordered, semi-structured and unstructured data. For such a massive amount of data, a traditional intrusion management system is incapable of resolving the issues. IDS in a big data environment is only possible via the application of machine learning algorithms.

3.4 *Fog Computing in IDS*

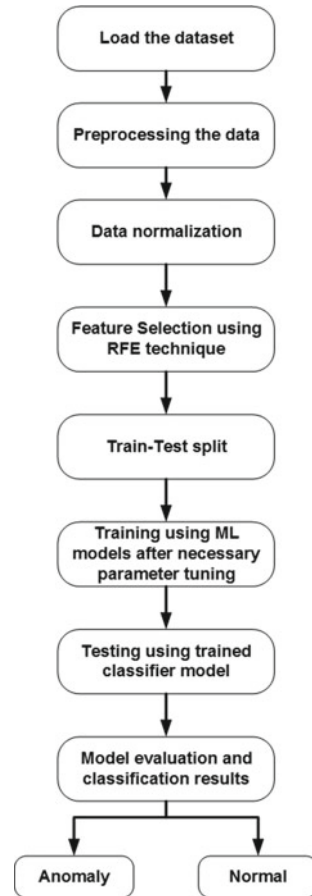
Fog computing is a novel computer paradigm that prioritizes analytical services and enhances performance via closer interaction between resources. In fog computing, there are three kinds of layers: user layer, cloud service layer and fog service layer. The fog service layer is composed of a globally distributed fog node that consists of routers, a gateway, a server, and a proprietary fog computing layer. Fog nodes connect heterogeneous systems, making them vulnerable to assaults such as DDoS, PROBE, Remote-to-Local (R2L) and User-to-Root (U2R) [32, 33].

4 **Dataset Description**

The dataset contains simulations of a variety of network intrusions conducted in a military simulation environment. The dataset has been collected from the Kaggle Repository [34]. Simulating a normal US Air Force LAN enabled the acquisition of raw IP/TCP dump data. The LAN resembled a real ecology and was always under

attack from a variety of threats. A connection is a collection of TCP packets in transit between two places, starting and ending within a specified time period, and delivering data to and from the source IP address to the destination IP address, all while adhering to a well-defined protocol. From 41 quantitative and qualitative attributes of each TCP/IP connection, three qualitative and 38 quantitative characteristics are extracted from normal and attack data. The class variable has two categories. (i) Anomaly: 11,743 Samples (ii) Normal: 13,449 Samples. The working procedures can be summarized using a workflow diagram shown in Fig. 1.

Fig. 1 Workflow diagram



5 Data Preparation and Analysis

1. To identify any inconsistencies, intriguing patterns, or correlations in the data, we went through the data to look for discrepancies, patterns, or correlations. Exploratory data analysis is also commonly known as this phase.
2. After using the usual data preparation methods, we finished the analysis. During this period, much of our time was dedicated to this activity.
3. Other techniques used include: (a) data cleaning to handle missing values by calculating an estimated mean value, and (b) data verification and cleaning.
4. A popular preprocessing method used in machine learning is normalization to a scale of 0–1.
5. For most of the models, strings and objects are completely unprocessable. Once the data has been obtained, it must be translated into numerical values. Data encoding is known as data transformation.
6. Feature Selection—Discarding or choosing superfluous features or just deciding which “valuable” features are used. To use recursive feature elimination for feature selection, we employed recursive feature elimination.

5.1 Feature Selection

Feature elimination, sometimes called recursive feature elimination, is a commonly used technique for selecting features. RFE is selected due to its easiness of setup, usage, and effectiveness in filtering the training dataset to retain just the most relevant characteristics (columns). Determining the number of features choose and the technique to select features are critical when applying RFE. Although the technique’s success is not entirely dependent on these hyperparameters being set correctly, they may be explored. Recursive Feature Elimination (RFE) has been utilized to perform feature elimination in Python. After completing this task, you will understand:

1. Feature selection is accomplished by using a feature elimination method.
2. Feature selection techniques may be used to identify classes and model predictive variables.
3. To determine the number of chosen features and method that was wrapped, RFE has a process that allows the exploration of these concepts.

Algorithms for feature selection, such as RFE, are wrapper-type algorithms. RFE is used in this instance to include another machine learning algorithm into the method’s core, encapsulate it, and use it to assist in feature selection. Contrary to feature selection filtering techniques, feature-based selections are made in which each feature is rated and the features with the highest (or lowest) score are chosen. While RFE may be considered a wrapper technique for feature selection, the underlying filter-based feature selection mechanism is unique to RFE. RFE begins with all of the characteristics in the training dataset and searches for a subset that matches. The procedure is repeated until the necessary number of characteristics remains.

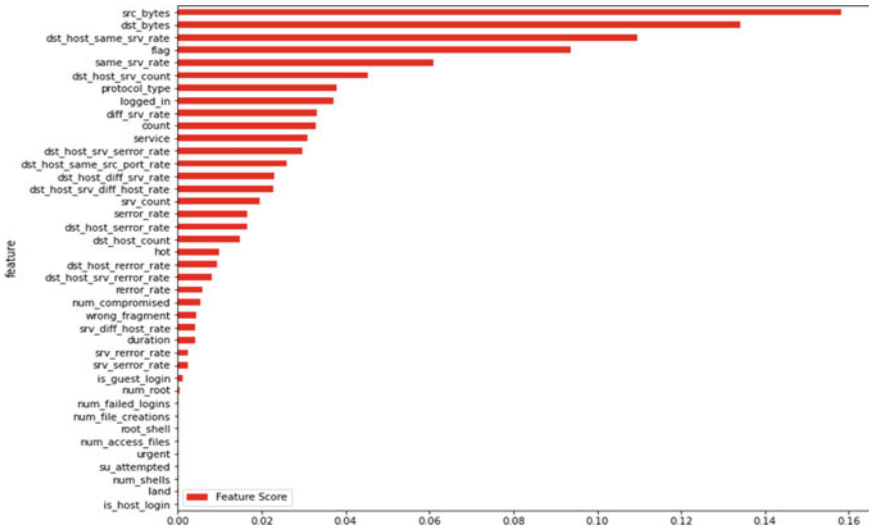


Fig. 2 Features of IDS dataset

This is accomplished by first fitting the supplied machine learning algorithm that serves as the model’s core, ranking features according to their importance, and then removing the least significant features. This procedure is continued until a model is discovered that is well-suited to the training data. Once the required number of attributes is achieved, the process is repeated. Figure 2 depicts the top important features to less important features. Moreover, Table 1 is listed the selected features for experiments.

6 Experimental Analysis

The machine learning findings on the IDS dataset are presented in this section. The top 15 features are shown in Table 1. There are 25,192 training samples which can be categorized into two classes: (i) Normal: 13,449 samples, (ii) Anomaly: 11,743 samples. Classifiers are deployed to determine the optimal number of features when the data is varied based on the number of features. Table 2 presents the values of confusion matrices for various well-known algorithms and the performances of the classifiers are calculated in Tables 3 and 4. TP, FP, TN and FN mean true positive, false positive, true negative and false negative respectively. Accuracy (Acc), Precision (P), recall (R), Specificity (S), False detection rate (FDR), MCC can be described by using the following mathematical expressions [35, 36]:

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

Table 1 List of selected features

Feature	Rank
'src_bytes'	1
'dst_bytes'	2
'dst_host_same_srv_rate'	3
'flag'	4
'same_srv_rate'	5
'dst_host_srv_count'	6
'protocol_type'	7
'logged_in'	8
'diff_srv_rate'	9
'count'	10
'service'	11
'dst_host_srv_serror_rate'	12
'dst_host_same_src_port_rate'	13
'dst_host_diff_srv_rate'	14
'dst_host_srv_diff_host_rate'	15

Table 2 TP, TN, FP and FN calculation of ML models

ML model	TP	TN	FP	FN
KNN	2346	2658	16	19
LR	2242	2572	102	123
NB	2023	2582	92	342
XGB	2350	2671	3	15
LGBM	2361	2671	3	4
DT	2358	2663	11	7

$$P = \frac{TP}{TP + FP} \quad (2)$$

$$R = \frac{TP}{TP + FN} \quad (3)$$

$$S = \frac{TN}{TN + FP} \quad (4)$$

$$FDR = \frac{FP}{TP + FP} \quad (5)$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (6)$$

Table 3 Performance analysis of ML models

Model	Class	Precision	Recall	F1-Score
KNN	Anomaly	0.99	0.99	0.99
	Normal	0.99	0.99	0.99
LR	Anomaly	0.96	0.95	0.95
	Normal	0.95	0.96	0.96
NB	Anomaly	0.96	0.86	0.90
	Normal	0.88	0.97	0.92
XGB	Anomaly	1.00	0.99	1.00
	Normal	0.99	1.00	1.00
LGBM	Anomaly	1.00	1.00	1.00
	Normal	1.00	1.00	1.00
DT	Anomaly	1.00	1.00	1.00
	Normal	1.00	1.00	1.00

Table 4 Accuracy, specificity and FDR comparison of ML models

ML model	Accuracy	Specificity	FDR	MCC
KNN	99.305	0.9940	0.0068	0.9861
LR	95.535	0.9619	0.0435	0.9103
NB	91.387	0.9656	0.0435	0.8302
XGB	99.643	0.9989	0.0013	0.9928
LGBM	99.861	0.9989	0.0013	0.9972
DT	99.643	0.9959	0.0046	0.9928

In Table 3, LGBM and DT provide the highest recall, precision and the value of F1 score. But, LGBM provides the highest 99.861% accuracy for learning rate of 0.3, whereas 99.643% accuracy is provided by DT and XGB in Table 4. Moreover, NB depicts the lowest recall, precision and the value of F1 score. Besides, it gives an accuracy of 91.387%. This is also the lowest value among the other classifiers. Performance metrics of existing technologies adopted by various researchers are compared with proposed method results in Table 5. Since a custom dataset has been used in this paper, direct comparison is not feasible.

7 Conclusion

The IDS is designed to provide the bare minimum security protections necessary to safeguard any networks connected to the internet. However, the network administrator is ultimately responsible for network security. IDS enables network managers

Table 5 Comparison with state of the art methods

References	Dataset	ML model	Accuracy	Precision	Recall
[26]	NSL-KDD (attack: 71,463, normal: 77,054)	HDT	83.149	97.219	72.469
		DT	80.908	96.775	68.752
		KNN	79.121	70.736	89.546
		SVM	78.522	71.429	85.227
[27]	NSL-KDD	KNN	98.51	98.9	97.8
	KDD Cup 99	NB	93.95	97.74	95.54
[37]	KDD Cup dataset (10% samples)	SVM	97.29	–	–
		NB	71.001	–	–
[15]	AWID-ATK-R (selecting top 10 features)	J48	96.41	92.9	96.2
		ZeroR	96.31	92.8	96.3
		Random Forest	96.43	93.1	96.4
		AdaBoost	95.88	96.2	95.9
Proposed ensemble method	Custom dataset	LGBM	99.861	100	100
		XGB	99.643	99	99

to identify and track intrusion on the internet whose only purpose is to infiltrate a network and facilitate attacks. This work illustrates the RFE algorithm for selecting features. After that, we have used these features for detecting intrusion more effectively. According to our experiment, LGBM provides better results after necessary parameter tuning than other existing ML algorithms included in this work.

References

1. A. Amouri, V.T. Alaparthi, S.D. Morgera, A machine learning based intrusion detection system for mobile Internet of Things. *Sensors* **20**(2), 461 (2020)
2. P. Podder, M. Mondal, S. Bharati, P.K. Paul, Review on the security threats of internet of things (2021). *arXiv preprint arXiv:2101.05614*
3. A.H.M. Shahariar Parvez, M. Robiul Alam Robel, M.A. Rouf, P. Podder, S. Bharati, Effect of fault tolerance in the field of cloud computing, in *Inventive Computation Technologies. ICICIT 2019. Lecture Notes in Networks and Systems*, ed. by S. Smys, R. Bestak, Á. Rocha, vol 98 (Springer, Cham, 2020). https://doi.org/10.1007/978-3-030-33846-6_34
4. S. Bharati, P. Podder, M.R.H. Mondal, P.K. Paul, Applications and challenges of cloud integrated IoMT, in *Cognitive Internet of Medical Things for Smart Healthcare. Studies in Systems, Decision and Control*, ed. by A.E. Hassanien, A. Khamparia, D. Gupta, K. Shankar, A. Slowik, vol 311 (Springer, Cham, 2021). https://doi.org/10.1007/978-3-030-55833-8_4
5. N. Sun, J. Zhang, P. Rimba, S. Gao, L.Y. Zhang, Y. Xiang, Data-driven cybersecurity incident prediction: a survey. *IEEE Commun. Surv. Tutor.* **21**, 1744–1772 (2018)
6. P. Podder, S. Bharati, M.R.H. Mondal, P.K. Paul, U. Kose, Artificial neural network for cybersecurity: a comprehensive review. *J. Inf. Assur. Secur.* **16**(1), 010–023 (2021). ISSN: 1554-1010

7. M. Wazid, A.K. Das, J.J. Rodrigues, S. Shetty, Y. Park, IoMT malware detection approaches: analysis and research challenges. *IEEE Access* **7**, 182459–182476 (2019)
8. F. Salo, M. Injadat, A.B. Nassif, A. Shami, A. Essex, Data mining techniques in intrusion detection systems: a systematic literature review. *IEEE Access* **6**, 56046–56058 (2018)
9. I. Ahmad, M. Basher, M.J. Iqbal, A. Rahim, Performance comparison of support vector machine, random forest, and extreme learning machine for intrusion detection. *IEEE access* **6**, 33789–33795 (2018)
10. M. Wang, K. Zheng, Y. Yang, X. Wang, An explainable machine learning framework for intrusion detection systems. *IEEE Access* **8**, 73127–73141 (2020)
11. S. Gulghane, V. Shingate, S. Bondgulwar, G. Awari, P. Sagar, A survey on intrusion detection system using machine learning algorithms, in *Innovative Data Communication Technologies and Application. ICIDCA 2019. Lecture Notes on Data Engineering and Communications Technologies*, ed. by J. Raj, A. Bashar, S. Ramson, vol. 46 (Springer, Cham, 2020). https://doi.org/10.1007/978-3-030-38040-3_76
12. T.T. Bhavani, M.K. Rao, A.M. Reddy, Network intrusion detection system using random forest and decision tree machine learning techniques, in *First International Conference on Sustainable Technologies for Computational Intelligence. Advances in Intelligent Systems and Computing*, ed. by A. Luhach, J. Kosa, R. Poonia, X.Z. Gao, D. Singh, vol. 1045 (Springer, Singapore, 2020). https://doi.org/10.1007/978-981-15-0029-9_50
13. K.A. Parmar, D. Rathod, M.B. Nayak, Intrusion detection system using semi-supervised machine learning, in *Data Science and Intelligent Applications. Lecture Notes on Data Engineering and Communications Technologies*, ed. by K. Kotecha, V. Piuri, H. Shah, R. Patel, vol. 52 (Springer, Singapore, 2021). https://doi.org/10.1007/978-981-15-4474-3_27
14. M. Chauhan, A. Joon, A. Agrawal, S. Kaushal, R. Kumari, Intrusion detection system for securing computer networks using machine learning: a literature review, in *Congress on Intelligent Systems. CIS 2020. Advances in Intelligent Systems and Computing*, ed. by H. Sharma, M. Saraswat, A. Yadav, J.H. Kim, J.C. Bansal, vol. 1334 (Springer, Singapore, 2021). https://doi.org/10.1007/978-981-33-6981-8_15
15. R. Abdulhammed, M. Faezipour, A. Abuzneid, A. Alessa, Effective features selection and machine learning classifiers for improved wireless intrusion detection, in *2018 International Symposium on Networks, Computers and Communications (ISNCC)*. IEEE, June 2018, pp. 1–6
16. M. Belouch, S. El Hadaj, M. Idhammad, Performance evaluation of intrusion detection based on machine learning using Apache Spark. *Procedia Comput. Sci.* **127**, 1–6 (2018)
17. K.S. Bhosale, M. Nenova, G. Iliev, Data mining based advanced algorithm for intrusion detections in communication networks, in *2018 International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS)*. IEEE, Dec 2018, pp. 297–300
18. K.K. Gulla, P. Viswanath, S.B. Veluru, R.R. Kumar, Machine learning based intrusion detection techniques, in *Handbook of Computer Networks and Cyber Security* (Springer, Cham, 2020), pp. 873–888
19. K.A. Taher, B.M.Y. Jisan, M.M. Rahman, Network intrusion detection using supervised machine learning technique with feature selection, in *2019 International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST)*. IEEE, Jan 2019, pp. 643–646
20. Z. Chkurbene, S. Eltanbouly, M. Bashendy, N. AlNaimi, A. Erbad, Hybrid machine learning for network anomaly intrusion detection, in *2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIOT)*. IEEE, Feb 2020, pp. 163–170
21. A.N. Iman, T. Ahmad, Improving intrusion detection system by estimating parameters of random forest in Boruta, in *2020 International Conference on Smart Technology and Applications (ICoSTA)*. IEEE, Feb 2020, pp. 1–6
22. M. Latah, L. Toker, An efficient flow-based multi-level hybrid intrusion detection system for software-defined networks. *CCF Trans. Network.* **3**(3), 261–271 (2020)
23. J. Gu, S. Lu, An effective intrusion detection approach using SVM with naïve Bayes feature embedding. *Comput. Secur.* **103**, 102158 (2021)
24. P. Pokharel, R. Pokharel, S. Sigdel, Intrusion detection system based on hybrid classifier and user profile enhancement techniques, in *2020 International Workshop on Big Data and Information Security (IWBSIS)*. IEEE, Oct 2020, pp. 137–144

25. A. Kumari, A.K. Mehta, A hybrid intrusion detection system based on decision tree and support vector machine, in *2020 IEEE 5th International Conference on Computing Communication and Automation (ICCCA)*. IEEE, Oct 2020, pp. 396–400
26. S.M. Taghavinejad, M. Taghavinejad, L. Shahmiri, M. Zavvar, M.H. Zavvar, Intrusion detection in IoT-based smart grid using hybrid decision tree, in *2020 6th International Conference on Web Research (ICWR)*. IEEE, Apr 2020, pp. 152–156
27. A.V. Kachavimath, S.V. Nazare, S.S. Akki, Distributed denial of service attack detection using Naïve Bayes and K-nearest neighbor for network forensics, in *2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)*. IEEE, Mar 2020, pp. 711–717
28. A. Ghasempour, Internet of Things in smart grid: architecture, applications, services, key technologies, and challenges. *Inventions* **4**(1), 22 (2019)
29. Jan SU, Ahmed S, Shakhov V, Insookoo, Towards a lightweight intrusion detection system for the Internet of Things. *IEEE Access* **7**(1), 42450–42471 (2019)
30. M. Hasan, M.M. Islam, M.I.I. Zarif, M.M.A. Hashem, Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches. *Internet of Things* **7**(1), 100059 (2019)
31. A. Elsaedy, K.S. Munasinghe, D. Sharma, A. Jamalipour, Intrusion detection in smart cities using Restricted Boltzmann Machines. *J. Netw. Comput. Appl.* **135**(1), 76–83 (2019)
32. J.S. Raj, Improved response time and energy management for mobile cloud computing using computational offloading. *J. ISMAC* **2**(01), 38–49 (2020)
33. V. Suma, W. Haoxiang, Optimal key handover management for enhancing security in mobile network. *J. Trends Comput. Sci. Smart Technol. (TCSST)* **2**(04), 181–187 (2020)
34. <https://www.kaggle.com/sampadab17/network-intrusion-detection>. Last accessed on 1 July 2021
35. M. Alam, P. Podder, S. Bharati, M.R.H. Mondal, Effective machine learning approaches for credit card fraud detection, in *Innovations in Bio-Inspired Computing and Applications. IBICA 2020. Advances in Intelligent Systems and Computing*, ed. by A. Abraham, H. Sasaki, R. Rios, N. Gandhi, U. Singh, K. Ma, vol. 1372 (Springer, Cham, 2021). https://doi.org/10.1007/978-3-030-73603-3_14
36. S. Bharati, P. Podder, M.R.H. Mondal, Diagnosis of polycystic ovary syndrome using machine learning algorithms, in *2020 IEEE Region 10 Symposium (TENSYP)*. IEEE, June 2020, pp. 1486–1489
37. A. Halimaa, K. Sundarakantham, Machine learning based intrusion detection system, in *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*. IEEE, Apr 2019, pp. 916–920

Performance Analysis of a FSO Link Considering Different Atmospheric Turbulence



Md. Rawshan Habib, Ahmed Yousuf Suhan, Abhishek Vadher, K. M. Monzur Rahaman, A. M. Rubayet Hossain, Md. Rashedul Arefin, Md Shahnewaz Tanvir, and Shuva Dasgupta Avi

Abstract Free space optical (FSO) technology is a quick-to-deploy and economical way of getting access to the fiber optic network. FSO technology not only offers fiber-quality connections, but it also offers the sector's cheapest transmission capacity. FSO systems complement legacy network commitments and function in harmony with any protocol, saving significant up-front investments as a completely protocol-independent broadband gateway. An FSO link may be purchased and deployed at a fraction of the cost of installing fiber cable and for roughly half the cost of equivalent microwave/RF wireless systems. With exception of RF wireless technologies, FSO does not need the purchase of expensive spectrum licenses or the fulfillment of additional regulatory criteria. The purpose of this study is to examine the performance of FSO-based optical access networks. Analysis of the performance in detail, with a focus on BER is also described.

Keywords FSO system · Optical access networks · Gamma-Gamma PDF · Atmospheric turbulence · Bit error rate (BER)

1 Introduction

FSO communication uses optical signals as carrier frequencies to allow point-to-point communication data transmission over the environment. It has gotten a lot of interest in the telecommunications sector because of its low cost, ease of installation, fast formation of communication links, particularly in emergency response scenarios, high bandwidth provision, as well as wide variety of applications. Because of the

Md. Rawshan Habib · A. Vadher
Murdoch University, Murdoch, Australia

A. Y. Suhan
Curtin University, Bentley, Australia

K. M. Monzur Rahaman
United International University, Dhaka, Bangladesh

A. M. Rubayet Hossain · Md. Rashedul Arefin (✉) · M. Shahnewaz Tanvir · S. D. Avi
Ahsanullah University of Science & Technology, Dhaka, Bangladesh

frequency band in which it works, FSO transmission is not subject to licensing. Unlike RF communication technologies, which have a maximum data transmission rate of 622 Mbps, FSO communication has a maximum data transfer rate of 2.5 Gbps. The optical transmission of sound, video, and information utilizing air as the channel of transmission is referred to as FSO. FSO technology allows for comparatively easy transmission. It comprises of two units, each with transceiver that includes laser transmitter and receiver for full duplex capabilities. Every such system consists of an optical source (including a laser) which generates enough power and telescope which releases light via atmosphere to a receiver telescope. A receiving telescope is then connected to a receiver through a cable.

Irrational illumination was used in different ways for communication prior to the invention of the laser. In 1792, the wave of light was employed to light the coded messages in the development of optic telegraph, allowing appropriate interceptions at relay stations [1]. Alexander Graham Bell showed utilization of light wave as carrier in the initial FSO almost a century later [2]. The transmission distance, meanwhile, was only 200 m, far smaller compared to 1000-km transmission afforded by electrical telegraphy, that completely superseded Claude Chappe's telegraphy [3]. Many such factors contributed to the full progression and mastery of optical fiber communication systems and techniques, such as the advancement of miniature transceivers functioning with a wavelength of 1300 nm, short fiber implementation, scattering shifting fiber, and growth of an optic amplifier [1]. Because of advancements in optical fiber, FSO communication systems research and development shifted to deep space and inter-satellite usage. The growth of interest in terrestrial applications of FSO model potentially initiated near 2001, when Merrill Lynch implemented FSO for connecting link via most of the workplaces at nearly 2.5 km range, following the attackers damage of remaining communication framework at the World Trade Center [4–6]. Almost a decade later, FSO network running at 10 Gbps are established and obtainable on market [7]. The goal of this study is summarized as follows:

- To examine as well as assess the execution of FSO-based optical networks.
- To determine potential designs for forthcoming FSO optical access networks, along with a multiple access strategy.
- To find beneficial and new designs and modeling approaches, and investigate their performance in detail, with a focus on bit error rate (BER). Since, BER is the prime figure of merit in an optical communications, it must be accurately calculated when designing systems.

2 Literature Review with In-Depth Investigation

The usage of mobile devices and data-hungry apps like social media, calls, and fast internet have both increased dramatically in recent years. As a result, the restricted and authorized radio frequency (RF) spectrum has been congested, forcing network operators to utilize new ways to satisfy the rising needs of end users. The FSO method transfers data by leveraging license-free optical channels with a line-of-sight

arrangement among the connecting endpoints as a requirement. Due to its superior benefits which including reliable data transfer, fast connections, quick ordination procedure, last-mile access, copyright free, and many more, FSO research has rapidly caught the interest of many researchers and has already been implemented for high-bandwidth consuming applications. The use of orthogonal FDM to reduce channel fading is presented in [8] for an FSO network with large transmission capabilities. BER and optical SNR analysis are also carried out for multi-level QAM designs. The findings show that a 4-QAM modulation technique delivers the maximum range while requiring a small OSNR. 64-QAM performs the best spectral efficiency. An effect of various climatic variables on performance of the system is also examined. A reliable transfer of 100 Gbps 4-QAM data at 15 km using simulations is also established here. Under the impact of air turbulence, an experimental assessment of the execution of a wireless network is described in [9] which is dependent on FSO utilizing amplify-and-forward relaying. Novel BER values are also reported for different turbulence aspects for FSO links, and confirmed through simulation studies depending on the Gamma-Gamma turbulence model, which demonstrate satisfactory outcome. It is also showed that when turbulence is close to the receiving side, the BER performance drops significantly. When compared with zero turbulence, the SNR impact for a planned BER of 10^{-4} can be as high as 9 dB.

Using nonzero boresight and jitter, probability density function (PDF) for combined atmospheric turbulence and pointing faults is obtained in [10]. Over Málaga atmospheric turbulence, precise and asymptotic shuttered equations is also presented for error rate and shutdown chance of M-ASK modulation. Furthermore, correctness of the suggested analytic derivations are tested through Monte-Carlo simulations and statistical method. The precise formulas and their accompanying asymptotic developments are perfectly matched, as seen by the findings. Under modest turbulence conditions, experts want to see in [11] how well several important suggested channel models for FSO systems function. Initially, the turbulence dispersing is measured and metadata is created using an outside experimental setup. The system is entirely optical, that allows to isolate solar radiation noise during observations. Next, the collected data is utilized to test the execution of five important FSO channels that are said to function aptly in low-turbulence environments. Finally, the collected particulars are used to develop a modish empirical channel that performs well in low-turbulence situations. The functionality of MIMO-FSO frameworks with severe atmospheric turbulence is investigated in [12]. Diversity approach is utilized to reduce turbulence, which is a significant performance lowering element. For simulation, the PolSK modulation is used, and the execution is assessed using BER and good SNR. The system's execution is evaluated in a variety of weather situations, including fog and rain. On the other hand, over partly and completely linked atmospheric turbulence fading, BER performance of a FSO system is examined in [13]. A probability density function is developed and calculated the immediate BER utilizing a differential signaling method to undertake the aforesaid study. Then, BER expressions are derived and Monte Carlo simulations illustrate the precision of the obtained BER formulas.

Table 1 Comparison in between FSO and other systems

Properties	FSO	Fiber optics	MW radio	Coaxial cable
Install	Moderate	Difficult	Difficult	Moderate
Data rate	Gbps	Independent	Mbps	Mbps
Security	Good	Very good	Poor	Good
Connectivity	P2P, P2MP short and long reach	P2P, P2MP short and long reach	P2P short reach	Multidrop short reach
Maintenance	Low	Low	Low	Moderate
Spectrum license	Not required	Required	Required	Required

A study is conducted in [14] to investigate the ascendancy of turbulence on the operation of FSO system, as well as to enhance the execution of the FSO system. First, BER expression is provided by analyzing both DD and IM FSO system. On a 6.2 km connections, a test in FSO system is conducted. The results reveal that there is a strong link in between BER and scintillation index. The operation of a MIMO-based FSO link is evaluated in [15] on the basis of BER using the Gamma-Gamma (GG) distribution. To decrease the complication, the authors have modified the Meijer-G function and generated PDF equations based on power series for GG model. A PolSK modulated FSO system including wavelength and temporal diversity is considered in [16] to solve the primary constraints which are air turbulence and pointing inaccuracies. To enhance the BER performance and range coverage of high-speed FSO systems, a new relaying approach is suggested in [17]. The suggested system is theoretically studied across atmospheric turbulence channels represented by GG distribution for evaluation process. The findings, that are confirmed by Monte-Carlo simulations, show that the suggested system outperforms traditional systems. A cyber-attack recognition system is described in [18] with the help of IoT. A key handover design is proposed in [19] for mobile network security. A comparison between FSO and other communication system is shown is Table 1.

3 FSO in Communication System

FSO is a type of communication in which the modified messages are carried by laser in the surrounding atmosphere. Such FSO framework's optical transmitter might be likened with an emitter antenna in regular wireless model, as well as the optic receptor with a receiver side in such generic model. FSO's initial move input and final move output inside an overall framework are a focus of FSO study: a transferred wave has to be related to such a detected wave in a coherent fashion, thus the BER is used to assess the system's dependability. Furthermore, the allocation of light wave energy at a reception port aids in the measurement of the channel's fading effectiveness, and it would ultimately interfere with the system's BER. The signal that enters the laser sensor is a mix of the signal's channel effect, channel noise, and alignment

error induced by transmitter jitter. Figure 1 depicts a contemporary communication system architecture.

An FSO model is comprised of multiple modules in most cases which include laser emitter, transmitting lens, receiving lens, and laser sensor. Structure of a collimated FSO communication framework is depicted in Fig. 2. The FSO operating concept is identical to that of fiber optics cable models, with an exception of optic ray is transmitted via free air rather than glass fiber cores in OFC. An optical transceiver

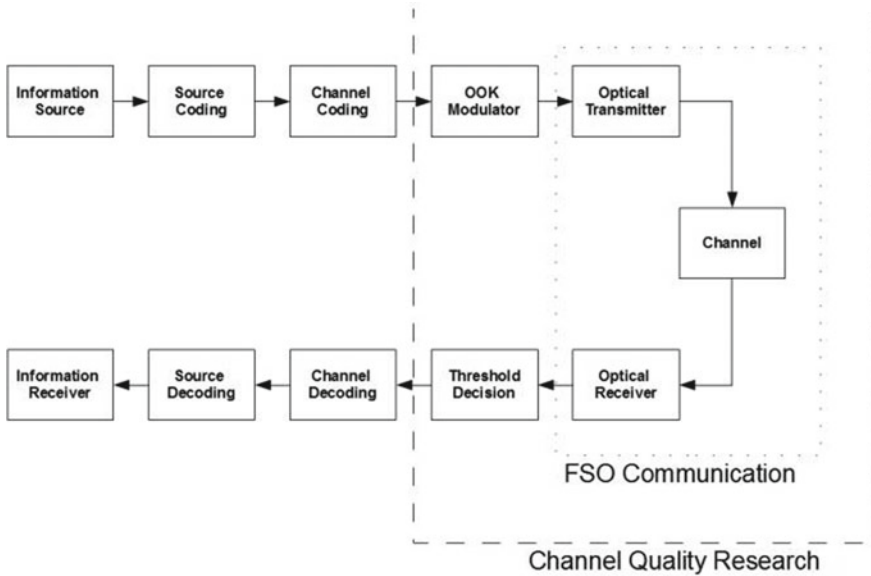


Fig. 1 A modern communication system architecture

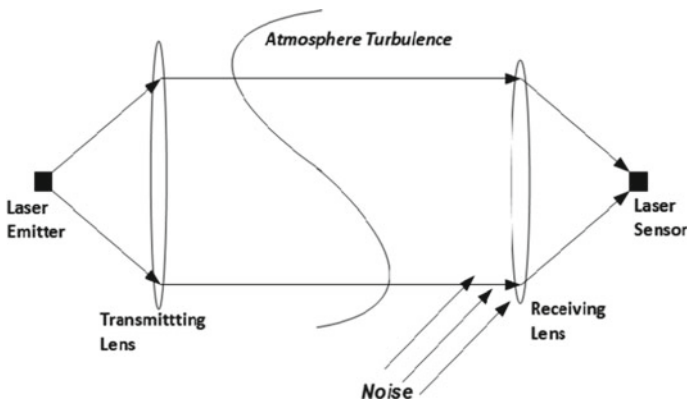


Fig. 2 Model of a collimated FSO framework

is used on each terminals of the FSO system to offer full duplex (bidirectional) capabilities. FSO system can function at higher power levels and across greater ranges. One important feature of FSO system is that it should be able to function in a broad range of temperatures with less performance deterioration for outdoor installations.

3.1 Atmospheric Effects on FSO System

The transmission medium for an FSO connection is the atmosphere. Its attenuation is dependent on a number of factors. Attenuation is mostly caused by climatic conditions. The meteorological conditions in the location where a network is being built are detailed enough that previous information about attenuation can be acquired. Atmospheric effects on FSO system is shown in Fig. 3. Haze and weighty flake, by instance, are considered the most common meteorological phenomena in temperate climates. Weighty rainfall and smog are known the most common weather conditions in tropical areas, and they show a momentous consequence on an obtainability of FSO framework in those areas [20]. Fog and haze are the usual causes of atmospheric attenuation. Dust and rain have a role as well. Atmospheric attenuation is thought to be wave based, however that is not the case. Smog is based on the wavelength. Climate and the composition of the surroundings cause atmospheric disturbance. Wind and convection are to blame, since they combined air parcels of varying temperatures. The density of air fluctuates, resulting in a switch in the refractive index of the air. Turbulence can cause a transmission optical beam to degrade. When the refractive index changes, the beam refracts at different angles and the optical beam spreads [21].

Whenever the optical beam and the scatterer meet, scattering occurs. It's a wavelength-dependent phenomena in which the energy of an optical beam remains constant. However, only directional transfer of optical energy occurs, resulting in a

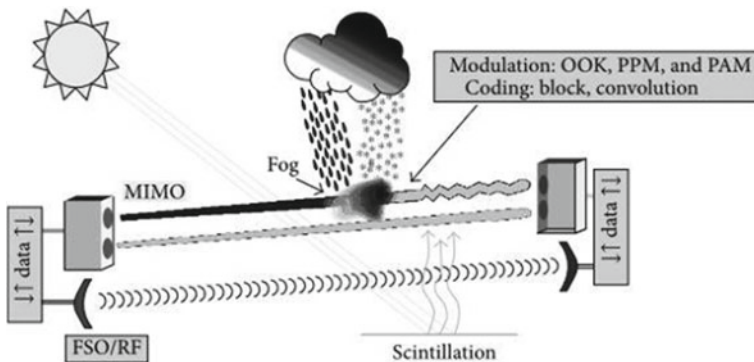


Fig. 3 Atmospheric effects on FSO system

drop in beam intensity over longer distances. Three types of scattering which include Rayleigh, Mie and nonselective scattering happen depending on the particle size [22].

4 Channel Modeling

The channel changes may be interpreted as discrete air or stream units of varying sizes and refractive indices, according to a frequently used turbulence model. Such eddies can be regarded of as lenses in geometrical optics which continuously scatter the light signal forward, resulting in a deformed potency outline at a receiver of the transmission network. Kolmogorov is responsible for the most commonly recognized theory of turbulence [23]. The theory proposes that dynamic power of big turbulent streams, as defined by the external line, is transmitted with zero reduction to smaller streams, as defined by the inner scale, down to dimensions of very small scales. The internal size depicts the size at which viscosity dissipates energy. The refractive index of the turbulent eddies fluctuates randomly, causing phase and amplitude fluctuations in the signal edge. Turbulence may induce optic beam to move randomly, a process known as wandering, and it can produce beam focussing. If an ideal probabilistic framework for the turbulence is applied, the communication link's dependability can be assessed. It is critical to analyze the atmospheric FSO network in order to build a high-performance network connectivity. Due to potency fluctuations in the receiver of an optic connection, many probability density functions are developed. The functionality of an FSO link is harmed by atmospheric turbulence, which causes the received optical signal to fluctuate arbitrarily, resulting in pulse withering. The withering ability is determined by the size of such connection, the wavelength of such light reflection, and a channel's refractive index structural features. The Rytov variance characterizes such structure, which is logically comprehensible. Such turbulence actuated withering is stated deficient when Rytov variance is smaller than 1 and such explicates the authenticity range of the framework. A statistical model is proposed in [24] which discretizes the luminescence as the component of dual unfettered arbitrary actions all with a Gamma PDF. For a range of turbulence condition (weak-strong) the fading gain I in FSO system might be fabricated by a GG distribution.

5 Results Analysis

Considering various channel modes, a signal loss rate of an FSO connection featuring Q-ary PPM with through detection technique is assessed here. Following the analytical approach presented is examined by Matlab. The intensity of every baseband wave is got from ($\frac{1}{2} \times$ optical modulation index) and intensity of such impulse response factor, which is expected for being rectangle as in simulation. The PDF versus irradiation for such lognormal scenario with a known design of scintillation index with

turbulence severity are illustrated in Fig. 4 whereas, Fig. 5 presents PDF for such GG method compared to irradiation. The PDF for the gamma-gamma theory is maximal via an irradiation point of 0.7, as seen in the figure. The PDF chart for the inverse exponent structure is shown in Fig. 6. The optimal level of a PDF is obtained in the negative area, as seen in the chart.

The following are such findings of BER versus average received signal power, with an emphasis on turbulence accentuation of interchannel crosstalk.

Here in those curves in Fig. 7, it can be seen that the three curves are for weak turbulence, medium turbulence and strong turbulence. Here, the model used for

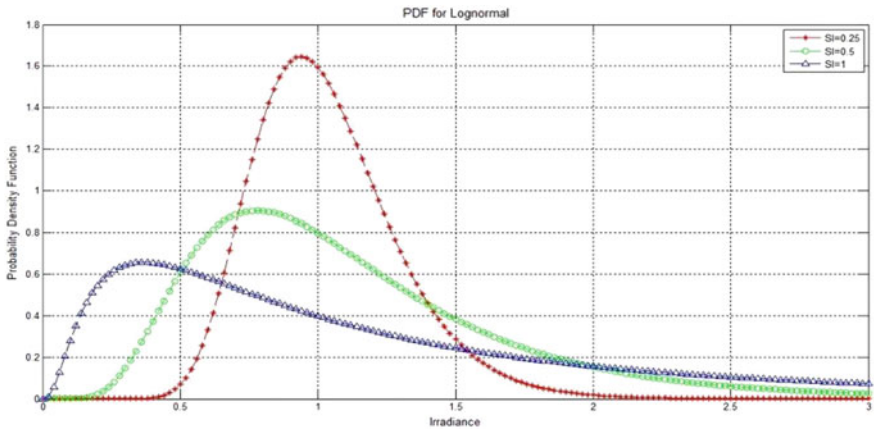


Fig. 4 PDF for lognormal

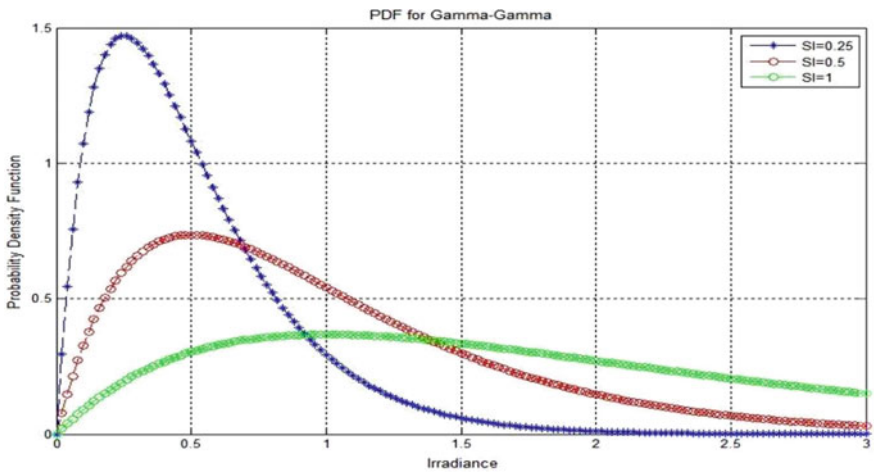


Fig. 5 PDF for Gamma-Gamma

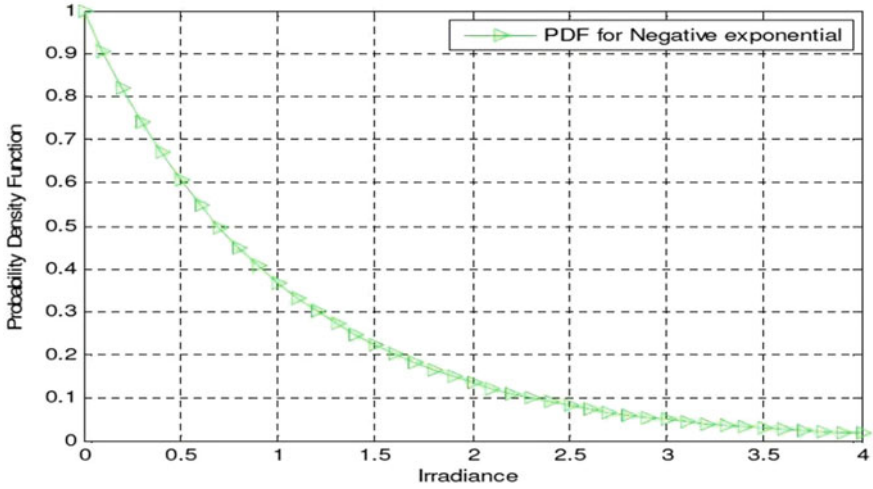


Fig. 6 Curve of PDF versus irradiance for negative exponential structure

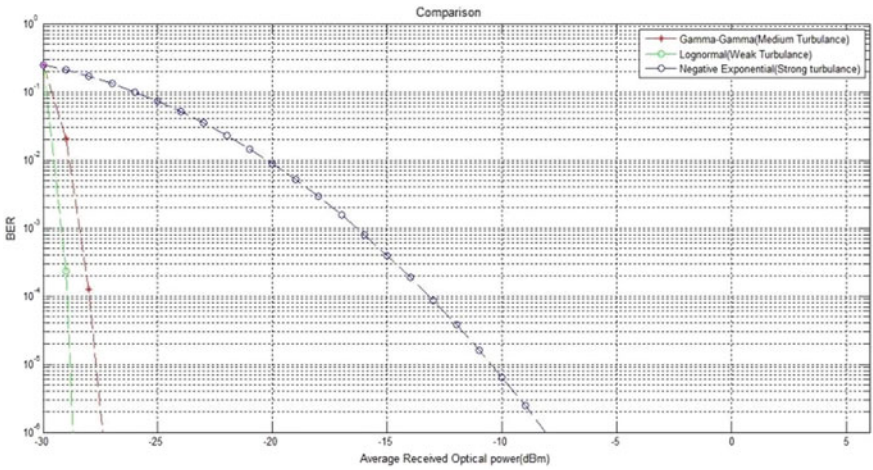


Fig. 7 Comparison among BER and average optical power

weak turbulence is Lognormal model, for medium turbulence the model is Gamma-Gamma model and for strong turbulence negative exponential model is used. Now, for this three model perfect crosstalk ratio is used for each. From this BER vs average received power curve, It's indeed obvious that, the average received energy is -29 dBm, -27.5 dBm and -8 dBm for weak, medium and strong turbulence. Simulation parameters are given in Table 2.

Table 2 Simulation parameters

Parameters	Values
Data rate	100 Mbps
Modulation type	Q-PPM
Sampling frequency	20 MHz
Laser wavelength	850 nm
PIN photodetector responsivity	2
Optical modulation index	1
Scintillation index	1
Symbol energy	10^{-16} J

6 Conclusion

FSO has emerged as a potential high-bandwidth wireless replacement for fiber optic cable. FSO has a number of benefits over fiber, the most important of which are its quick implementation time and considerable cost reductions. The downside of FSO over fiber is that laser power attenuation via the weather is unpredictable and hard to estimate, because it is weather airports, the connection accessibility as a function of distance can be anticipated for any FSO system. In this study, the main focus is to examine and assess the execution of FSO optic access model. Investigation of their performance in detail, with a focus on BER is also presented here. The simulation results in the intermediate turbulence area are excellent agreement with the predicted behavior and fit in between weak and strong turbulence theories. An analysis of the BER and signal to noise ratios for various turbulence levels reveals that aperture averaging can considerably enhance the link's performance, particularly as the turbulence becomes stronger. The information given is useful in directing receiver layout for FSO communication systems.

References

1. G.P. Agrawal, *Fiber-Optic Communication Systems*, 3rd edn. (Wiley, New York, 2002)
2. A.G. Bell, On the production and reproduction of sound by ligh. *Am. J. Sci.* **20**, 305–324 (1880)
3. O. Bouchet, H. Sizun, C. Boisrobert, F. De Fornel, *Free-Space Optics: Propagation and Communication* (Wiley, New York, 2010)
4. R. Bansal, The aftermath of 9/11. What light through yonder window breaks? [free-space laser link]. *IEEE Antennas Propag. Mag.* **44**, 146 (2002)
5. D. Killinger, Free space optics for laser communication through the air. *Opt. Photonics News* **13**, 36–42 (2002)
6. D.K. Berman, Lasers, broadband wireless hookups speed data around lower manhattan. *Wall Street J.* (2001)
7. D. Borah et al., A review of communication-oriented optical wireless systems. *EURASIP J. Wirel. Commun. Netw.* **2012**, 1–28 (2012)

8. M. Singh et al., A long-haul 100 Gbps hybrid PDM/CO-OFDM FSO transmission system: impact of climate conditions and atmospheric turbulence. *Alexandria Eng. J.* **60**, 785–794 (2021)
9. N.A.M. Nor et al., Experimental analysis of a triple-hop relay-assisted FSO system with turbulence. *Opt. Switch. Netw.* **33**, 194–198 (2019)
10. M. Yasser, T. Ismail, A. Ghuniem, FSO Communication with Nonzero Boresight and Jitter over Málaga Atmospheric Turbulence, in *2020 22nd International Conference on Transparent Optical Networks*, Bari (2020), pp. 1–5
11. M.A. Esmail, Experimental performance evaluation of weak turbulence channel models for FSO links. *Opt. Commun.* **486**, 126776 (2021)
12. J. Jeyarani, S.D. Kumar, B.E. Caroline, Performance analysis of PolSK MIMO–FSO over strong atmospheric turbulence conditions, in *2019 TEQIP III Sponsored International Conference on Microwave Integrated Circuits, Photonics and Wireless Networks*, Tiruchirappalli (2019), pp. 432–434
13. X. Li, X. Zhao, P. Zhang, S. Tong, BER performance of FSO communication system with differential signaling over correlated atmospheric turbulence fading. *China Commun.* **17**, 51–65 (2020)
14. N. Xiaolong, et al., Experimental study of the atmospheric turbulence influence on FSO communication system, in *2018 Asia Communications and Photonics Conference*, Hangzhou (2018), pp. 1–3
15. D. AnandKumar, R.G. Sangeetha, Performance analysis of power series based MIMO/FSO link with pointing errors and atmospheric turbulence, in *2021 International Conference on COMMunication Systems & NETWORKS*, Bangalore (2021), pp. 78–81
16. K. Prabu, S. Cheepalli, D.S. Kumar, Analysis of PolSK based FSO system using wavelength and time diversity over strong atmospheric turbulence with pointing errors. *Optics Commun.* **324**, 318–323 (2014)
17. P.V. Trinh, N.T. Dang, A.T. Pham, All-optical relaying FSO systems using EDFA combined with optical hard-limiter over atmospheric turbulence channels. *J. Lightwave Technol.* **33**, 4132–4144 (2015)
18. S.R. Mugunthan, Decision tree based interference recognition for fog enabled IOT architecture. *J. Trends Comput. Sci. Smart Technol.* **2**, 15–25 (2020)
19. V. Suma, W. Haoxiang, Optimal key handover management for enhancing security in mobile network. *J. Trends Comput. Sci. Smart Technol.* **2**, 181–187 (2020)
20. S.A. Al-Gailani, A.B. Mohammad, R.Q. Shaddad, Enhancement of free space optical link in heavy rain attenuation using multiple beam concept. *Optik* **124**, 4798–4801 (2013)
21. H.A. Fadhil et al., Optimization of free space optics parameters: an optimum solution for bad weather conditions. *Optik* **124**, 3969–3973 (2013)
22. S.A. Zabidi, et al., Investigating of rain attenuation impact on free space optics propagation in tropical region, in *2011 4th International Conference on Mechatronics*, Kuala Lumpur (2011), pp. 1–6
23. A.N. Kolmogorov, On the Shannon theory of information transmission in the case of continuous signals. *IRE Trans. Inf. Theory* **2**, 102–108 (1956)
24. H. Willebrand, B.S. Ghuman, *Free Space Optics: Enabling Optical Connectivity in Today's Network* (SAMS, Indianapolis, 2001)

Sentiment Analysis of Unstructured Data Using Spark for Predicting Stock Market Price Movement



Miss Dhara N. Darji, Satyen M. Parikh, and Hiral R. Patel

Abstract In this digital era, social media generates a large quantity of online financial data, which includes a substantial amount of investor sentiment. On the other hand, only technical and fundamental indicators are no longer adequate to forecast the stock price movement. The investors' sentiments on social media likes, tweets on twitter, comments and post on Facebook as well as other online financial information like online news, google trend, and forum discussion are also affecting the stock price movement. In particular, researchers have gained a lot of interest for analyzing the financial tweets on Twitter and online financial news to study public sentiments. This would be extremely helpful to develop an efficient solution for automating the sentiment analysis of such vast quantities of online financial texts. Henceforth, the proposed sentiment analysis model aims to predict the stock price movement based on the unstructured data like financial tweets on Twitter and news data used, and this research work also introduces Spark NLP-based text preprocessing pipeline to remove noise data and extract the features using the TFDIF by organizing the text in structured format. For sentiment analysis, two library Textblob and Vader are used. Further, the performance comparison has been carried out. The main aim of the proposed sentiment analysis model is to understand the perspective of the writer from a piece of text whether it is positive, negative, or neutral. In an extensive manner, news and tweets about a security will certainly inspire individuals to invest in that company's stocks, and as a result, the company's stock price will increase.

Keywords Spark NLP pipeline · Sentiment analysis · Data preprocessing · Stock price movement · TFIDF · Textblob · Vader · Logistic regression · Naïve Bayes · Random forest

M. D. N. Darji (✉) · H. R. Patel
DCS, Ganpat University, Mehsana, India
e-mail: dnd01@ganpatuniversity.ac.in

H. R. Patel
e-mail: hrp02@ganpatuniversity.ac.in

S. M. Parikh
FCA Ganpat University, Mehsana, India
e-mail: satyen.parikh@ganpatuniversity.ac.in

1 Introduction

Text mining is the process of autonomously extracting unique, non-trivial information from unstructured text sources by combining data mining, machine learning (NLP), information retrieval, and knowledge management approaches. Popular text mining tasks include classification of documents, summarization, clustering of similar documents, extraction of concepts, and sentiment analysis. Recently, text mining has leveraged a large variety of applications. To carry out the proposed experimental study, Spark NLP has been utilized.

As social media grow more popular and reach a wider range of users, the data available on these sites gradually represents the real life and the market [1] Since this data is available in an unstructured manner, it is highly required to organize it and utilize the data to infer about future relationships between markets and opinions shared in the network [2]. As a data source, the Twitter microblogging site and financial news are used for sentiment analysis and big data platform Apache Spark is used for text preprocessing and identify the correlation between stock and social media data [3].

The natural language processing (NLP) is considered as a key component in several data science systems that require an understanding about a text. The popular use cases are question answering, language modeling, paraphrasing, and sentiment analysis. In the broader field of NLP, there are several more libraries, but here, we emphasized on general-purpose libraries and not on the ones that cater to particular use cases. The only Spark NLP is a single unified solution to include all the NLP and all-in-one solution to ease the burden of preprocessing text and link the dots between multiple phases to solve the NLP-related challenges in data science. Spark NLP is developed on top of Apache Spark, and Spark ML is an open-source natural language processing library, which covers several popular NLP tasks, including tokenization, speech tagging, stop-word removal, lemmatization and stemming, sentiment analysis, text classification, spell checking, named entity recognition, and more. The core components of the Spark NLP are annotators, pipelines, transformers, and pre-trained models.

- Term frequency–inverse document frequency (TF-IDF);
- Spark’s machine learning (ML) library (Spark MLlib);
- Spark’s natural language processing (Spark NLP);
- Logistic regression (LR);
- Random forests (RF);
- Naïve Bayes (NB).

2 Related Work

Derakhshan et al. [4] discuss about the growth of social media sites, which have provided space for many individuals to share their views. This research work

discusses about the most sensitive field in the world is financial market, where people can share their opinion, and it changes the trend of the overall market. In fact, there are several variables that influence the movement of the stock market, and one of them is the sentiment of investors, who drive the market.

Haddi et al. [5] analyze the importance of the text preprocessing in the sentiment analysis, and the resulted outcome shows that the appropriate feature extraction, and interpretation has improved the accuracy of the sentiment analysis using SVM. They also point out that sentiment analysis is a daunting field to obtain valuable insights from the opinion has expressed on social media requires natural language processing.

The work proposed in [6] by Ashish Pathak et al. discusses about the advantages of implementing machine learning on historical data and sentiment analysis on news headlines that builds the fuzzy logic module, which improves the accuracy of the stock market predication and also describes the limitation of the conventional stock market analysis methodology. This research uses text preprocessing techniques on textual data that is news headlines and finds out the most effective feature, which is categorized as positive and negative.

Elagamy et al. observed in [7] that data mining approaches using historical data to anticipate stock price movement are limited to making judgments within the context of current knowledge and are unable to detect random stock market activity or give triggers behind events. Thus, this research added that the huge financial data available on textual format focuses on the random behaviors of the stock market events, but this data is unstructured so that the text mining is applied on it to provide combined approach of random forest (RF) algorithm with text mining (TM) to study the critical indicator for predicting the abnormal movement of the stock market.

Wang et al. [8] have used social media mining technology to quantitatively determine the market segment and forecast the short-term stock price movement in conjunction with the other indicators.

Ho et al. [9] stated that the sentiments of financial news play an important role in investors' decision-making processes.

Das et al. [10] claim that the sentiment analysis of public's opinion obtained from social media feeds can be used to predict individual stock price fluctuations in the future.

Pagolu et al. [11] reported that incorporating Twitter sentiment analysis adds useful data to the prediction model and enhance accuracy. There is a strong correlation between the rise and fall of the company stock price and public opinions expressed on twitter via tweets about that company.

Pradha et al. [12] conducted research in which they proposed a suitable preprocessing approach for textual information, which plays a significant role in the classifier's prediction accuracy and efficiency while utilizing unstructured data. To remove the noise from the data and rendering such unstructured data into organized and meaningful, text data preprocessing is considered as one of the successful methods. This research work compares various preprocessing techniques for the textual data and their effect on the generated sentiment.

3 Proposed Model

The proposed approach is built on the Apache Spark big data framework, in which Spark NLP is used for data preparation and Spark MLlib is used to categorize news and Twitter data using a machine learning algorithm. The news data collected from the Moneycontrol Web site for BSE top 100 stock companies is based on the market capitalization for 2010–20, and the Twitter data is collected by using the Python Tweepy API. The specific company data or security-wise tweets data is collected by passing the relevant hashtags.

The Spark NLP pipeline created for the data preprocessing includes different phases. The clean text data is converted into the vector by using the TF-IDF for feature selection and extraction, and finally, the Textblob and Vader library are used to check the sentiment impact of the news and tweets. Finally, machine learning algorithms, logistic regression, naïve Bayes, and random forest are applied to classify the data as positive, negative, and neutral for performing sentiment analysis. Furthermore, the accuracy score generated from different libraries as mentioned is compared.

4 Phases of Sentiment Analysis

Sentiment analysis: This is called as logical mining of text, which distinguishes and extricates the abstract data in source material and helping a context to comprehend the social estimation of their phenomena or administration while observing on the web discussion. Henceforth, in stock market, the insiders and outsiders give some extent of information to understand the market movement, which really gives improvement in the prediction of price movement. In the proposed model, the expert and user reviews are observed in terms of news feed and tweets. To classify the reviews in terms of positively added statement, complement statement and no effect statement, the sentiment analysis method is used. The most widely recognized content characterization device examines an approaching message and tells whether the basic supposition is positive, negative, or neutral. It is also helpful to investigate the intention behind the reviews given by users. For stock market, the proposed model is utilized for handling both structured and unstructured data. Structured data preprocessing is followed by statistical data analytics and preprocessing. Unstructured data is processed by using sentimental analysis (Fig. 1).

The sentiment analysis is incorporated by using the following phases.

Tokenization: Tokenization is the common task involved in natural language processing (NLP). Tokenization is generally considered as a way for separating a piece of text into smaller units called tokens. Here, tokens can be either words, characters, or sub-words. The sentences or tweets are converted into words.

Stemmer: It is fundamentally eliminating the postfix from a word and decrease it to its root word. For instance: “Moving” is a word and its addition is “ing”, in the event

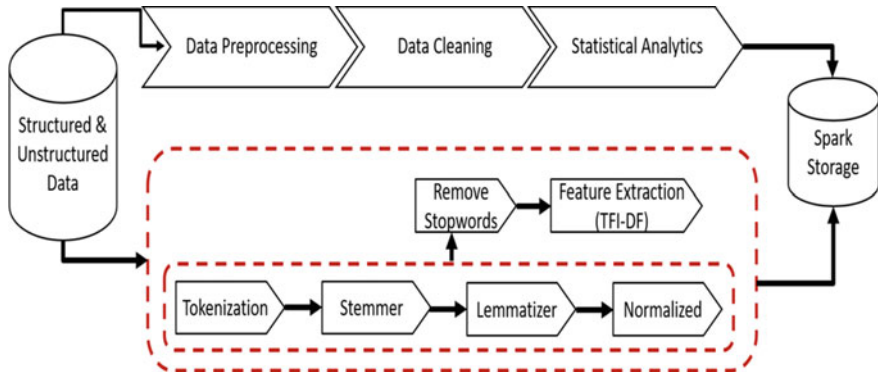


Fig. 1 Proposed model using Spark NLP

that we eliminate “ing” from “Moving” at that point we will get base word or root word which is “Move.” Model utilizes these additions to make another word from unique stem word.

Lemmatization: For linguistic reasons, the records will utilize various types of a word, for example, sort out, puts together, and arranging. Also, there are groups of derivationally related words with comparative implications, for example vote-based system, majority rule, and democratization. As a general rule, it appears as a quest for one of these terms and it might be beneficial in restoring the reports that contain another word in the set. The goal of stemming and lemmatization is to reduce inflectional structures and occasionally the derivationally related word types to a standard base structure. The result of this text mapping will be something like: The stock’s drives are different, and the stock drive may differ.

Normalized: Text standardization is the way toward changing a content into an accepted (standard) structure. For instance, “gooood” and “gud” can be changed to “acceptable,” its standard structure. Another model involves planning the related indistinguishable words, for example “stopwords,” “stop-words,” and “stop words” to simply “stopwords.”

Stop Word Removal: Stop words are a bunch of normally utilized words in a language. Stop words are ordinarily utilized in text mining and natural language processing (NLP) to dispense the words that are so generally utilized to convey next to no helpful data.

Feature Extraction (TF-IDF): In data recovery, TF-IDF or term recurrence converse record recurrence is a mathematical measurement that is planned to reflect how significant a word is to an archive in an assortment or corpus. The TF-IDF weight is a weight regularly utilized in data recovery and text mining.

5 Model Implementation

```

model_pipeline = Pipeline(
    stages=[document_assembler,
            tokenizer,
            normalizer,
            stopwords_cleaner,
            stemmer,
            finisher,
            hashingTF,
            idf,
            label_stringIdx,
            ml_classifier,
            label_to_stringIdx])
sentimentmodel = model_pipeline.fit(traindata).transform(testdata)

```

The above pipeline contains three parts.

The first part contains document_assembler, tokenizer, normalizer, stopwords cleaner, stemmer, and finisher, which is the process of implementing Spark NLP for data cleaning and data preprocessing.

Second part contains hashingTF, idf, and ml classifier, which is process of Spark MLlib for the implementation of machine learning, feature extraction as well as the implementation of machine learning algorithms for sentiment classification.

label_stringidx and label_to_stringIdx just for the string labeling.

The last part is the implementation of single execution plan for the testing and training data.

6 Results and Discussion

This section shows the results of a sequence of stages of the text preprocessing and steps for building the Spark NLP pipeline. Spark NLP comes with a number of annotators and transformers, and it also seamlessly integrates with Spark MLLib to build a data preprocessing pipeline.

Step 1: Initialize Spark.

Step 2: Load the Twitter and news data.

Step 3: Build NLP pipeline using Spark NLP [This pipeline can include feature extraction modules like HashingTF and IDF and machine learning classifier model].

Step 4: Implement and evaluate the model.

Original Text: In Spark NLP, the first stage transforms raw data into document type for further process. A special transformer `DocumentAssembler()` with desired parameters is used for that.

Tokens: The next stage identifies tokens with tokenization standards. `Tokenizer` annotator splits the documents into token according to the parameters like min max width of the token, case sensitivity of the text, and character list that is used to separate from the inside of tokens based on the patterns it will be separated from inside tokens and many more.

Normalized Text: This stage removes all dirty characters from text using the normalizer annotator, which has followed the regex pattern and transform the words based on the provided directory.

Clean Tokens: This stage obtains the clean tokens by removing the stop words using the `StopWordsCleaner` because in NLP process these are useless words.

Stemmer Text: This stage performs stemming process for removing a part of a word or reducing a word to its stem or root and for that stemmer and annotator is used.

Token Features: This is an important stage, where NLP pipeline is ready to go, we might as well put our annotation results to use somewhere else. Annotation values are the output obtained by the finisher as a string.

TF Features: Finally, the feature token form the documents. The TF-IDF feature extraction technique converts the token into vectors. TF and IDF are implemented in `HashingTF` and `IDF`. This stage utilizes the `HashingTF` to convert the documents to fixed size vectors.

Features: The last and final stage generates the inverse document frequency. The `minDocFreq` variable of the `IDF` supports filtering out terms, which do not appear in a minimum number of documents. For terms that are not in at least `minDocFreq` documents, the `IDF` is found as 0, resulting in TF-IDFs of 0.

7 Sentiment Analysis Comparison

This research work utilizes company wise NEWS data of BSE 100 stock company based on the market capitalization. It is difficult to represent scratch view of all stock modeling, where the paper show top 5 companies [Reliance Industries Ltd, Tata Consultancy Services Ltd, HDFC Bank Ltd, Infosys Ltd, and Hindustan Unilever Ltd.] data (Figs. 2, 3 and 4).

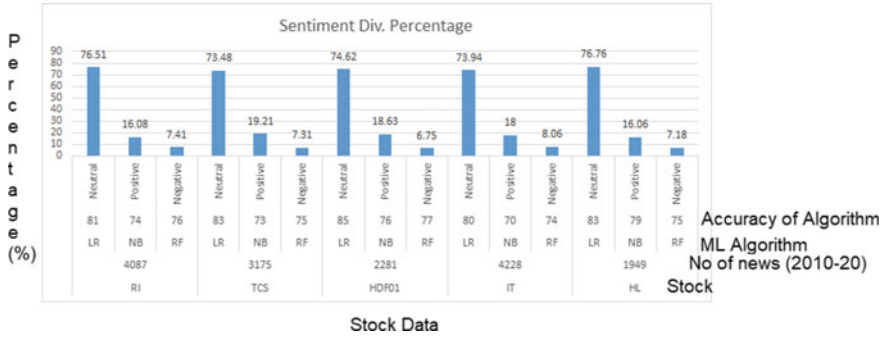


Fig. 2 Top 5 stock company sentiment analysis using Textblob

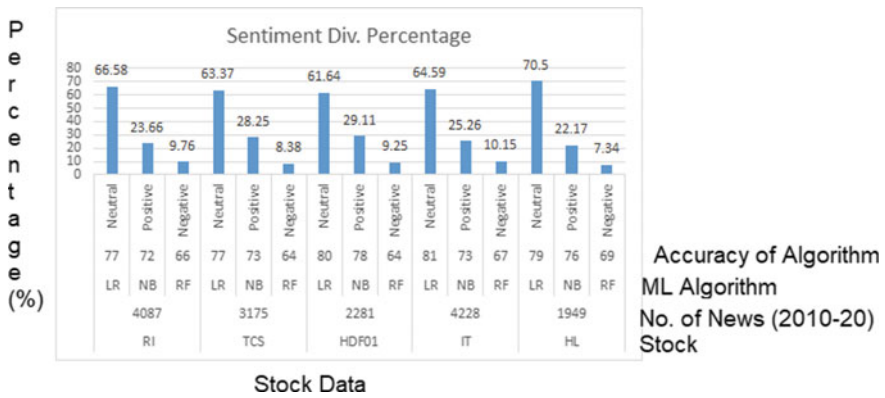


Fig. 3 Top 5 stock company sentiment analysis using Vader

8 Conclusion and Future Scope

The proposed research study has successfully implemented the sentimental analytics for performing stock market predictive analysis. This paper clearly shows the methods and model implementation of sentiment on Twitter’s tweets and news for stock. The proposed model includes a script to fetch online news from online verified sources and tweets from twitter, and then, the text normalization is performed, and finally, Textblob and Vader library are used to obtain the sentiment impact score of the financial text. After that, the sentimental analysis steps are incorporated and each step results are discussed. According to experiments, the Spark NLP gives best performance for calculating the sentiments from text. As mentioned, in this experimental study, different algorithms and libraries were applied. As per the result study, the Textblob library gives better result in the context of sentiment analysis. After generating sentimental results, the machine learning algorithms were applied and among that logistic regression gives accuracy between 80 and 85% for all companies

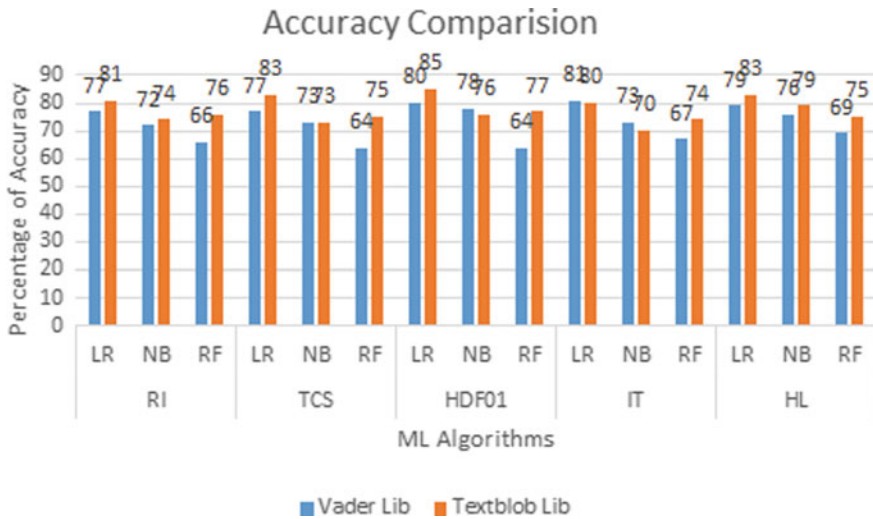


Fig. 4 Top 5 stock company sentiment analysis accuracy comparison for Textblob and Vader

stock. The experiment is carried out with sample data if more data is applied with advance Spark NLP methods than increasing the accuracy.

References

1. T.H. Nguyen, K. Shirai, J. Velcin, Sentiment analysis on social media for stock movement prediction. *Expert Syst. Appl.* **42**(24), 9603–9611 (2015). <https://doi.org/10.1016/j.eswa.2015.07.052>
2. A. Romanowski, M. Skuza, Towards predicting stock price moves with aid of sentiment analysis of Twitter social network data and big data processing environment. *Adv. Bus. ICT: New Ideas Ongoing Res.* **658**, 105–123 (2016). https://doi.org/10.1007/978-3-319-47208-9_7
3. C. Lee, I. Paik, Stock market analysis from Twitter and news based on streaming big data infrastructure, in *2017 IEEE 8th International Conference on Awareness Science and Technology (iCAST)* (2017). <https://doi.org/10.1109/icawst.2017.8256469>
4. A. Derakhshan, H. Beigy, Sentiment analysis on stock social media for stock price movement prediction. *Eng. Appl. Artif. Intell.* **85**, 569–578 (2019). <https://doi.org/10.1016/j.engappai.2019.07.002>
5. E. Haddi, X. Liu, Y. Shi, The role of text pre-processing in sentiment analysis. *Procedia Comput. Sci.* **17**, 26–32 (2013)
6. A. Pathak, N.P. Shetty, Indian Stock Market prediction using machine learning and sentiment analysis. *Adv. Intel. Syst. Comput. Comput. Intel. Data Mining* 595–603 (2018). https://doi.org/10.1007/978-981-10-8055-5_53
7. M.N. Elagamy, C. Stanier, B. Sharp, Text mining approach to analyse stock market movement, in *The International Conference on Advanced Machine Learning Technologies and Applications (AMLTA2018) Advances in Intelligent Systems and Computing*, pp. 661–670 (2018). https://doi.org/10.1007/978-3-319-74690-6_65

8. Y. Wang, Y. Wang, Using social media mining technology to assist in price prediction of stock market, in *2016 IEEE International Conference on Big Data Analysis (ICBDA)* (2016). <https://doi.org/10.1109/icbda.2016.7509794>
9. K. Ho, W. Wang, Predicting stock price movements with news sentiment: an artificial neural network approach. *Artif. Neur. Netw. Model. Stud. Comput. Intel.* **628**, 395–403 (2016). https://doi.org/10.1007/978-3-319-28495-8_18
10. S. Das, R.K. Behera, M. Kumar, S.K. Rath, Real-time sentiment analysis of twitter streaming data for stock prediction. *Procedia Comput. Sci.* **132**, 956–964 (2018). <https://doi.org/10.1016/j.procs.2018.05.111>
11. V.S. Pagolu, K.N. Reddy, G. Panda, B. Majhi, Sentiment analysis of Twitter data for predicting stock market movements, in *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)* (2016). <https://doi.org/10.1109/scopes.2016.7955659>
12. S. Pradha, M.N. Halgamuge, N.T. Vinh, Effective text data preprocessing technique for sentiment analysis in social media data, in *2019 11th International Conference on Knowledge and Systems Engineering (KSE)* (2019). <https://doi.org/10.1109/kse.2019.8919368>

Intrusion Detection and Prevention Using RNN in WSN



Ashok Yadav and Arun Kumar

Abstract The wireless sensor network involves sensor nodes, communicating protocols, and gateways for interaction with the Internet. Due to limited memory availability in wireless sensor network, the advanced encryption algorithm of securities and authentication protocol is not deployable due to which wireless sensor networks are prone to attacks such as distributed denial of service and distributed denial of service attacks. The intrusion detection and prevention is used to detect, notify malware activities, avoid, and stop them. The proposed system is mainly to detect and prevent the distributed denial of service and denial of service attack in wireless sensor network. In the proposed model, recurrent neural network is taken as a classifier. The model is validated using the ten-fold cross validation in nine is to one repeated iteration and is then tested for making of false positive alerts on data set (WSN-DS). The accuracy of this model is 99.8% with positive fault rate of 0.3%.

Keywords Recurrent neural network · WSN-DS · SVM · Random forest · Decision tree · CNN

1 Introduction

The wireless sensor networks have many applications in the field of detection of the air quality, volcanoes, earthquake, flood, health care, and in telecommunication. The wireless sensor network has physical insecurities, limited processing power, less availability of memory, and no well-defined boundaries; i.e., the boundaries are changed on the movement of either device or users; due to these, the wireless sensor network is prone to threat. The distributed denial of service attack and denial of service attacks are easily deployable attack in the wireless sensor network [1]. The intrusion detection and prevention mechanism is more important because in wireless sensor network the implementation of advanced encryption algorithm, large

A. Yadav (✉) · A. Kumar
Centre for Advanced Studies, AKTU, Lucknow, Uttar Pradesh, India

A. Kumar
e-mail: draran@cas.res.in

authentication protocols, and other cryptographic algorithm are not feasible [2]. The intrusion detection is done easily using the concept of deep learning as well as machine learning techniques. The wireless network mainly threatens in areas such as attack on sensors, attack on the network service, and attacks on the application services.

The attacks which mainly occur on sensors are location tracking, device cloning, and physical attacks. Similarly, attacks on network service are routing attacks and on application services are distributed denial of service, denial of service attack, and eavesdropping etc. The intrusion detection main aim is to avoid compromises to confidentiality, and availability [3]. Due to the advancement in the field of IOT devices, automation system results in the smart parking system, automated homes, smart cities, smart traffic light system, smart electric meters, and sensors nodes, etc. These are interconnected with communication protocols, and gateway is used to connect with Internet due to which the securities attack increased [4]. The devices which are used in wireless sensor network mainly have less memory and also depend upon the batteries.

These devices have almost negligible security because of lack of deployment of encryption algorithm, antivirus, and other cryptographic techniques. The propositioned tactic is centered on the anomaly intrusion detection, and their prevention with the recurrent neural networks as classifier and validation of the model takes place using tenfold cross-validation mechanism on the wireless sensor network data set (WSN-DS). The feature which is generally used for classification is the abnormal traffic on network, data transfer rate, etc. The proposed model easily detects the attacks in the network. The recurrent neural network is trained for detecting the attack such as user to root attack, remote to local, denial of service, and distributed denial of service attack. Some artificial neural network-based mechanism is proposed such as backpropagation which is not lightweight and attack type is flooding attack whose accuracy is closer to 90%, and the feedforward algorithm which is lightweight in nature and attack type which is malicious node, and accuracy which is almost 95%. The remaining paper is described as follows. Section 2 describes the related work, Sect. 3 describes the proposed methodology, Sect. 4 involves the result section of the paper, and Sect. 5 has the conclusion of the paper.

2 Related Work

One of previously proposed models for detection of denial of service attack and the KDD Cup99 data set is used, and this model is capable of detecting the flooding attack and denial of service attack with higher precision and accuracy [5]. Papers [6] and [7] have proposed a model in which the artificial neural network is used for detection of the intrusions. The KDD Cup99 data set is used, and the feature selection takes place using backpropagation algorithm. This model is suitable for the real-time applications also, and with this, gray-hole attack as well as denial of service attack is easily detected with higher accuracy. Papers [8] and [9] have proposed a model for intrusion

detection using artificial neural network, and in this, the classification can be done using backpropagation algorithm, and the KDD Cup99 data set is used for training and testing purposes. Papers [5] and [10] have defined a model in which layered categories are used for the classification purpose to detect intrusions and the artificial neural network as well as the support vector machine and KDD Cup99 data set is used in the implementation of model. Papers [11] and [12] have proposed a model in which the machine learning classifier such as random forest and artificial neural network are used for the classification, detection and prevention of network-based intrusion respectively. Papers [8] and [13] have given a model in which the machine learning classifier such as decision tree is for the classification and artificial neural network is for detection and prevention of network-based intrusion. Paper [14] has proposed a technique for intrusion detection and classification of the attacks with help of the artificial neural network. In this, the multi-layer perceptron architecture is used. The KDD Cup99 data set is used for training and testing the model, and it detects various attacks and after that classifies in into six different clusters. Paper [15] proposed model for detection of network intrusion with the help of the multi-layer perceptron architecture and the artificial neural network. In this, some relevant features of attacks are used instead of all features of the packet. The model accuracy is better in case of detection of denial of service attack. Papers [16] and [17] proposed a model which is based on feature-reduced intrusion detection, and it analyzed important features of data dimensionality reduction take place then the reduced features are feed to feed-forward neural network for training and testing using the KDD Cup99 data set, and this model uses artificial neural network classify normal and abnormal data. In papers [15] and [18], the given model for detection of intrusion in wireless sensor network is based on the mechanism of the genetic programming. The genetic programming involves gene-expression mechanism, linear genetic programming mechanism, and multi-programming mechanism for the detection of the intrusions, and the accuracy of the model is more than 95%. In papers [2] and [19], another model is proposed which is totally based on the fuzzy logic for intrusion detection in wireless sensor network. In this, the author claims that using this model, all types of the intrusions are detected easily with accuracy of 100%. Papers [14] and [20] have given a mechanism which is based on the concept of rule-based decentralized mechanism which detects the different type attacks of the wireless sensor network such as black hole attack, worm hole attack, and selective hole attack. The accuracy of the model is better, and the positive fault rate is minimal. Papers [6] and [21] have proposed a model which is based on the concept of the clustering mechanism. In this, the detection of intrusions takes place on the basis of differentiating between the abnormal traffic on the network and normal traffic on the network. Have proposed one of the models in which support vector machine is used as a classifier and for training and testing the model using distributed learning algorithm is used. Papers [5] and [22] have proposed one of the models in which decision tree is used as a classifier, and for training and testing, distributed learning algorithm is used. Papers [3] and [11] have proposed one of the models in which convolution neural network is used as a classifier, and for training and testing, distributed learning algorithm is used. Papers [4] and [10] have proposed one of the models in which random forest is used as a classifier, and for training and

testing, the distributed learning algorithm is used. The detection of malicious file in this model is more accurate and also applicable in real-life scenario. One of models is proposed in which deep learning algorithm is used. In this, the fog node used is of high bandwidth and power of computation enhanced the deployment of the deep learning services. The farmer's get more information about their crop, and also the quality of life of farmers is improved. The result of the proposed work shows that accuracy of the model is good [23]. One model is proposed for addressing the data mining chaos such as scalability, security and privacy, and efficiency. The complexity of the model is linear in nature due to which the model is more efficient. The model provides more resistant to the system from attacks, and also, the accuracy of model is better [24]. A technique to decide highest quality time and highest quality fee to withdraw a voluntary retirement scheme thinking about chance of recognition of a retirement request of a retirement request fee because of saying voluntary retirement scheme and to the enterprise because of one-time unique bills to folks that voluntarily retire for the duration of the term is discussed. A specific case wherein a Poisson n a Poisson manner is believe for the statement of the voluntary retirement scheme [25]. One of the models is proposed which helps in the identification of the name of the resources which are allocated in the cloud. The mechanism used is round robin and first come first serve for minimizing the cost of demand and time [26].

3 Method and Material

In the intrusion detection and prevention system proposed involves the following stages such as feature extraction, classifier, training and testing, data set, and decision. At the stage of feature extraction, some features are extricated from the provided data and used as a feature and also some features are mixed with other features and considered as single feature for classification with the help of which the classification result accuracy is improved. The next stage is of classifier, and recurrent neural network is used as a classifier. In the papers [19] and [8], training and testing are done using the WSN-DS. The resilient backpropagation learning strategy is applied for training neural network in which rate of learning is 0.01, and to train, 1000 epochs are used. According to received data at classifier stage, the classification take place and then result is forwarded at the decision stage and decision stage decisions are made either data packet is accepted or rejected and automatically notify at the base station. The given model in the paper [13] intrusion detection and prevention system uses the only header of the data, but in this, both header and the payload of the data are considered for making decision due to which the accuracy of the model is enhanced [13]. The anomaly-based intrusion detection system is mainly compromises of only two phases that is training phase and testing phase. In this, the deviation between the perceived behavior and the model is regarded as an abnormality and the feature selection is considered during the training phase of the recurrent neural network [27]. The ability of learning from data set depends upon neural network used, and categorizing the file or packet coming through network as abnormal or normal will

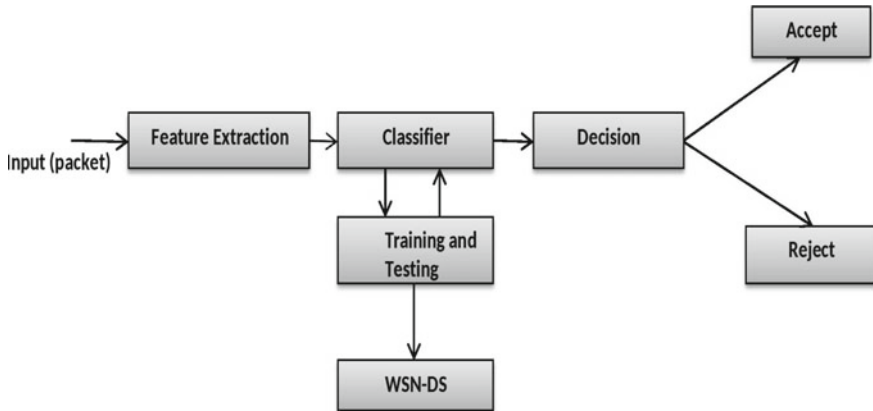


Fig. 1 Model of IDPS using RNN

be done by some computing techniques of neural network. The network traffic data is collected using image, library files, log file, dynamic link, and other files such as log file, music file, and word file documents (Fig. 1).

4 Result and Discussion

In our proposed model, recurrent neural network is trained using the WSN-DS data set with help of the tenfold cross validation method in nine is to one iteration with two hidden layers and three hidden layers. The classification of the attacks classified correctly up to 98.6% with two hidden layers and error is approximately 0.0343, when three hidden layer are used for the classification of attacks take place correctly up to 98.34 with error of 0.0643. In case of the using two hidden layers, at first layer, the number of neurons used is 11, and in the second layer, the number of neurons is five, and in case of three hidden layers, the number of neurons at first layer is 11, and at second hidden layer, five neurons are used, and at last hidden layer, the number neurons used is two. The number of passes or epochs used through training data is 1000. The proportion of validation set from the data used for training is 20%, the learning rate in proposed model is used for the adjustment of the weight at each iteration, and the learning rate of this model is approximately 0.3, and the momentum of model is used for adjustment of weight during the backpropagation in order to prevent local minima and speed up convergence, and momentum of this model is 0.2. The tenfold cross-validation method is used in (9:1) repeated manner with the help of which the accuracy of classification is enhanced. Some of the term is used for showing the result of the model which is the true negative means of the number of normal attacks that are classified as normal (no attack), as well as false negative, which refers to the number of attack cases that are wrongly classified as normal (no

attack), and the false positive which means the normal (no attack) cases classified incorrectly as attack. The rate of true positive and false positive is calculated with the help of formulae (Figs. 2, 3 and 4; Tables 1 and 2).

$$(TPR = (TP / (TP + FN)))$$

$$(TNR = (TN / (FN + TP)))$$

$$(FPR = (FP / (FP + TN)))$$

The receiver operating characteristic curve is used to describe the total distinction of the classification model. If the area under the curve is high, then it means that the classifier used is better. In the above ROC curve, bold blue indicates the norm of receiver operating characteristics curve of all 500 iterations of the repeated tenfold cross-validation, and the gray-shaded area directs the extent of the receiver operating characteristic curve produced over all iterations. The dashed red line in the curve indicates the ability of the classifier that is the accuracy of the classification of files either it is malicious or no malicious to which class it belongs to at random a baseline for the worst case class. In another way the red dashed line is the base line for worst case classification performance.

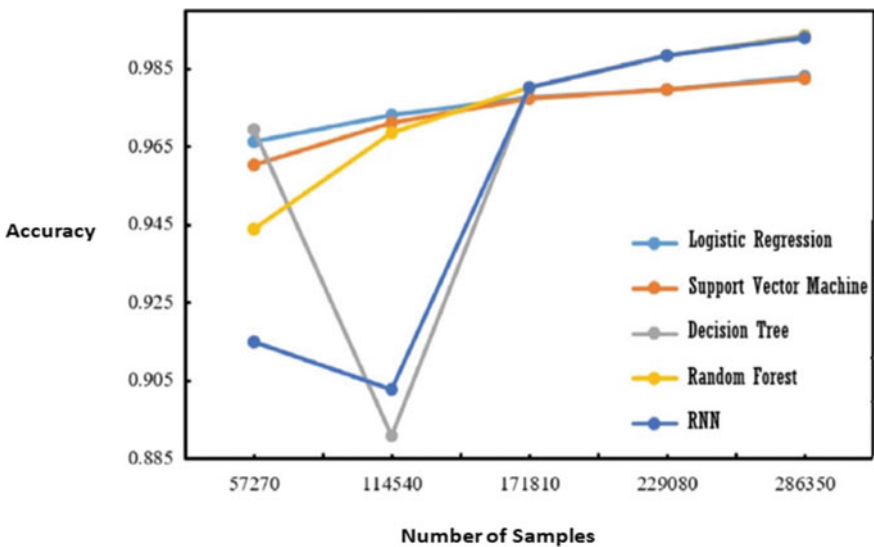


Fig. 2 Training of model using multiple classifiers

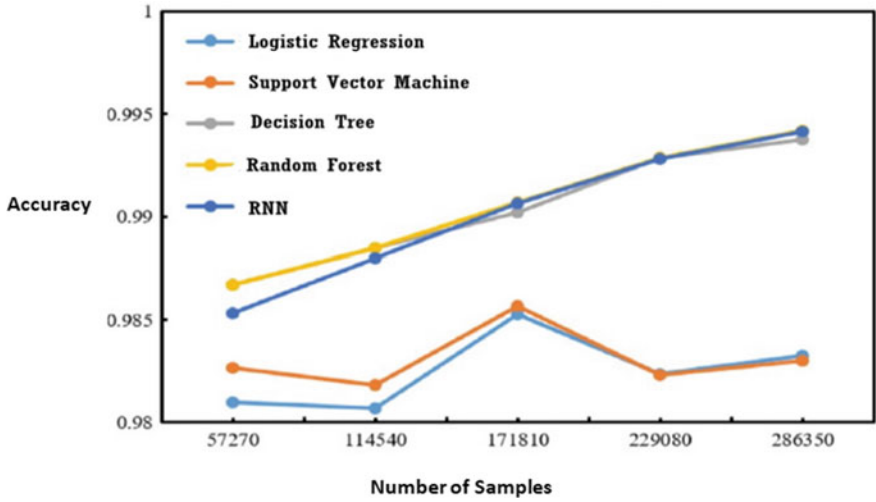


Fig. 3 Testing ROC curve using multiple classifiers

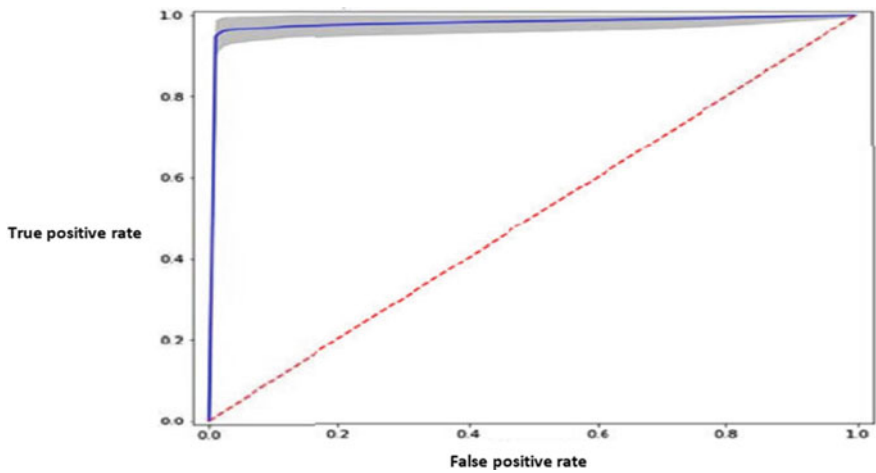


Fig. 4 ROC curve of RNN

Table 1 Result using two hidden layer

	TPR	FPR	FNR	TNR	P
Normal	0.998	0.02	0.002	0.98	0.998
DDOS	0.979	0.006	0.211	0.899	0.899
DOS	0.924	0	0.079	1	0.979
Flooding	0.898	0.002	0.005	0.979	0.969
Scheduling	0.798	0.02	0.004	0.897	0.991
AVG	0.985	0.004	0.014	0.962	0.999

Table 2 Result using three hidden layer

	TPR	FPR	FNR	TNR	P
Normal	0.984	0.046	0.007	0.896	0.995
DDOS	0.843	0.013	0.157	0.987	0.938
DOS	0.769	0.01	0.311	0.99	0.946
Flooding	0.789	0.001	0.219	0.977	0.976
Scheduling	0.0.874	0.002	0.196	0.988	0.989
AVG	0.969	0.041	0.028	0.959	0.963

5 Conclusion

The main aim of intrusion detection system is to avert compromise to CIA triads of security model of the system. In the proposed method, RNN is used as classifier using which the classification of malicious and non-malicious file is detected. The data set used is WSN-DS which is created using the leach protocol. A WSN-DS data set consists of 17 attributes and 374,000 rows. The accuracy of the model is better with two hidden layer in detection of distributed denial of service attack and denial of service attack with positive fault rate of 0.3. The validation of model is done using of tenfold in nine is to one repeated iteration mechanism due to which the fault rate is minimal and the accuracy is better. The flooding attack, gray-hole attack, and other attacks are also detected with better accuracy.

References

1. H. Mi, Z. Wang, A. Ittycheriah, Supervised attentions for neural machine translation. EMNLP 2016—Conf. Empir. Methods Nat. Lang. Process. Proc. **4**, 2283–2288 (2016). <https://doi.org/10.18653/v1/d16-1249>
2. V. Jyothisna, V.V. Rama Prasad, K. Munivara Prasad, A review of anomaly based intrusion detection systems. Int. J. Comput. Appl. **28**(7), 26–35 (2011). <https://doi.org/10.5120/3399-4730>
3. E.G. Dada, J.S. Bassi, O.O. Adekunle, *An Investigation Into the Effectiveness of Machine Learning Techniques for Intrusion Detection*, vol. 13, no. 6, pp. 764–778 (2017). Available: www.azojete.com.ng
4. J. Asharf, N. Moustafa, H. Khurshid, E. Debie, W. Haider, A. Wahab, A review of intrusion detection systems using machine and deep learning in internet of things: Challenges, solutions and future directions. Electron **9**(7) (2020). <https://doi.org/10.3390/electronics9071177>
5. A. Saeed, A. Ahmadinia, A. Javed, H. Larjani, Random neural network based intelligent intrusion detection for wireless sensor networks. Procedia Comput. Sci. **80**, 2372–2376 (2016). <https://doi.org/10.1016/j.procs.2016.05.453>
6. R. Krishnan, Y.H. Robinson, E.G. Julie, *An Intrusion Detection and Prevention Protocol for Internet of Things Based Wireless Sensor Networks*, pp. 0–18
7. K.A. Molinaro, M.L. Bolton, Evaluating the applicability of the double system lens model to the analysis of phishing email judgments. Comput. Secur. **77**, 128–137 (2018). <https://doi.org/10.1016/j.cose.2018.03.012>

8. O.E. Elejla, B. Belaton, M. Anbar, A. Alnajjar, Intrusion detection systems of ICMPv6-based DDoS attacks. *Neural Comput. Appl.* **30**(1), 45–56 (2018). <https://doi.org/10.1007/s00521-016-2812-8>
9. R. Chen, J. Gaia, H.R. Rao, An examination of the effect of recent phishing encounters on phishing susceptibility. *Decis. Support Syst.* **133**, 113287 (2020). <https://doi.org/10.1016/j.dss.2020.113287>
10. M.A. Rezvi, S. Moontaha, K.A. Trisha, S.T. Cynthia, S. Ripon, Data mining approach to analyzing intrusion detection of wireless sensor network. *Indones. J. Electr. Eng. Comput. Sci.* **21**(1), 516–523 (2021). <https://doi.org/10.11591/ijeecs.v21.i1.pp516-523>
11. Y. Fu, Z. Yan, J. Cao, O. Koné, X. Cao, An automata based intrusion detection method for internet of things. *Mob. Inf. Syst.* **2017**, 6–10 (2017). <https://doi.org/10.1155/2017/1750637>
12. M. Jakobsson, Modeling and preventing phishing attacks. *Lect. Notes Comput. Sci.* **3570**, 89 (2005). https://doi.org/10.1007/11507840_9
13. S. Duque, M.N. Bin Omar, Using data mining algorithms for developing a model for intrusion detection system (IDS). *Procedia Comput. Sci.* **61**, 46–51 (2015). <https://doi.org/10.1016/j.procs.2015.09.145>
14. A. Hendrawan, A.F. Daru, A.M. Hirzan, Intrusion detection with wireless sensor network (WSN) internet of things. *EEE Access* **13**(2), 45–48 (2021)
15. J.P. Ananth, S. Balakrishnan, S.P. Premnath, Logo based pattern matching algorithm for intrusion detection system in wireless sensor network. *Int. J. Pure Appl. Math.* **119**(12), 753–762 (2018). <https://acadpubl.eu/hub/2018-119-12/articles/7/1636.pdf>
16. M. Hasan, M.M. Islam, M.I.I. Zarif, M.M.A. Hashem, Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches. *Internet of Things* **7**, 100059 (2019). <https://doi.org/10.1016/j.iot.2019.100059>
17. A.S. Ahmed, R. Hassan, N.E. Othman, Denial of service attack over secure neighbor discovery (SeND). *Int. J. Adv. Sci. Eng. Inf. Technol.* **8**(5), 1897–1904 (2018). <https://doi.org/10.18517/ijaseit.8.5.6427>
18. L. Alsulaiman, S. Al-Ahmadi, Performance evaluation of machine learning techniques for DOS detection in wireless sensor network. *Int. J. Netw. Secur. Its Appl.* **13**(2), 21–29 (2021). <https://doi.org/10.5121/ijnsa.2021.13202>
19. L. Ashiku, C. Dagli, Network intrusion detection system using deep learning. *Procedia Comput. Sci.* **185**(June), 239–247 (2021). <https://doi.org/10.1016/j.procs.2021.05.025>
20. V. Suryani, S. Sulistyono, W. Widyawan, Two-phase security protection for the Internet of Things object. *J. Inf. Process. Syst.* **14**(6), 1431–1437 (2018). <https://doi.org/10.3745/JIPS.03.0106>
21. N. Kaur, P. Rattan, A critical review of intrusion detection systems in WSN: challenges & future directions. *IDS WSN* **25**(4), 3020–3028 (2021)
22. N.A. Alrajeh, S. Khan, B. Shams, Intrusion detection systems in wireless sensor networks: a review. *Int. J. Distrib. Sens. Netw.* (2013). <https://doi.org/10.1155/2013/167575>
23. K. Lee, B.N. Silva, K. Han, Deep learning entrusted to fog nodes (DLEFN) based smart agriculture. *Appl. Sci.* **10**(4) (2020). <https://doi.org/10.3390/app10041544>
24. W. Haoxiang, S. Smys, Big data analysis and perturbation using data mining algorithm. *J. Soft. Comput. Paradig.* **3**(1), 19–28 (2021). <https://doi.org/10.36548/jscp.2021.1.003>
25. B. Thilaka, N. Theetharappan, Optimal time for withdrawal of voluntary retirement scheme with a time-varying threshold, in *The second International Conference on Innovative Mechanisms for Industry Applications ICIMIA 2020—Conference Proceedings*, vol. 02, no. 04, pp. 598–602 (2020). <https://doi.org/10.1109/ICIMIA48430.2020.9074885>
26. D.W. Haoxiang, D.S. Smys, MC-SVM based work flow preparation in cloud with named entity identification. *J. Soft Comput. Paradig.* **2**(2), 130–139 (2020). <https://doi.org/10.36548/jscp.2020.2.006>
27. A. Martin, N.B. Anuthamaa, M. Sathyavathy, M.M. Saint Francois, D.V.P. Venkatesan, A framework for predicting phishing websites using neural networks. *Int. J. Comput. Sci.* **8**(2), 330–336 (2011). Available: <http://arxiv.org/abs/1109.1074>

Error Evaluation of Short-Term Wind Power Forecasting Models



Upma Singh and M. Rizwan

Abstract Inconsistency and randomness of wind power impose massive challenges to large-scale wind power production. An accurate production of the wind power for the upcoming hours is imperative, in order that accurate planning and scheduling of the wind power production from conventional units can be accomplished. In the present work, we have proposed three intelligent forecasting models using fuzzy logic, artificial neural network (ANN) and adaptive-neuro-fuzzy inference system (ANFIS) approaches. These models can efficiently incorporate the uncertainty and nonlinearity linked with climatic parameters. To implement these models, the forecasting has been done using historical data of various stations. The performance of these intelligent forecasting models are estimated with statistical indicators and observed that the results obtained using ANFIS forecasting model are found quite accurate. Consequently, ANFIS model can be useful for accurate forecasting of wind power and for efficiently utilizing the wind resources.

Keywords ANN · ANFIS · Fuzzy logic (FL) · Renewable energy resources · Wind power

1 Introduction

With the rising industrial and agricultural activities, especially in developing countries like India, enhancing the demand of electricity and also conventional energy sources are in limited amount, so we have to be more responsible in using natural resources in more effective way [1–3]. As rising industrial and agricultural activities also increase environment pollution, hence it is a matter of concern for all growing and developed countries to focus on natural resources [4, 5]. Wind power is growing source of electricity and can significantly ease the problems of global environmental

U. Singh (✉) · M. Rizwan
Delhi Technological University, Maharaja Surajmal Institute of Technology, Delhi, India
e-mail: upma@msit.in

M. Rizwan
e-mail: rizwan@dce.ac.in

pollution [6, 7]. Wind power depends on weather conditions and having intermittent nature may leads to irregularity and uncertainty in wind power output. Also, the conventional sources of energy are also diminishing at a very fast rate. Therefore, it is extremely important to look toward renewable sources such as hydro, wind, solar and biomass [8–11]. Alarmingly, the availability of wind data is scarcely available due to finite spatial coverage, limited length of record and high cost of instrumentation. On account of, inaccessibility of the measured data, forecasting of wind energy is significantly important at the surface of earth. In this regard, it is imperative to construct intelligent systems or models on the basis of easily accessible meteorological data-set to forecast global wind power.

The wind power is clean and world-wide distributed having low cost of power generation [12]. Wind energy-model varies from mathematical models to hybrid models namely persistence model, Kalman filter, ARMA model, etc., have been grown for predicting wind power. Latest research carried out represents that the mathematical models presented in the literature are not providing satisfactory results for all situations is universally accepted, primarily due to highest simplicity of parameterization [13]. As illustrated in literature several techniques for example, linear predictors, exponential smoothing models and gray predictors, etc., have developed and presented for the purpose of wind energy estimation. All these models utilize historical data for modeling, but because of the intermittent nature of wind they cannot yield precise prediction of wind power [14–17].

Presently, with the improvements in the artificial intelligent technique various algorithms like artificial neural network, WPT (wavelet packet transform) and SVM (support vector machines) have been employed for forecasting of wind power [18, 19]. Furthermore, an intelligent hybrid forecasting model relying on Markov model and least square support vector machine has also been integrated for forecasting of wind power. But these techniques are not reliable for real-time implementations due to over computational complexity associated with them which many times reduces the reliability of forecasting [20]. In another research work radial basis function network (RBFN), adaptive neuro-fuzzy inference system (ANFIS) and artificial neural network (ANN) approaches were utilized to compare 1-h ahead prediction of wind power. It was found in that fuzzy logic based algorithm works well even when mathematical model of the network is not obtainable [21]. To enhance the quality of wind power interval prediction, a fuzzy interval prediction model (FIPM) is developed. To optimize the FIPM gravitational search algorithm is used. The experimental output represents that FIPM model gives better performance than traditional forecasting models [22, 23]. In other paper to predict the wind energy output, two-hidden layer neural network is used. By utilizing proper data in combination with a back-propagation algorithm, neural-network model is prepared. Simulation output shows that predicted wind power is in better agreement with the experimentally measured values [24]. The long short term memory (LSTM), rectified linear unit activation function and Adam-optimization algorithm are investigated to perform daily to monthly estimation with the help of recurrent neural network process. It was investigated that a univariate single-layer recurrent neural network architecture is

preferred for wind speed estimation and multilayer recurrent neural network model may be considered for improving the prediction accuracy over a longer period [25].

In recent study, to estimate the performance of wind power of china's 29 provinces and cities from 2011 to 2018, adaptive neuro-fuzzy approach is utilized, and it is found that ANFIS shows a vital improvement in accuracy [26]. Therefore, it has been seen that, a hybrid adaptive neuro-fuzzy model provides better accuracy in forecasting of wind power in comparison with standalone model [27]. For this aim, three modeling techniques, Support Vector Machines, ANFIS and Multi-Nonlinear Regression were utilized. After evaluating the performance, it was seen that modeling based on Subtractive-Clustering provides better outputs than Grid-Partitioning [28]. In other research, Weibull probability density function for estimating the wind speed and power has been used in particular Dakhla and Taza cities [29]. In the present paper, three forecasting models (Fuzzy logic, ANN and ANFIS) have been developed for forecasting of wind power by using wind speed and air density taken as input parameters keeping in view of aforesaid literature. These three wind power forecasting models demand limited amount of dataset.

This paper is structured in seven sections as a concise introduction related to the topic with meticulous literature study has been described in Sect. 1. Section 2 presents the determination of input variables Sect. 3 deliberates the study area and collection of data-set Sect. 4 presents implementation of the fuzzy logic based system for the forecasting of wind power. Section 5 describes the implementation of ANN based model for forecasting of wind power. ANFIS implementation is outlined in Sect. 6. Section 7 deliberates results and discussion. Finally, Sect. 8 presents concluding remarks.

2 Determination of Input Parameters for the Wind Power Forecasting Model

The meteorological input parameters that affect the wind turbine output are identified as wind direction, dew point temperature, speed of the wind, temperature, relative humidity, rainfall, air density and pressure, etc. The meteorological parameter, air density is associated with the change in temperature and relative humidity. However, the other parameters like dew point temperature, pressure and rainfall may impact on the production of wind turbine output, but these factors are not likely to be considered so significant so these meteorological factors are not taken into consideration. Though, the effect of wind speed and air density on wind turbine output production is considered more significant. Thus, speed of wind and air density factor (derived from the Eq. 1) is chosen as input features for developing the wind power forecasting model. The equation of air density parameter is represented as follows:

$$\text{Air Density} = D \left(\frac{B - 0.3783e}{760} \right) \left(\frac{273.15}{T} \right) \quad (1)$$

where, e is the vapor pressure of moist air in torr, B is the barometric pressure in torr, T is the absolute temperature in Kelvin and D is the density of dry air at standard atmospheric temperature (25 °C) and pressure (100 kPa) ($D = 1.168 \text{ kg/m}^3$) [30].

3 Study Area and Collection of Data-Set

As described in previous section, factors that affect the wind turbine output are wind speed, relative humidity and temperature. To attain the aim, a continuous record of all meteorological parameters is needed, which is rarely available. The required data-set was collected from NIWF (National Institute of Wind Energy) and IMD (Indian Meteorological Department), Pune, which is 10 years averaged data for the period January 2008–June 2018 for the study area, New Delhi, Rajasthan, Maharashtra and Chennai [31, 32]. Before applying the data-set to the forecast models as input, the whole data-set were analysed and pre-processed. The normalization of the data-set is done and defined in the range of 0.1–0.9, so as to bypass the convergence problems during operation for four weather stations such as New Delhi, Rajasthan, Maharashtra and Chennai that show different climatic conditions. Approximately, 70% data-set are utilized for training and 30% data-set are utilized for testing purposes.

4 Implementation of Fuzzy Logic Based Model

Fuzzy-systems are comparable to human-decision making having capability to generate reliable and accurate results from imprecise information. In the paper, main motive is to forecast wind power with the help of two input variables, i.e., wind speed and air density. Both parameters are utilized as input to fuzzy logic system and output variable is wind power. Hence, there are two input variables which are utilized to forecast wind power as an output. Later on, input and output variables are normalized by examining all the parameters behavior and stated in the range from 0.1 to 0.9 to prevent or by pass convergence problem throughout the operation. Membership functions are of many types such as trapezoidal, bell shape, triangular and Gaussian membership function and these are designated using trial and error strategy.

To develop the forecasting model, triangular membership function has been selected for input as well as output parameters. Prior to developing the fuzzy rules, we have done partitioning the all parameters range into five regions namely very low, low, medium, high and very high.

The proposed forecasting model use if–then rules. To acquire rule base forecasting, accuracy is checked by using a distinct set of past data. If it is inadequate, then we can change the shape of the membership functions and number of membership functions. Hence, error range is reduced by nearly four percent using fuzzy logic model. The modeling takes into account the uncertainties exhibit in the environment caused by

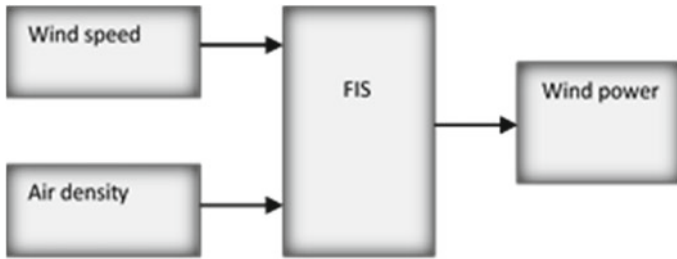


Fig. 1 Fuzzy inference system based model for the forecasting of wind power

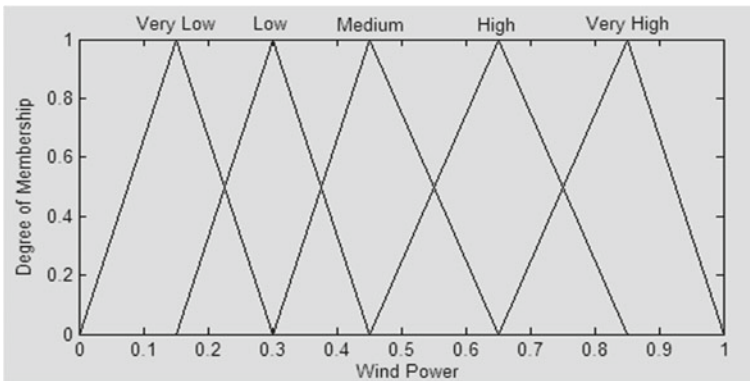


Fig. 2 Fuzzy membership function of air density and wind speed

varying weather conditions [33]. Figure 1 represents fuzzy inference system based model for the forecasting of wind power.

Figure 2 displays the membership functions linked with each of the variables. To specify the shape of all membership function, membership function editor is employed. Fuzzy membership is a curve that represents how each point in the input space is mapped to membership value or degree of membership between 0 and 1. So, forecasting results are accomplished using triangular membership function.

The proposed fuzzy logic based forecasting model is depicted in Fig. 3. Rule editor is used to modify and view the rules, which describes the system performance shown in Figs. 4 and 5 represents fuzzy rule viewer, which is useful in viewing the inference process of the fuzzy system. By adjusting the input values, correspondent output of each fuzzy rule can be viewed.

The display of the fuzzy inference plot comprises of a figure window with seven small plots nested in it. The two small plots across the top of the figure shows the antecedent and consequent of the first rule. Each column is a variables, and each rule is a row of plots.

Also, the first two columns of the plots represents the membership functions referenced by the antecedent or if-part of each rule. The third column of the plots

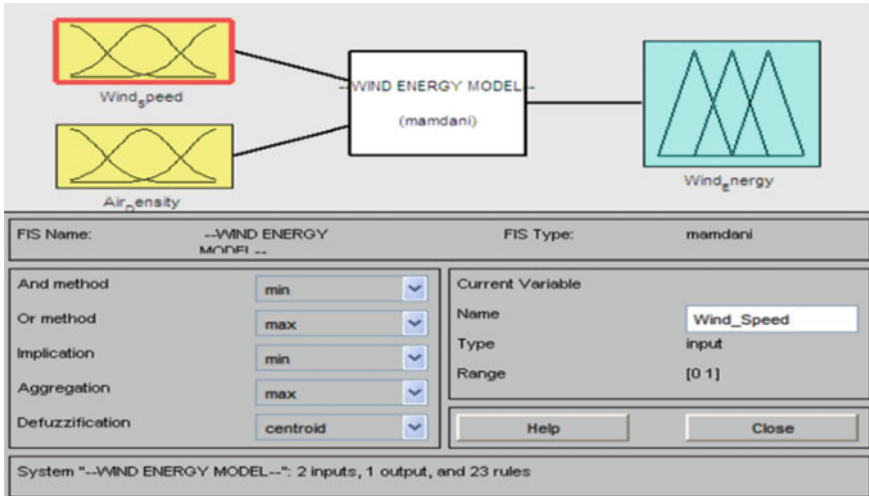


Fig. 3 Fuzzy logic based forecasting model

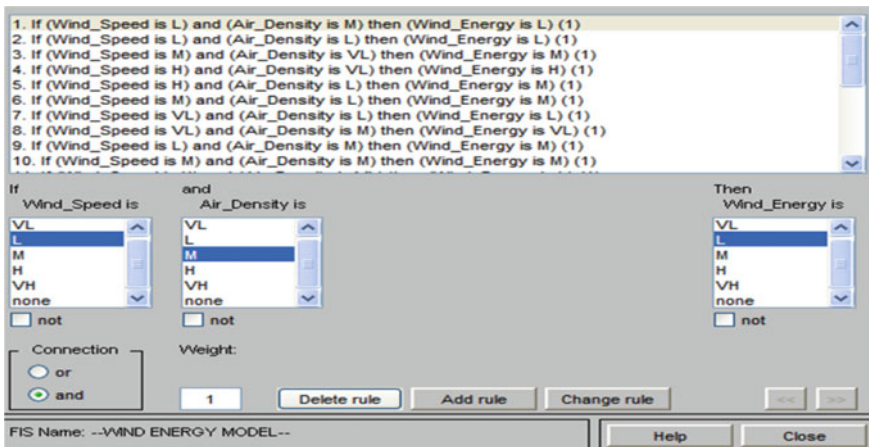


Fig. 4 Rule editor for fuzzy logic based model

represents the membership functions referenced by the consequent, or then-part of each rule. The last plot in the third column shows the aggregate-weighted decision and the bold vertical line on it shows the defuzzified output.

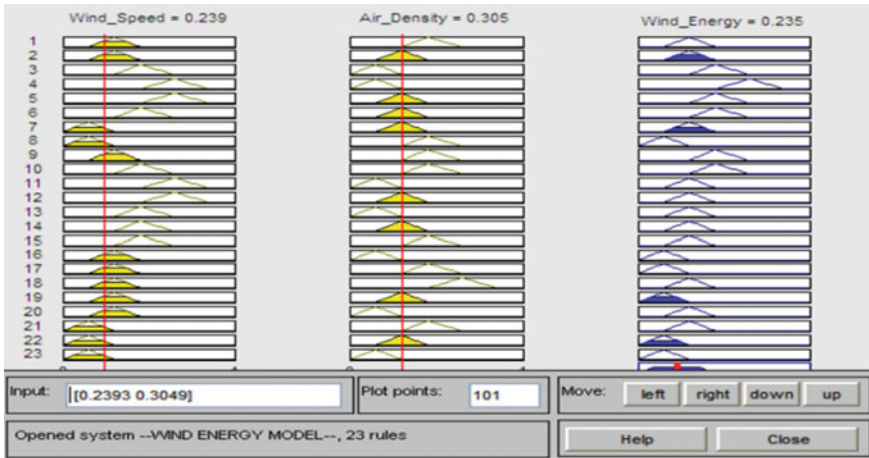


Fig. 5 Fuzzy rule viewer for the forecasting of wind power

5 Implementation of ANN Based Model

In this section, forecasting technique using ANN tool is implemented to estimate the produced energy from wind turbine with the help of neural network trained by using past data. An artificial neural network is used in forecasting, on account of its ability of approaching nonlinear mapping between numbers of inputs and outputs and dig out unidentified data or information within the huge available data. ANN is distributed in data storage and computing in terms of structure. Hence, model developed by artificial neural network retain robustness and capability of solving troublesome problems. This section uses excellent nonlinear mapping ability of neural network to forecast wind energy at distinct stations namely New Delhi, Rajasthan, Maharashtra and Chennai by utilizing the data-set of wind speed and air density.

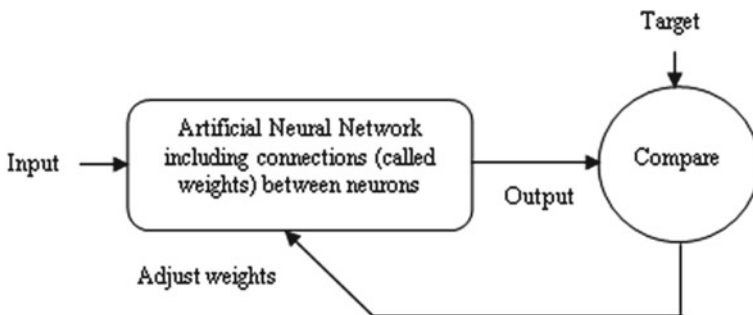


Fig. 6 Operating scheme of artificial neural network

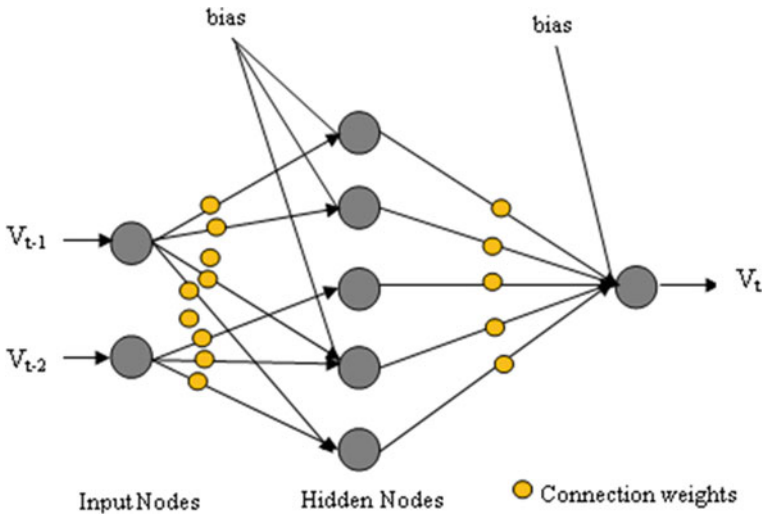


Fig. 7 Architecture of a feed-forward neural network having 2-input nodes, 5-hidden nodes and 1-output node

The basic operating scheme of artificial neural network is represented in Fig. 6. For development of the model all data has been divided into two parts, i.e., training data-set and testing data-set, for this 70% data-set is utilized for training and 30% data-set is utilized to test the model.

Figure 7 presents the architecture of a feed-forward neural network having 2-input nodes, 5-hidden nodes and 1-output node. In this network, the information (data) will enter through the input nodes to output nodes by means of hidden nodes. Here, the first layer having two input parameters (i.e., wind speed and air density), a hidden layer has tansigmoid function “tansig” which is expressed by the equation represented as:

$$f(x) = (1/(1 + \exp(-x))) \tag{2}$$

where, input is represented by x . The output layer having linear activation “purelin” transfer-function which would solve hard problems. To implement a neural network algorithm, neural network toolbox has been used in Matlab. The network output is shown below:

$$y = \sum_{j=1}^n w_{ij}x_{ij} + \theta_i \tag{3}$$

where w_{ij} is connection weights directed from j neuron to i neuron at the hidden-layer, θ_i is the i neuron bias, x_{ij} is the j_{th} neuron incoming signal at the input-layer.

Table 1 The ANN model properties

ANN model parameters	Type or value
Input number	2
Output number	1
Number of hidden layers	1
Number of hidden-neurons	8
Transfer-function of output layer	Purelin
Transfer-function of hidden layer	Tansig
Training cycles, epochs(e)	30
Learning rate	0.01
Optimization method	Feed-forward back-propagation
Scaling method	Normalization

Artificial neural networks have a built in capability to adapt their synaptic weights with respect to the varying atmosphere condition. It also has ability to carried out tasks that a linear program cannot [34, 35].

It produces good results by using enormous amount of data. ANNs learns with the help of examples and they can be programmed to carry out a specified task. Neural networks are also fault-tolerant; therefore, these networks handles insufficient and noisy data [36]. They also has ability to handle nonlinear problems so, they can be used in forecasting problems once trained [37]. Feed-forward back-propagation network is applied in this proposed work and for training and adaptation of the neural network TRAINLM training function with LEARNGDM adaptive learning function has been utilized to develop the forecasting model (Table 1).

6 Implementation of ANFIS Based Model

From the previously implemented techniques, it is observed that neural network is a flexible and powerful approach for modeling several real world problems, but it has few limitations, such as if input data-set are ambiguous or highly uncertain than fuzzy system like adaptive neuro-fuzzy inference system may be a favorable approach. In addition, ANN also involves huge data-set to train, selecting adequate number of hidden-units, input and output samples.

In the given segment, ANFIS tool has been utilized in the Matlab software to train and test by utilizing “anfisedit” function in the command to forecast wind power.

In 1993, Jang first evolved the ANFIS method and successfully implemented its principles to several problems. Neuro-fuzzy system is used in wide range of scientific applications due to its several features such as accurate and fast learning, robust generalization capabilities, easy to implement, great explanation possibility with fuzzy rules. By combining the advantages of neural network and fuzzy logic,

adaptive neuro-fuzzy inference system can solve any kind of nonlinear and complex problems efficiently. Neuro-fuzzy is basically the rule based fuzzy modeling. Fuzzy-rules are molded via the process of training. The training process is done by utilizing a data-set. The neuro-fuzzy designs a fuzzy inference system and based on the training dataset, parameters of membership-functions are designed.

In neuro-fuzzy system, neural networks take out automatically fuzzy-rules from the numerical data and membership functions get adaptively adjusted with help of learning action. It has multi-layered feed-forward architecture. This architecture is generally comprised of five layers in which first and forth layer encompass adaptive nodes whereas, other layers comprises of fixed nodes [38]. Process of fuzzification is carried out in the first layer having adaptive neurons. Second layer having fixed neurons, represent incoming signals. Third layer consists of fixed neurons (nodes) wherein every node is stable marked as M shows fuzzy rules. The forth layer having adaptive neurons showing the rule inference and the fifth layer which is output layer marked as " \sum ", i.e., summation neuron. On the basis of previous past data as utilized for earlier models, adaptive neuro-fuzzy model has been constructed to forecast wind energy [39, 40]. By adopting the given procedure neuro-fuzzy model has been constructed. First of all, given past data is partitioned into two sections one for training purpose (70% of data-set) and other section utilized for testing operation (30% of data-set). In neuro-fuzzy system, there are M^n fuzzy-rules where, membership function is denoted by " M " and number of inputs are denoted by " n ." In this model, 5 membership functions and 2 inputs are selected. Therefore, number of fuzzy-rules are 25. For each input variable, 5 membership functions are formed. To train FIS, two distinct optimization methods (back-propagation and hybrid) are used. For developing the model gauss2mf membership function is used because it gives superior outcomes in contrast to other membership functions for the given data-set. While output membership function is selected to be linear as shown in Fig. 10. In this model, the grid partitioning is selected to generate FIS as the forecasting accuracy obtained is more compared to the subtractive clustering. The parameters of the neuro-fuzzy model are given in Table 2. Later on, neuro-fuzzy model has been checked and validated after the successful training operation by using the testing data. Validation of the proposed model was done using statistical indicators. Using the above procedure ANFIS model was evolved. It is essential to indicate that the number of input sets and the number of rules to be composed increases if the number of variables utilized to execute the forecasting increases. The neuro-fuzzy system comprises of hybrid and back-propagation leaning algorithms that reduces the error between forecasted and observed data. To develop the model, both algorithms are used to compare the outcomes of them. Minimization of error was achieved by using learning process.

Figure 8 shows the flowchart of training and testing neuro-fuzzy model. Training error in ANFIS is given in Fig. 9. Figure 11 shows ANFIS architecture comprises of five layers. It is a feed-forward neural network, which comprises of fuzzification-layer, rule-layer, normalization-layer, defuzzification-layer and a summation neuron. All the nodes are adaptive nodes in the first layer. Output of neuro-fuzzy model structure is shown in Fig. 12 (Fig. 10).

Table 2 The ANFIS model properties

ANFIS model parameters	Type or value
Input number	2
Output number	1
Type of fuzzy inference system	Sugeno
Number of input membership function	5 5
Input membership function type	gauss2mf
Output membership function type	Linear
Optimization method for training FIS	Grid partition
Optimization method	Hybrid; back-propagation
Training epoch numbers	500

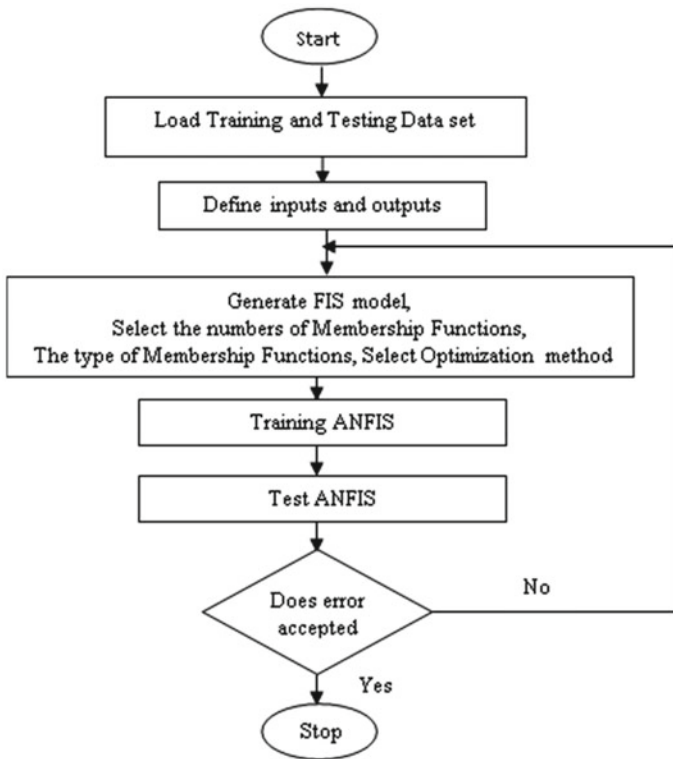


Fig. 8 Training and testing ANFIS method flowchart

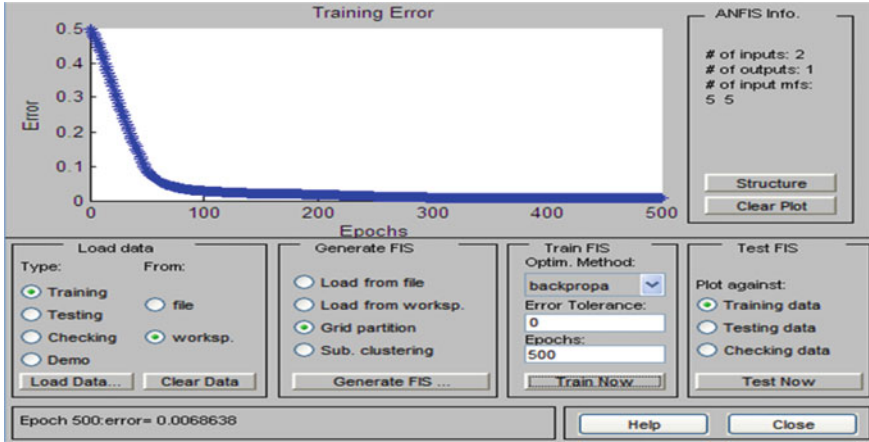


Fig. 9 Training error in ANFIS

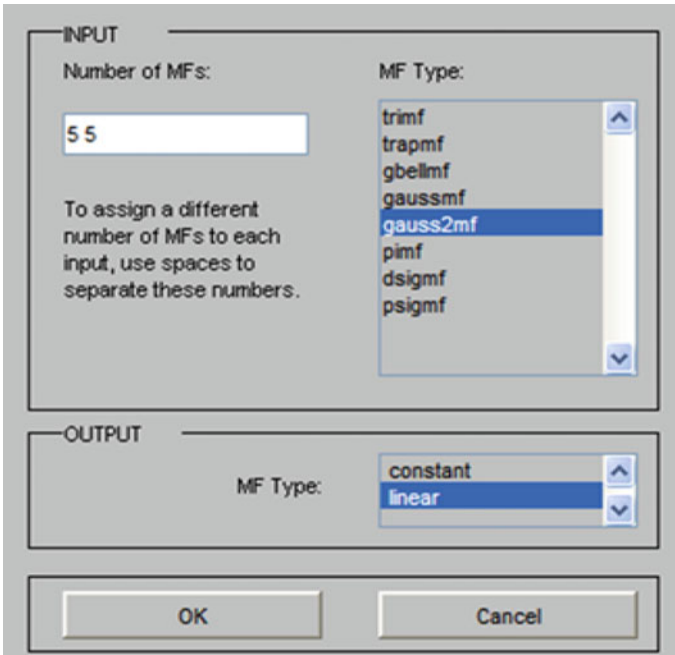


Fig. 10 Initial ANFIS generation

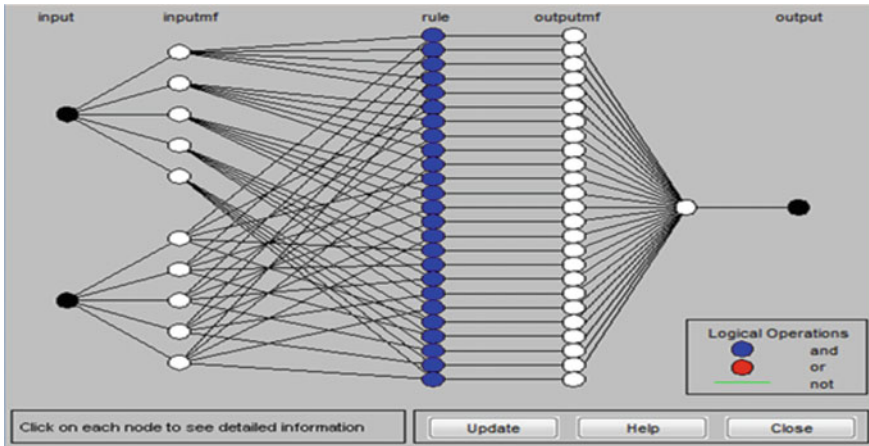


Fig. 11 ANFIS model structure

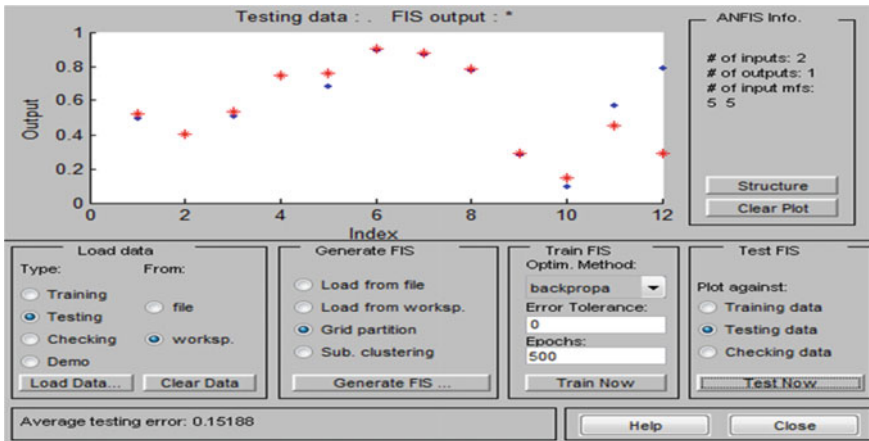


Fig. 12 Output of ANFIS model structure

7 Results and Discussions

In this research work, the FL, ANN and ANFIS based methodologies are employed for developing forecasting models to estimate wind turbine output. The meteorological input parameters, wind speed and air density considered in implementing the models to estimate the wind turbine output power. For developing the all three models, normalized data has been utilized.

Figures 13, 14 and 15 depict the month by month comparison between forecasted wind power using fuzzy logic, neural network and neuro-fuzzy techniques and those calculated from the measured data, respectively.

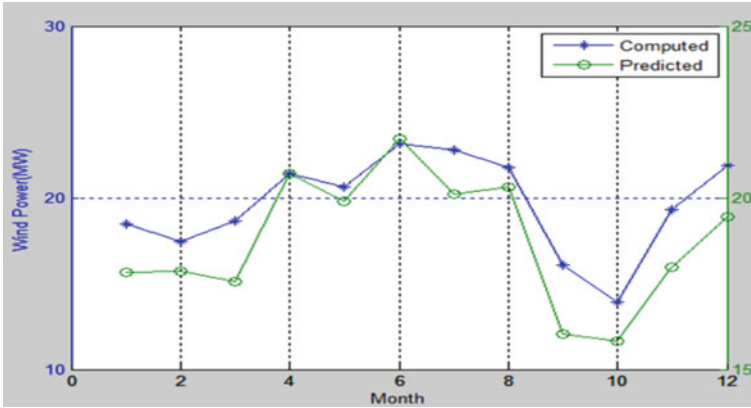


Fig. 13 Predicted values of wind power in comparison with computed power with FL



Fig. 14 Predicted values of wind power in comparison with computed power with ANN

As we can infer from Fig. 15 that mostly forecasted data are overlapping and substantially near to the computed data. Therefore, it can be inferred that the proposed neuro-fuzzy model gives better results than the other two forecasting models. The forecasted wind power in comparison with computed values of wind power with FL, ANN and ANFIS techniques are displayed in Table 3. From Table 3, it is quantifies that ANFIS hybrid intelligent system provides best results for forecasting of wind power. Hence, neuro-fuzzy based model provides a stipulated mathematical arrangement that makes it an excellent adaptive approximator. Additionally, neuro-fuzzy system delivers good learning ability and minimizes convergence error for a network of same complexity and displays supremacy to the ANN technique and other techniques of same complexity.

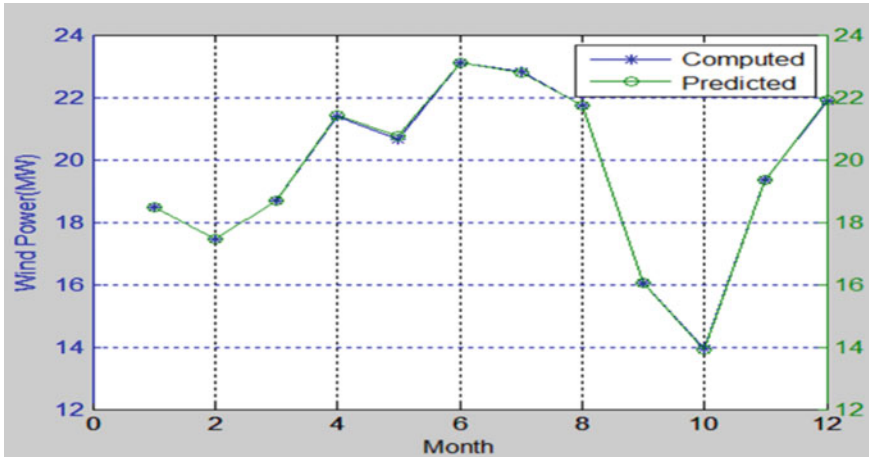


Fig. 15 Predicted values of wind power in comparison with computed power with ANFIS

Table 3 Monthly mean forecasting wind power in comparison with computed power

Months	Wind power (computed), MW	Wind power (forecasted), MW		
		FLs	ANNs	ANFISs
Jan	18.491	17.83	18.49	18.52
Feb	17.542	17.85	17.46	17.54
Mar	18.584	17.57	18.68	18.73
Apr	21.471	20.69	21.42	21.57
May	20.552	19.89	20.77	20.77
Jun	23.021	21.74	23.10	23.19
Jul	22.802	20.11	22.80	22.80
Aug	21.816	20.31	21.73	21.82
Sep	16.143	16.04	16.07	16.14
Oct	13.640	15.83	13.92	13.58
Nov	19.362	17.98	19.35	19.48
Dec	21.361	19.45	21.90	21.11

There are many assessment measures to estimate the performance of all given models on the basis of various widely used statistical indicators such as absolute relative error (ARE), standard deviation of error (SDE), the sum squared error (SSE), the mean absolute percentage error (MAPE).

It can be inferred with the help of statistical indicators results for forecasting of wind power, ANFIS based forecasting generates minimum errors when compared to the other intelligent forecasting models. Hence, ANFIS model provides quite favorable results than fuzzy logic and ANN models. Table 4 summarizes performance

Table 4 Performance of the developed models for wind power forecasting based on diferent statistical indicators

Model	Statistical Indicators			
	ARE	MAPE	SSE	SDE
FL	4.60	14.225	0.0540	0.0543
ANN	0.07	0.0838	2.331e−006	3.964e−004
ANFIS	0.034	0.0227	2.0417e−007	1.2822e−004

evaluation of FL, ANN and ANFIS models for the forecasting of wind power with regard to statistical indicators.

The MAPE, SSE and SDE criterion are stated below:

$$MAPE = \frac{100}{N} \sum_{h=1}^N \frac{|\hat{p}_h - p_h|}{\bar{p}} \tag{4}$$

$$\bar{p} = \frac{1}{N} \sum_{h=1}^N p_h \tag{5}$$

where, N is the number of forecasted hours, \bar{p} is the average wind power of the forecasting period, \hat{p}_h and p_h are the forecasted and actual wind power at hour h .

$$SSE = \sum_{h=1}^N (\hat{p}_h - p_h)^2 \tag{6}$$

$$SDE = \sqrt{\frac{1}{N} \sum_{h=1}^N (e_h - \bar{e})^2} \tag{7}$$

$$e_h = \hat{p}_h - p_h \tag{8}$$

$$\bar{e} = \frac{1}{N} \sum_{h=1}^N e_h \tag{9}$$

where \bar{e} is the average error of the forecasting period and e_h is the forecast error at hour h .

From the results of Table 4, it has been observed that the forecasting results of all three intelligent models based on statistical indicators shows that ANFIS model gives better performance than the other two models (fuzzy logic and ANN).

8 Conclusion

For effective wind power harnessing, the reliable and precise wind resource evaluation plays a significant role. In this study, FLs, ANNs and ANFISs based models are developed and analyzed for forecasting of wind power at four distinct places. These models have two input variables, i.e., wind speed and air density and one output variable, i.e., wind power. A comparative study was also conducted to validate functioning of all intelligent models is determined by using statistical indicators. ARE, MAPE, SSE and SDE, respectively, are 4.60, 14.225, 0.0540 and 0.0543% for fuzzy logic, are 0.07, 0.0838, $2.331e-006$ and $3.964e-004\%$ for ANN, are 0.03, 0.0227, $2.0417e-007$ and $1.2822e-004\%$ for ANFIS. The results demonstrated superior performance of ANFIS model in contrast to the other two intelligent forecasting models (FLs and ANNs). Hence, ANFIS based model could be important tool to forecast wind power. The forecasting of wind power would be practically utilized for optimization, real-time power dispatch, power smoothening, selection of appropriate energy storage and requirements of additional generating stations. Such forecasting would be useful for handling demand and supply for power building in a smart-grid environment, which may reduce the problems of power fluctuations generated from wind based energy systems. For the development of smart energy management system, this work will help the stakeholders such as designer, operation engineer, service provider, utility, technocrats and power engineer.

References

1. Z. Jinhua, Y. Jie, W.U. Wenjing, L.I.U. Yongqian, Research on short-term forecasting and uncertainty of wind turbine power based on relevance vector machine. *Energy Procedia* **158**, 229–236 (2019)
2. W.Y. Chang, A literature review of wind forecasting methods. *J. Power Energy Eng.* **2**(04), 161 (2014)
3. H.P. Oak, S.J. Honade, ANFIS based short term load forecasting. *Int. J. Current Eng. Technol.* **5**(3), 1878–1880 (2015)
4. Prabhas, K., Vishavdeep, J.: Review: wind power forecasting & grid integration. *Int. J. Adv. Res. Electr. Electron. Instrum. Eng.* **5**(9), (2016)
5. R. Sharma, D. Singh, A review of wind power and wind speed forecasting. *J. Eng. Res. Appl.* **8**, 1–9 (2018)
6. S. Wagh, P.V. Walke, The hybrid solar and wind power extraction for domestic purposes: a review. *Int. J. Res. Advent Technol.* **5**(3) (2017)
7. L. Suganthi, S. Iniyan, A.A. Samuel, Applications of fuzzy logic in renewable energy systems— a review. *Renew. Sustain. Energy Rev.* **48**, 585–607 (2015)
8. S. Sukhdev, K. Naresh, Wind power forecasting: a survey. *Int. J. Eng. Res. Gener. Sci.* **4**(3) (2016)
9. A. Ahmed, M. Khalid, A review on the selected applications of forecasting models in renewable power systems. *Renew. Sustain. Energy Rev.* **100**, 9–21 (2019)
10. A. Ul Haque, J. Meng: Short-term wind speed forecasting based on fuzzy artmap. *Int. J. Green Energy* **8**, 65–80 (2013)
11. A. Kumar, M.Z.U. Khan, B. Pandey, S. Mekhilef, Wind energy: a review paper. *Gyancity J. Eng. Technol.* **4**(2), 29–37 (2018)

12. M.H. Baloch, J. Wang, G.S. Kaloi, Stability and nonlinear controller analysis of wind energy conversion system with random wind speed. *Int. J. Electr. Power Energy Syst.* **79**, 75–83 (2016)
13. W. Li, X. Jia, X. Li, Y. Wang, J. Lee, A Markov model for short term wind speed prediction by integrating the wind acceleration information. *Renew. Energy* **164**, 242–253 (2021)
14. P.K. Chaurasiya, V. Warudkar, S. Ahmed, Wind energy development and policy in India: a review. *Energ. Strat. Rev.* **24**, 342–357 (2019)
15. S.P. Mishra, P.K. Dash, Short term wind speed prediction using multiple kernel pseudo inverse neural network. *Int. J. Autom. Comput.* **15**(1), 66–83 (2018)
16. K. Gunavardhan, A convolution neural network based deep learning neural network forecast model for wind energy prediction. *Int. J. Adv. Sci. Technol.* **28**(19), 141–150 (2019)
17. A.P. Marugán, F.P.G. Márquez, J.M.P. Perez, D. Ruiz-Hernández, A survey of artificial neural network in wind energy systems. *Appl. Energy* **228**, 1822–1836 (2018)
18. A.A. Helen, A. Ojokoh Bolanle, O. Falaki Samuel, Comparative analysis of rainfall prediction models using neural network and fuzzy logic. *Int. J. Soft Comput. Eng.* **5**, 4–7 (2016)
19. V. Vanitha, Adaptive neuro-fuzzy inference system based short term wind speed forecasting. *Int. J. Innov. Technol. Explor. Eng.* **9**(5), 1–6 (2020)
20. L. Yang, M. He, J. Zhang, V. Vittal, Support-vector-machine-enhanced markov model for short-term wind power forecast. *IEEE Trans. Sustain. Energy* **6**(3), 791–799 (2015)
21. M. Godinho, R. Castro, Comparative performance of AI methods for wind power forecast in Portugal. *Wind Energy* **24**(1), 39–53 (2021)
22. M. Shahzad, U. Naeem, R. Sadiq, E. Muhammad, Fuzzy logic based algorithm for wind energy prediction, in *International Symposium on Recent Advances in Electrical Engineering (RAEE)*, vol. 4, pp. 1–6, IEEE (2019)
23. W. Zou, C. Li, P. Chen, An inter type-2 FCR algorithm based T-S fuzzy model for short-term wind power interval prediction. *IEEE Trans. Industr. Inf.* **15**(9), 4934–4943 (2019)
24. G. Grassi, P. Vecchio, Wind energy prediction using a two-hidden layer neural network. *Commun. Nonlinear Sci. Num. Simul.* **15**(9), 2262–2266 (2010)
25. S. Pasari, A. Shah, U. Sirpurkar, Wind energy prediction using artificial neural networks, in *Enhancing Future Skills and Entrepreneurship* (Springer, Cham, 2020), pp. 101–107
26. F. Dong, L. Shi, Regional differences study of renewable energy performance: a case of wind power in China. *J. Clean. Prod.* **233**, 490–500 (2019)
27. P.A. Adedeji, S. Akinlabi, N. Madushele, O.O. Olatunji, Wind turbine power output very short-term forecast: a comparative study of data clustering techniques in a PSO-ANFIS model. *J. Cleaner Prod.* **254**, 120135 (2020)
28. İ Mert, F. Üneş, C. Karakuş, D. Joksimovic, Estimation of wind energy power using different artificial intelligence techniques and empirical equations. *Energy Sour. Part A: Recovery Util. Environ. Effects* **43**(7), 815–828 (2021)
29. Y. El Khchine, M. Sriti, N.E.E.K. Elyamani, Evaluation of wind energy potential and trends in Morocco. *Heliyon* **5**(6), e01830 (2019)
30. F. Ji, X. Cai, J. Zhang, Wind power prediction interval estimation method using wavelet-transform neuro-fuzzy network. *J. Intel. Fuzzy Syst.* **29**(6), 2439–2445 (2015)
31. A. Mani, *Handbook of Solar Radiation and Wind Data for India* (Allied Publishers, New Delhi, 2019)
32. *Solar Radiation and wind data Handbook*, SEC & IMD Pune (2018)
33. O. Badran, E. Abdulhadi, Y. El-Tous, Fuzzy logic controller for predicting wind turbine power generation. *Int. J. Mechan. Mater. Eng.* **6**(1), 51–66 (2011)
34. P. Devyani, C. Krishna Teerth, A study on short term wind power prediction using machine learning approach. *Int. J. Adv. Res. Electric. Electron. Instrum. Eng.* **7**(5) (2018)
35. F. Farivar, T. Negar, M.A. Rosenet, Short-term wind speed forecasting using artificial neural networks for Tehran. *Int. J. Energy Environ. Eng.* **7**, 377–390 (2016)
36. F. Gökgöz, F. Filiz, Deep learning for renewable power forecasting: an approach using LSTM neural networks. *Int. J. Energy Power Eng.* **12**(6), 416–420 (2018)
37. M.H. Baloch, G.S. Kaloi, Z.A. Memon, Current scenario of the wind energy in Pakistan challenges and future perspectives: a case study. *Energy Rep.* **2**, 201–210 (2016)

38. C. Wan, Z. Xu, P. Pinson, Z.Y. Dong, K.P. Wong, Optimal prediction intervals of wind power generation. *IEEE Trans. Power Syst.* **29**(3), 1166–1174 (2013)
39. S. Qin, F. Liu, J. Wang, Y. Song, Interval forecasts of a novelty hybrid model for wind speeds. *Energy Rep.* **1**, 8–16 (2015)
40. Q. Chen, K.A. Folly, Wind power forecasting. *IFAC-Papers OnLine* **51**(28), 414–419 (2018)

Vision-Based Personal Face Emotional Recognition Approach Using Machine Learning and Tree-Based Classifier



R. Sathya, R. Manivannan, and K. Vaidehi

Abstract The facial emotion classification is a crucial task in human behavior analysis. By taking static images, the emotion is identified from the face expression. It is one of the categories in image processing that is utilized in a variety of disciplines, including human and computer interaction. Some resources are projected to perform automatic emotion recognition, which utilizes benchmark datasets. This research work is focused on real-time dataset that is used to identify six human facial emotions that are implemented by using SVM and tree-based classifier. Experimental outcome symbolizes the top most presentation on the SVM radial basis function (RBF) kernel recognition (95.49%) when associated to the tree-based classifier.

Keywords Human emotion · Face detection · Cascade classifier · SVM · Random forest · Decision tree · Naïve Bayes · Performance measure

1 Introduction

Human expression plays a vital role in establishing non-audio communication between human beings. Nowadays, the facial expression identification technique is gaining more and more attention from the people. Facial appearance includes key information about psychological, emotional, and even physical state of the chat. Facial appearance recognition will also create a practical impact. It has a very wide application forecast such as comprehensible interface among human and machine, humanistic mean of goods, and emotional robot. With facial appearance identification structures, the system can review the human emotions [1]. The intellectual processor will be capable to recognize, understand, and act in reaction to human intentions, expressions, and moods [2].

R. Sathya (✉)

Kongunadu College of Engineering and Technology (Autonomous), Trichy, India

R. Manivannan · K. Vaidehi

Department of Computer Science and Engineering, SCETW (Autonomous), Hyderabad, India

e-mail: drmanivannan@stanley.edu.in

The human facial expression detection systems are implemented in several living places like safety or examination; they can forecast the criminal and behavior by scanning the figures of their features to facilitate and confined by the control camcorder. In addition, facial appearance identification scheme has been utilized to create the response engine that is additionally cooperative with humans. The respond engine has turn out to be more intellectual by evaluating the customer's tone and commerce with the reactions according to their face sensation [3].

In addition, facial sensation is dominant in signed and any language identification approach that agreements with the challenge of hearing and mutilate humans. Person facial appearance identification approach has a significant contact on the diversion pasture and in addition its utilization boost the effectiveness of machines for particular medical-oriented robots and industrialized checking [4]. Usually, the automated system with facial appearance identification approach has been utilized to progress in our everyday lives.

Depression is a generally unattended health issue, unbounded by work stress, health issue, and commonly affecting majority of the student's performance in their study. To avoid it in future, the researchers construct a real-time face expression classification approach, so the teacher can supervise the student's mentality through classroom movement.

1.1 Categories of Facial Emotions

Universally six fundamental human expressions and unbiased feel are recognized globally. The real-life human facial emotions are divided into seven types of modules. The expressions are categorized as happy, fear, anger, surprise, sad, disgust, and neutral. Example pictures from online Japanese Female Facial Emotions (JAFPE) Database for six modules are given in Fig. 1. Henceforth, this paper concentrates on real-time Indian human facial expressions like fear, happy, anger, disgust, surprise, and neutral.

2 Related Work

This segment deals with the proposed work performed till now by different developers in the area of expression identification through facial language. Intentions of a number of writers are discussed in this section. Daisy and Kannan [5] projected a face appearance identification approach based on an original limited rotated local Gabor filter method. Gabor filter technique utilizes two-step feature firmness techniques such as principal component analysis (PCA) and linear discriminant analysis to choose and constrict the Gabor feature selection and smallest amount of space classification to identify the facial emotions. These techniques are valuable for a combined measurement decrease and excellent appreciation act in association with



Fig. 1 Facial emotions from JAFFE database

conventional entire Gabor filter method. The finest common identification achieves good recognition results for GTAVE, MIT, CMU, PIE, and real-time home databases.

Bashyal et al. [6] have projected a well-organized method for face appearance identification by using Gabor filters method as feature identification. JAFFE dataset is utilized for testing, and the obtained result is greater than 91%. Xie et al. [7] have projected spatial highest occurrence model based on numerical factors and employed elastic figure-based texture matching techniques for figure or surface-oriented human face appearance detection.

Zhang et al. [8] obtain the starting point as mixture corresponding to features of face as fixed, vibrant, position-based arithmetic, or area-based exterior characteristics. Novelist faced on fixed frames and experimental output for dimensional characters. These types of features have been identified by Gabor filters. The algorithm has presented excellent experimental outputs. In [9], the authors have published a novel technique that developed mutually geometric and texture data of facial positions. Gauss Laguerre method is employed for the identification of texture data for a variety of face emotions. The backpropagation neural network and probabilistic neural network techniques are utilized to detect the different type of X-ray images [10]. Fuzzy logic system-based aeration control approach for contaminated stream water [11]. Fuzzy chaos whale optimization and BAT integrated techniques for limitation evaluation are in sewage management [12]. Wireless rechargeable sensor network mistake techniques and immovability investigation [13].

This research exposes a variety of techniques tracked by researchers for facial emotion classification. Exhaustive investigation is carried out in additional subsections of this research work.

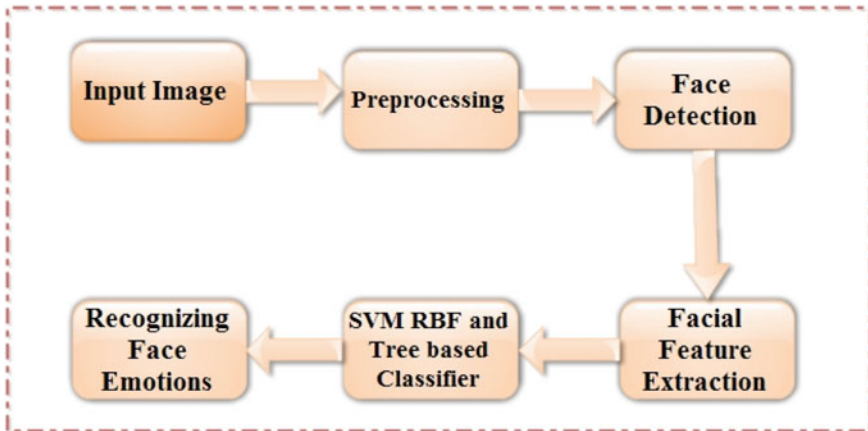


Fig. 2 System architecture for facial emotion recognition system

3 Proposed Facial Expression Recognition System

Facial language replicates the emotions of person, which disclose expensive information of one's sentiment, feedback, etc. Properly distinguishing these emotions is a difficult task. This segment describes the facial emotion identification system.

3.1 System Architecture

Overall system architecture of the projected facial expression identification is given in Fig. 2. Four modules collect the projected system: first phase is preprocessing, face detection as second phase, facial feature extraction as third phase, and facial emotional recognition as final phase by using two techniques named as SVM and tree-based classifier.

3.2 Preprocessing

Preprocessing method is the primary phase by inflowing the frame records into the face recognition and facial expression identification system. The essential data required for most facial appearance identification method is face location. In preprocessing component, frames are reformed from 640×640 pixel rates to 400×400 pixel rates.

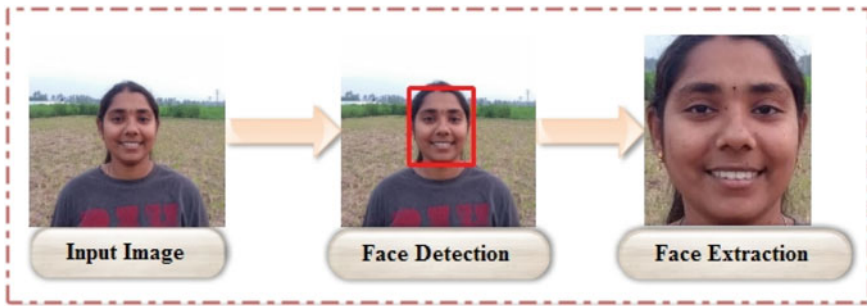


Fig. 3 Face detection and extraction using cascade classifier

3.3 Face Detection Using Cascade Classifier

In some applications such as human–computer boundary, face detection, video surveillance, and facial expressions, the very first step used here is localizing and detecting the human face. Viola–Jones object recognition process supported on cascade of recognizer is worn to trace the person face within every image of the sequence of video. More distinctively, we apply 14 feature prototypes [14], which comprise four corner features, eight line total features, and two corner-surround features. These samples are balanced separately in horizontal way in order to produce a wealthy and over-complete group of characteristics. These set of characteristics can be figured out in a regular and small time irrespectively of the location as given in [15]. Face detection and extraction using cascade classifier are given in Fig. 3.

3.4 Feature Extraction for Emotion Detection

Feature information is an informative area identified from a frames or a large sequence of video. Visual information exhibits various models of characteristics that could be used to recognize or represent the relevant information it opens. The gray length method (GRLM) is a way of extracting higher-order statistical texture features. The theory and techniques behind the method are presented in [16]. A position of successive grayscale level, collinear in a given path constitutes a grayscale level run. The run length is the amount of pixels in the run, and the run length charge is the amount of times such a run occurs in a face image. The grayscale level run length of matrix (GRLM) is a four-dimensional matrix in which every factor $f(x, y | \theta)$ provides the entire number of incidences of runs of length ‘y’ at grayscale level ‘x’, in a particular direction θ .

4 Facial Emotion Recognition Using SVM and Tree-based Classifier

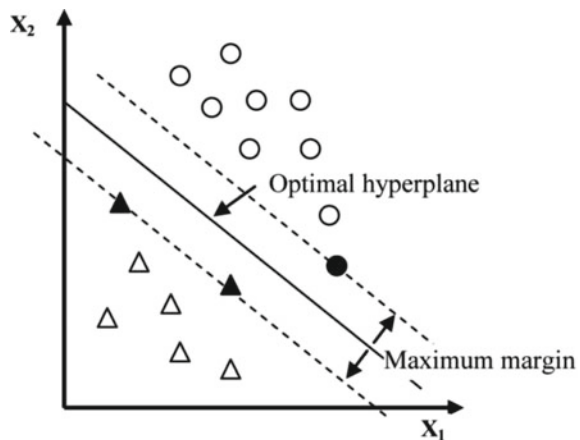
Machine learning (ML) contains many processes such as support vector machine (SVM), artificial neural networks (ANN), genetic algorithms (GA), Bayesian training, and probabilistic models [17], in that which we required only the accomplishment of the very first one machine learning (ML) methods on person fingerprint classification approach.

4.1 Support Vector Machine (SVM)

SVM methods is well-accepted practice for recognition in chart prototype classification [18]. The SVM is generally demoralized in kernel-based knowledge techniques. It achieves rational very important pattern classification presentation in optimization method [19]. Categorization tasks are commonly concerned with the use of testing and training data. The training data is subdivided into $(x_1, y_1), (x_2, y_2), \dots (x_m, y_m)$ into two classes, with $x_i \in R^n$ enclosing an n -dimensional feature vector and $y_i \in \{+1, -1\}$ enclosing class labels. SVM's goal is to create a copy that predicts the target value from the testing dataset. The hyperplane $w \cdot x + b = 0$, where $w \in R^n, b \in R$ is used to separate both classes in any space Z , is used in binary classification [20]. $M = 2/\|w\|$ gives the maximum scope.

The hyperplane is utilized to categorize the input feature space into the required target model. However, in order to fit the decision boundary in a hyperplane to make the most of distance boundary is preferred from feature data points for recognition. The sample maximum margin is given in Fig. 4.

Fig. 4 Representation of hyperplane



SVM can be structured by using a variety of kernel function to improve the accuracy: polynomial, Gaussian, and sigmoidal. SVM is well appropriate for mutually unstructured and structured feature data.

4.2 Tree-Based Classifier

4.2.1 Random Forest

A random forest [RF] is a group of decision trees skilled with random features. Random forest moves as follows. Given a set of training examples, a set of random trees H is shaped such that for the k th tree in the forest, a random vector ϕ_k is produced autonomously of the past random vectors $\phi_1 \dots \phi_{k-1}$. This vector k is then used to raise the tree resulting in a classifier $hk(x, \phi_k)$, where x is a feature vector. Random forest [21] is the fast and robust categorization performance that can handle multiclass problem.

4.2.2 Decision Tree (J48)

Decision trees are commonly used methods for pattern classification. Chi-squared automatic integration detection (CHAID) introduced in [22] and classifier 4.5 (C4.5, J48) in [17]. In this study, J48 algorithm decision tree was applied to traffic personnel hand features. J48 classifier is a standard model in C4.5 decision tree for supervised classification. In decision tree, feature collection procedure is done by information index. The information index for an exacting feature data Z at a node is calculated as

$$\text{InformationIndex}(X, Z) = \text{Entropy}_1(X) - \sum_{\text{Valueatz}} \frac{|X|}{|X_n|} \text{Entropy}_1(X)$$

where X is the combination of instance at that exacting node and

$$|X| : \text{Cordinality}_1$$

Entropy₁ of X is found as:

$$\text{Entropy}_1(X) = \sum_{n=1}^X -p_n \log_2 p_i$$

4.2.3 Naive Bayes (NB)

NB tree concept is a hybrid algorithm that characterizes a cross between Naive Bayes recognizer and C4.5 decision tree recognizer, and it is most excellent explained as a decision tree with nodes and branches [23]. The feature space which is classified in Naive Bayes is always independent to every other. If a is class variable and b is dependent feature space.

$$a = \text{avgmax}_a p_1(a) \prod_{n=1}^k p_1\left(\frac{b_n}{a}\right)$$

$P_1(a)$ is called class probability and is provisional probability.

$$p_1 \frac{b_n}{a}$$

Bayesian theorem probability states

$$\text{Posterior}_1 = \frac{\text{Prior}_1 * \text{Likelihood}}{\text{Evidance}_1}$$

5 Proposed Experimental Results

In this segment, the investigational outputs acquired in facial emotion identification systems are offered. The experimentations are performed by using SVM Torch and Weka Tool. LIBSVM [24] technique to expand the mold for every sensation and the molds are utilized to experiment the presentation, and recognitions are utilized for performance reason. Weka is a whole group of Java technique methods that are utilized to bring out several data mining and machine learning algorithms [10]. We evaluate the presentation of tree methods namely NBTree, REPTree, random forest, and random tree.

5.1 Database

The experimentations are accomplished with 25 subjects and the objects being clicked by the camera in natural scene at 25 fps with a 640×640 declaration. Real-time dataset includes 1500 objects of six person face emotion including neutral caused by 25 Indian female objects. Every objects has been priced on six emotions modules by 1500 Indian subjects.



Fig. 5 Real-time database for facial emotion

Figure 5 shows the real-time database images considered for facial expression recognition. The proposed training set is organized of 900 objects (the every set organizes 15 persons and every person includes 60 objects). And remaining procedure, the proposed test set includes 600 objects that are evaluated of random choosing 10 objects from each and every emotions.

5.2 Performance Evaluation

This research work provides a methodical investigation of multiclass system. Standard estimation calculation include precision $(P) = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$, recall $(R) = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$, specificity $(S) = \frac{\text{True Negative}}{\text{True Negative} + \text{False Positive}}$, and F-measure $= \frac{2 \text{ Precision Recall}}{\text{Precision} + \text{Recall}}$ that are utilized to estimate the presentation of the projected facial emotion classification system. These types of calculations provide the excellent perception on the recognition and presentation for facial emotion system.

5.3 Results Obtained Using SVM and Tree-Based Classifiers

This section presents human facial emotion classification system using support vector machine technique and using WEKA Tool [10]. For this reason, human facial emotion datasets shown in Fig. 4 were used. The presentation result of the decision trees (DT)

and random forests (RF) classifiers were calculated using tenfold cross-validation model and support vector machine using multiclass classification problem. The complete outputs for the average of recall, precision, and specificity examination for the experimental results are obtained in Fig. 5. Table 1 gives the confusion matrix for proposed facial emotion recognition using SVM (RBF) multiclass classification with proposed feature. Tables 2, 3, and 4 correspondingly tabulate the confusion matrix of the tree-based classifiers using the proposed facial features on six emotions by using tenfold cross-validation for emotion recognition.

The accuracy results obtained with proposed feature are by using SVM RBF kernel and tree-based classifier for each individual human facial emotion recognition, and they are presented in Table 5. Figure 6 shows the comparison of the precision, recall, and specificity value of proposed facial features by using support vector

Table 1 Confusion matrix for proposed facial emotion recognition using SVM (RBF)

	Happy	Fear	Anger	Surprise	Disgust	Neutral
Happy	93.64	0.57	1.18	1.26	1.45	1.9
Fear	0.03	96.41	0.96	0.46	1.5	0.64
Anger	1.5	0.07	96.5	0.21	0.75	0.97
Surprise	0.75	0.97	1.4	96.5	0.25	0.13
Disgust	0.65	0.97	1.36	1.09	94.48	1.45
Neutral	1.52	0.86	1.92	0.03	0.26	95.41

Table 2 Confusion matrix for proposed facial emotion recognition using random forest

	Happy	Fear	Anger	Surprise	Disgust	Neutral
Happy	95.76	0.51	1.97	0.62	1.02	0.12
Fear	1.5	93.82	1.18	1.26	1.49	0.75
Anger	1.36	0.65	95.65	1.21	0.8	0.33
Surprise	1.18	0.86	1.5	94.64	0.13	1.69
Disgust	0.03	1.09	0.96	0.26	96.41	1.25
Neutral	1.05	1.02	0.77	1.97	1.55	93.64

Table 3 Confusion matrix for proposed facial emotion recognition using decision tree (J48)

	Happy	Fear	Anger	Surprise	Disgust	Neutral
Happy	94.5	1.19	0.55	1.85	1.4	0.51
Fear	1.5	91.82	1.18	2.26	1.49	1.75
Anger	1.36	0.65	93.65	1.21	1.8	1.33
Surprise	1.18	0.86	1.5	94.64	0.13	1.69
Disgust	1.03	1.09	0.96	0.26	95.41	1.25
Neutral	1.5	0.75	1.18	1.26	1.49	93.82

Table 4 Confusion matrix for proposed facial emotion recognition using Naive Bayes

	Happy	Fear	Anger	Surprise	Disgust	Neutral
Happy	91.76	1.51	1.97	1.62	2.02	1.12
Fear	1.5	92.82	1.18	1.26	2.49	0.75
Anger	1.36	0.65	94.65	1.21	1.8	0.33
Surprise	1.18	0.86	1.5	93.64	1.13	1.69
Disgust	1.03	2.09	1.96	1.26	91.41	2.25
Neutral	1.05	2.02	0.77	1.97	1.55	92.64

Table 5 Comparison for the accuracy of all facial emotions

	Happy	Fear	Anger	Surprise	Disgust	Neutral
SVM	95.62	97.23	96.48	97.29	92.59	93.72
DT	95.62	97.25	94.24	95.74	93.24	93.78
RF	94.68	93.97	94.58	94.29	93.73	92.54
NB	92.52	92.31	92.58	92.52	94.29	92.73

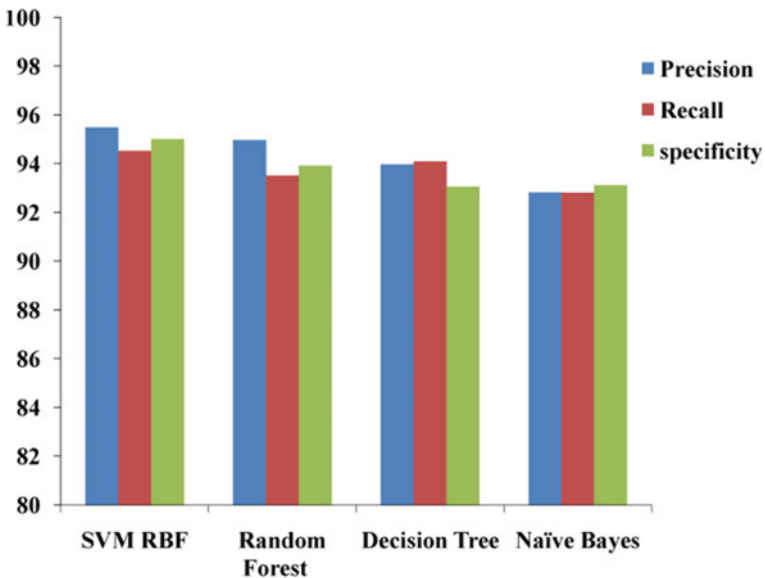


Fig. 6 Comparison of the precision, recall, and specificity value of SVM with RBF, random forests, decision trees, and Naïve Bayes classifier

machine [SVM] with radial basis function, random forests, decision trees (DT), and Naïve Bayes techniques. The projected approach provides superior performance value of proposed feature by using SVM RBF when compared to tree-based classifiers. Computer-aided detection and diagnosis of techniques is proposed in Balaji et al. [25].

6 Conclusion and Future Work

The proposed research has shown the accurate performance calculation by using SVM and tree-based classifier for human facial emotion classification approach. The research result gives significant attention to four steps, such as preprocessing, face identification, facial feature identification, and finally human facial emotion recognition. SVM RBF kernel gives higher accuracy compared to tree-based classifier. In forthcoming research, the suppleness of this proposed research is developed by using deep learning (DL) to classify facial emotions by using a variety of methods in a complicated environment.

References

1. B. Dixit, A. Gaikwad, Facial features based emotion recognition. IOSR J. Eng. (IOSRJEN) **8**(8), 2250–3021 (2018). ISSN (p): 2278-8719
2. L. Zhang, M. Kamlesh, C.N. Siew, P.L. Chee, Intelligent facial emotion recognition using moth-firefly optimization, Elsevier Publ. Int. J. Knowl. Based Syst. **111**, 248–267 (2016)
3. B.-F. Wu, C.-H. Lin, Adaptive feature mapping for customizing deep learning based facial expression recognition model. IEEE Access **6**, 12451–12461 (2018)
4. Y. Ding, Q. Zhao, B. Li, X. Yuan, Facial expression recognition from image sequence based on LBP and Taylor expansion. IEEE Access Spec. Sect. Sequential Data Model. Emerg. Appl. **5**, 19409–19419 (2017)
5. M.M. Daisy, P. Kannan, Investigation of rotated local Gabor features in face recognition using fusion techniques. J. Ambient Intell. Human Comput. **12**, 5895–5908 (2021)
6. Y. Zhao, D. Chen, *A Facial Expression Recognition Method Using Improved Capsule Network Model* (Hindawi Scientific Programming, 2020). Article ID 8845176
7. V.G.V. Mahesh, C. Chen, V. Rajangam, A.N. Joseph Raj, P.T. Krishnan, Shape and texture aware facial expression recognition using partial pyramid Zernike moments and Law's textures feature set. IEEE Trans. J. **4** (2021). Article in IEEE Access
8. L. Zhang, D. Tjondronegoro, Facial expression recognition using facial movement features. IEEE Trans. Affect. Comput. **2**(4), 219–228 (2011)
9. A. Poursaberi, H. Ahmadi, S.N. Yanushkevich, M. Gavrilova, Gauss–Laguerre wavelet textural feature fusion with geometrical information for facial expression identification. EURASIP J. Image Video Process. Springer Open (2012)
10. I.H. Witten, E. Frank, *Datamining: Practical Machine Learning Tools and Techniques with Java Implementations* (1999)
11. T. Vijayakumar, R. Vinothkanna, M. Duraipandian, Fuzzy logic based aeration control system for contaminated water. J. Electron. **2**(1), 10–17 (2020)

12. A. Sungeetha, R. Sharma, Fuzzy chaos whale optimization and BAT integrated algorithm for parameter estimation in sewage treatment. *J. Soft Comput. Paradigm (JSCP)* **3**(01), 10–18 (2021)
13. S.R. Mugunthan, Wireless rechargeable sensor network fault modeling and stability analysis. *J. Soft Comput. Paradigm (JSCP)* **3**(01), 47–54 (2021)
14. S.O. Adeshina, H. Ibrahim, S.S. Teoh, S.C. Hoo, Custom face classification model for classroom using Haar-like and LBP features with their performance comparisons. *Electronics* **10**(102) (2021). <https://doi.org/10.3390/electronics10020102>
15. J. Barreto, P. Menezes, J. Dias, Human-robot interaction based on Haar-like features and eigenfaces, in *International Conference on Robotics and Automation* (2004)
16. S.A. Ramola, A.K. Shakya, D.V. Pham, *Study of Statistical Methods for Texture Analysis and Their Modern Evolutions* (Wiley, 2020). <https://doi.org/10.1002/eng2.12149>
17. S. Theodoridis, A. Pikrakis, K. Koutroumbas, D. Cavouras, *Introduction to Pattern Recognition: A Matlab Approach* (Academic Press, 2010).
18. J. Liu, F.-P. An, *Image Classification Algorithm Based on Deep Learning-Kernel Function* (Hindawi Scientific Programming, 2020). Article ID 7607612. <https://doi.org/10.1155/2020/7607612>
19. V.N. Vapnik, V. Vapnik, *Statistical Learning Theory*, vol. 1 (Wiley, New York, 1998)
20. C.-C. Chang, C.-J. Lin, Libsvm: a library for support vector machines. *ACM Trans. Intel. Syst. Technol. (TIST)* **2**(3), 27 (2011)
21. R. Sathya, M. Kalaiselvi Geetha, Framework for traffic personnel gesture recognition, in *International Conference on Information and Communication Technologies (ICICT2014), Procedia Computer Science*, vol. 46, pp. 1700–1707 (2015)
22. T. Mitchell, *Machine Learning* (McGraw-Hill Computer Science Series, 1997)
23. J.R. Quinlan, *C4.5: Programs for Machine Learning* (Morgan Kaufmann, San Mateo, California, 1993)
24. C.-C. Chang, C.J. Lin, LIBSVM: a library for support vector machines. *ACM Trans. Intel. Syst. Technol.* **2**, 1–27 (2011)
25. G.N. Balaji, T.S. Subashini, P. Madhavi, C.H. Bhavani, A. Manikandarajan, Computer-aided detection and diagnosis of diaphyseal femur fracture, in *Smart Intelligent Computing and Applications* (Springer, Singapore, 2020), pp. 549–559

Design and Development of Smart Charger for Automotive Application



K. Vinutha, A. Usha, and Poonthugilan Jayaraman

Abstract Automotive system has the electronic control unit designed to withstand high-transient pulses and overvoltage conditions. The design and development of USB smart charger with protection circuit play an important role in charging application for mobile phones, tablets, and power banks. In this work, buck converter is designed for a wide input voltage of 9–16 V with 5 V, 6 A output and achieved CISPR 25 class 5 limit compliance for conducted emissions. This circuit is protected from overvoltage/current, undervoltage, and reverse polarity condition which frequently occur in automotive systems. This smart charger is designed to compliance with USB type-A and type-C port which has a separate USB controller for retrieving the charging profile from portable device. Design is carried out for dedicated battery charging of 1.2 standard and simulated using TINA software for normal and faulty conditions. The hardware prototype is developed using AEC-Qualified components and functional testing is performed.

Keywords Automotive system · USB smart charger · Electromagnetic compatibility PI filter · Buck converter · Controller · CISPR 25 · Battery charging 1.2

1 Introduction

In recent days, due to rapid development of technologies, the automotive manufacturers are working toward enhancing sophisticated designs and technologies that enable better vehicle–user connectivity by integrating all the control units to one

K. Vinutha (✉) · A. Usha
Department of Electrical and Electronics, B.M.S. College of Engineering, Bengaluru, India
e-mail: vinuthak.epe19@bmsce.ac.in

A. Usha
e-mail: usha.eee@bmsce.ac.in

P. Jayaraman
CMS Department, Molex India Business Services Pvt Ltd., Bengaluru, India
e-mail: Poonthugilan.Jayaraman@molex.com

infotainment system. Most smart chargers are not capable of withstanding electrical surges which occurs in automotive system and also non-AECQ standard. In this work, USB smart charger modules are designed in such a way to fit into limited space in a front panel of the vehicle and are capable of withstanding positive and negative surges. The USB smart charger may get affected by sudden surges which frequently occurs for a short duration time during load dump condition, so surge protection plays a prominent role [1]. The module which powered from a battery power requires protection from overvoltage, undervoltage, overcurrent, and reverse connection caused by miswiring during system maintenance or reinstallation. Therefore, most front-end power systems need protection from dynamic reverse polarity conditions that can occur during an inductive load disconnect from battery [2]. This can be avoided by employing protection circuit. In a system, electromagnetic interference can result in undesirable effects like noise in system, malfunction of controller which can lead to accidents [3]. Electronic systems that have no interference with adjacent systems or with themselves are known as EMC systems. Since EMC yields different EMI performances, it is important to choose a proper operating frequency and switching topology. The switching regulator is used to step down the input voltage to the desired output voltage [4–7]. The switch ringing noise is generated from buck converter which affects the performance, so noise filter is employed to avoid the interference [8]. The different control method is used in switching converter to reduce the noise and increase the efficiency [9]. In charging circuit, each port has a separate USB controller in order to retrieve the battery profile from connected portable device [10]. Each port may need to protect from surges which occurs on the configuration channel of type-C port and also from electrostatic discharge [11]. The block diagram for the designed USB smart charger is illustrated in Fig. 1.

In automotive system, power supply is accessed from battery by using mini-50 connector which must be protected from sudden surges arise in the circuit by using surge protection device. By employing the EMC PI filter, the protected power line

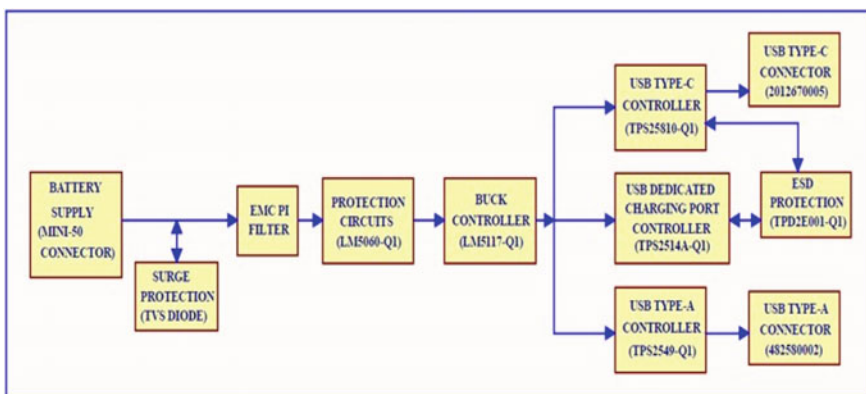


Fig. 1 Block diagram of developed USB Smart Charger

is filtered out from unwanted noise and emission which can be conductive or radiated emission generated by the surrounding electrical environment. This filtered-out signal is protected from over/undervoltage, overcurrent, and reverse battery connection before applied to the buck converter. Buck converter is used to step down the supply voltage that ranges from 9–16 V to 5 V, 6 A output. The current limit switch used to limit the current flow of 2.4 A to type-A connector and 3 A to type-C connector. Further, ESD protection is used to avoid any electrostatic discharge near to connector and controller.

There are separate type-A and type-C controllers to retrieve the battery charging profile based on D+ and D– data lines. There are three types of charging ports defined in battery charging such as dedicated charging ports (DCP), standard downstream ports (SDP), and charging downstream ports (CDP) [12]. The DCP is used only for charging USB port without data support and CDP is used for charging USB port with data support. The SDP is a standard USB port which can support up to maximum 900 mA based on the connection speed. Here, the USB smart charger is designed to operate in dedicated charging port; the USB controller will perform the charging with integrated short to VBUS protection. When a portable device is connected for charging purposes, then the host and portable device will start enumeration and host identifies the battery charging profile of connected device. Here, both the ports are used for charging purpose which is used at in-vehicle USB smart charging application for mobile phones, power bank, tablets etc.

2 USB Smart Charger

The design specification of buck converter for USB smart charger is illustrated in Table 1.

Table 1 Buck converter specifications

Input voltage range	9–16 V
Nominal input voltage	12 V
Output voltage	5 V
Output current	6 A
Switching frequency	400 kHz
Maximum Output power	30 W
Operating temperature range	–40 to 125 °C
Maximum input voltage ripple	100 mV
Maximum output voltage ripple	50 mV

2.1 Surge Protection

The surge protection is mainly used to protect the circuit from unwanted sudden transient voltage pulses or spikes in the vehicle wiring from damaging systems like control modules and infotainment equipment. Mainly, these pulses are generated by electrostatic discharge from load dump for a short duration of time. To ensure optimal performance, the protection from transients induced by positive and negative transient pulses and load dump from harsh automotive environment is provided by using bidirectional TVS diode.

2.2 EMC PI Filter

EMI will emit noise in system which may cause malfunction of controller which even can lead to hazardous accidents. The EMC PI filter is used to remove the unwanted high-frequency noise and emission in the system as illustrated in Fig. 2.

EMC PI filter is designed using CISPR 25 which is the automotive EMI standard for conducted and radiated emissions that most OEMs reference for requirements are considered. The required attenuation for filter is calculated using Eq. (1).

$$|Att|_{dB} = 20 * \log \left(\frac{\frac{\sin(\pi * D)}{(\pi^2 * f_{SW} * C_{IN})}}{1 \mu V} \right) - V_{max} \tag{1}$$

Where D is the duty ratio of buck converter, f_{SW} is switching frequency, C_{IN} is input capacitor, V_{max} is the maximum noise level as per CISPR 25. The selected inductor value is $4.7 \mu H$. The capacitor (C_f) is calculated by using Eqs. (2) and (3).

$$C_{fa} = \left(\frac{C_{IN}}{\left(C_{IN} * L_f * \left(\left(\frac{2 * \pi * f_{SW}}{10} \right)^2 \right) \right) - 1} \right) \tag{2}$$

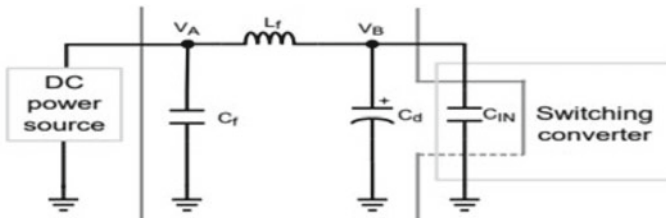


Fig. 2 EMC PI filter

$$C_{fb} = \frac{1}{L_f} * \left(\frac{10^{\frac{|Arr|_{dB}}{40}}}{(2 * \pi * f_{SW})} \right)^2 \quad (3)$$

The obtained C_{fa} and C_{fb} values are 3.669 and 0.4115 μF . So, the selected capacitor value is 4.7 μF which is greater than 3.669 μF . The damping capacitor C_d is calculated using Eq. (4) and obtained 164.4 μF .

$$C_d \geq 4 * C_{IN} \quad (4)$$

The equivalent series resistance (ESR) value of damping capacitor is calculated using Eq. (5) and obtained 0.338 Ω .

$$\text{ESR} \geq \sqrt{\frac{L_f}{C_{IN}}} \quad (5)$$

2.3 Protection Circuit

The electronic modules which powered from a battery power requires protection from overvoltage, undervoltage, overcurrent, and reverse connection caused by miswiring during system maintenance or reinstallation. It will lead to damage or malfunction of the system. So here, LM5060 is used as a protection controller which has low quiescent current, adjustable undervoltage lockout, and overvoltage protection. When fault occurs on the power line, then LM5060 will control the transitions of power NMOS switch and eventually turn OFF the supply.

2.4 Synchronous Buck Converter

In DC–DC converters, buck converters step down the voltage from the input to the output of the converter. The output voltage of a filter is equal to the average input voltage. If the inductor current remains positive during switch closure, the filter V_X will have V_S as input, while it will have zero upon switch opening as illustrated in Fig. 3.

As a synchronous buck controller, the LM5117-Q1 is used for step-down application from an input supply that ranges from 9 to 16 V to 5 V output. For automotive applications, the LM5117-Q1 is certified according to AEC-Q100 and has an ambient operating temperature of -40 to 125 $^{\circ}\text{C}$. An emulated current ramp is used as a control method for current mode control. It provides cycle-by-cycle current limiting, automatic line feed forwarding and ease the loop compensation. The noise sensitive of PWM circuit is reduced by using an emulated control ramp which allows

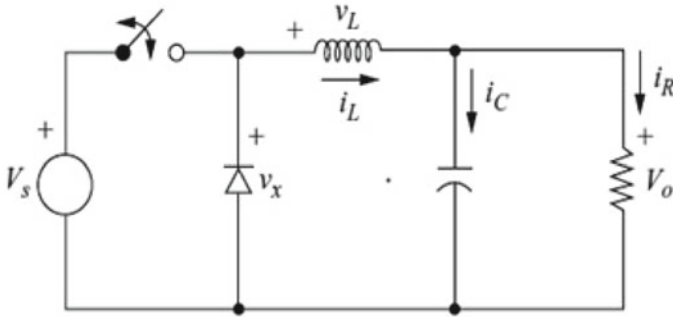


Fig. 3 Buck converter circuit

control of small duty ratio in high-input voltage application. It uses adaptive dead time control to drive both high- and low-side external NMOS power switches. The design steps for synchronous buck converter as follows: To set the minimum input operating voltage of the regulator, an external undervoltage lockout voltage divider is used. The voltage divider resistors R_{UV1} and R_{UV2} are calculated using Eqs. (6) and (7).

$$R_{UV2} = \frac{V_{HYS}}{20 \mu A} [\Omega] \tag{6}$$

$$R_{UV1} = \frac{1.25V * R_{UV2}}{V_{IN(STARTUP)} - 1.25V} [\Omega] \tag{7}$$

The obtained R_{UV1} and R_{UV2} values are 16.129 and 100 k Ω , respectively. A single external resistor (R_T) is used to program the switching frequency of 400 kHz to the buck controller. The R_T value can be calculated from Eq. (8) and obtained as 12.052 k Ω .

$$R_T = \frac{5.2 * 10^9}{f_{SW}} - 948 [\Omega] \tag{8}$$

where f_{SW} is the switching frequency. The output inductor L_O value is calculated by considering 30% of full load current as a ripple current using Eq. (9) and obtained as 4.774 μ H.

$$L_O = \frac{V_{OUT}}{I_{PP(MAX)} * f_{SW}} * \left(1 - \frac{V_{OUT}}{V_{IN(MAX)}} \right) \mu H \tag{9}$$

where $I_{PP(MAX)}$ is maximum ripple current, $V_{IN(MAX)}$ is maximum input voltage. The performance of the converter will vary depending on the value of K . For this design, $K = 1$ was chosen to control sub-harmonic oscillation and to achieve one-cycle damping. The current sense resistor value can be calculated using Eq. (10) and

obtained the value 13.33 m Ω .

$$R_{\text{RAMP}} = \frac{L_o}{K * C_{\text{RAMP}} * R_S * A_S} [\Omega] \quad (10)$$

The time for the output to reach final regulated value is known as soft-start time (t_{SS}). It is determined by the capacitor connected at the SS pin. For given t_{SS} , the capacitor C_{SS} is calculated as 25 nF using Eq. (11).

$$C_{\text{SS}} = \frac{t_{\text{SS}} * 10 \mu\text{A}}{0.8 \text{ V}} [\text{F}] \quad (11)$$

In order to filter out the input ripple occurring at switching operation, input capacitor is used. The minimum input capacitor value is calculated by using Eq. (12) and obtained 37.5 μF .

$$C_{\text{IN}} = \frac{I_{\text{OUT}}}{4 * f_{\text{SW}} * \Delta V_{\text{IN}}} [\text{V}] \quad (12)$$

where ΔV_{IN} is input ripple voltage, I_{OUT} is required output current. The output capacitor is used to supply charge in transient loading stage and filter the ripple caused by inductor current. The minimum output capacitor can be calculated as 16.7560 μF using Eq. (13).

$$\Delta V_{\text{OUT}} = I_{\text{PP}} * \sqrt{R_{\text{ESR}}^2 + \left(\frac{1}{8 * f_{\text{SW}} * C_{\text{OUT}}} \right)^2} [\text{V}] \quad (13)$$

where R_{ESR} is equivalent series resistance. To regulate the output voltage, a resistor divider circuit is used. The ratio of resistor R_{FB1} and R_{FB2} is calculated by using Eq. (14).

$$\frac{R_{\text{FB2}}}{R_{\text{FB1}}} = \frac{V_{\text{OUT}}}{0.8 \text{ V}} - 1 \quad (14)$$

By considering R_{FB1} as 1 k Ω , we obtained R_{FB2} value as 5.25 k Ω .

2.5 USB Type-C Controller

In this work, TPS25810-Q1 USB type-C downstream-facing port controller with power switch of 3 A rated current is used. It has two selectable fixed current limits that align with the type-C current level by 34 m Ω RDS (ON) power switch. For detection of whether a USB device is connected, the controller will monitor the configuration channel of type-C connector. This controller does not consist of data

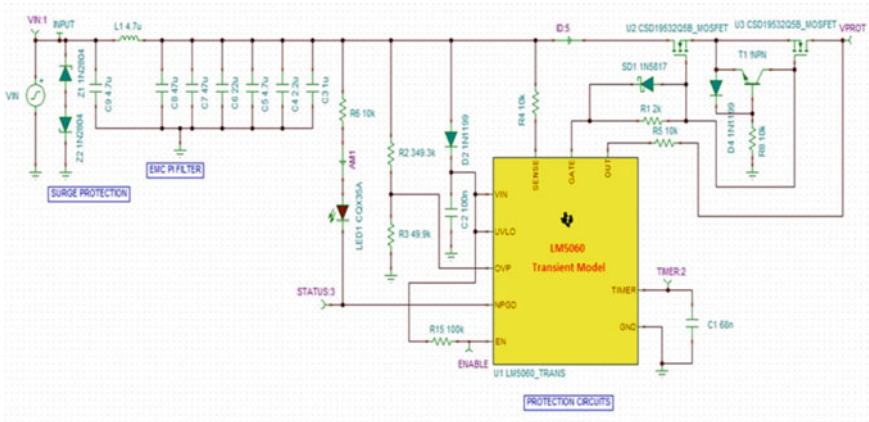


Fig. 4 Simulation for protection circuits along with EMC PI filter and surge protection

lines D+ and D–, so however the separate TPS2514A-Q1 is used as a USB dedicated charging port (DCP) controller used for charging application. By using an auto detect feature, USB data lines are automatically monitored and retrieved the charging profile of portable device. When the switch is in over current or over temperature condition, then the FAULT output signal is generated.

2.6 USB Type-A Controller

The TPS2549-Q1 device is an AEC-Q100 qualified USB charging port controller and power switch which integrates cable compensation, electrostatic discharge, and D+ and D– short to VBUS protection. It is suitable for automotive USB charging and USB port protection applications. It has a current limited power distribution switch using N-channel MOSFET which has a 47 mΩ RDS(ON) resistor to encounter the short circuit, heavy capacitive loads. The device allows the user to set the current limit threshold through external resistor. Depending on the electrical characteristics of the data line, the mode is detected. The device may have three different charging scheme such as shorted, divider 3 and 1.2 V mode but emulates only one state at a time. The shorted DCP mode supports BC 1.2 and YD/T 1591-2009 Chinese telecommunication standard.

3 Simulation Study and Results

The simulation for protection circuits having EMC PI filter and surge protection using LM5060-Q1 are illustrated in Fig. 4. The simulation is effectively carried out using TINA software. The transient analysis of each block is carried out for 100 ms.

When the battery power supply is applied to the TVS diode, then it acts as an open circuit. If the positive and negative transient occurs on power supply line, then it acts as a short circuit and input voltage is clamped to the respective rating of the selected TVS diode. The surge-protected supply is connected to the EMC PI filter to filter out the noise or emission on supply line as per the CISPR 25 automotive transient standard. The filtered power supply eventually gets protected for over voltage, under-voltage, over current, and reverse polarity circuit using LM5060 IC. The simulation results for protection circuits with EMC PI filter and surge protection under normal conditions are illustrated in Fig. 5.

Figure 6 illustrates the various terminal outputs when the reverse polarity potential of -12 V is applied. Here, in this waveform, it is observed that output voltage is -7.70 mV which is approximately equal to zero.

Figure 7 illustrates the various terminal output waveforms when overvoltage of 18 V is applied. When the overvoltage event occurs, then the status terminal voltage is high, and the capacitor starts discharging through the timer capacitor and in turn the output voltage is 0 V .

The simulation circuit for the synchronous buck converter using LM5117-Q1 IC is illustrated in Fig. 8. The nominal input voltage 12 V is stepped down to 5 V using LM5117-Q1 buck converter. In order to set the switching frequency of 400 kHz , the RT resistor is set to $12.1\text{ k}\Omega$. When the MOSFET switch T1 is switched on, then the current will flow through the inductor, output capacitor, and resistive load. The starting inrush current is limited by employing a soft-start technique, when the switch

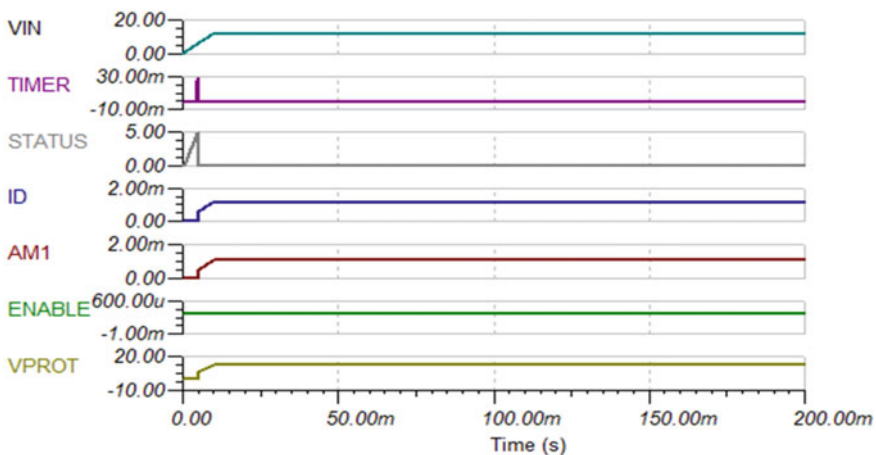


Fig. 5 Various terminal output waveforms under normal condition

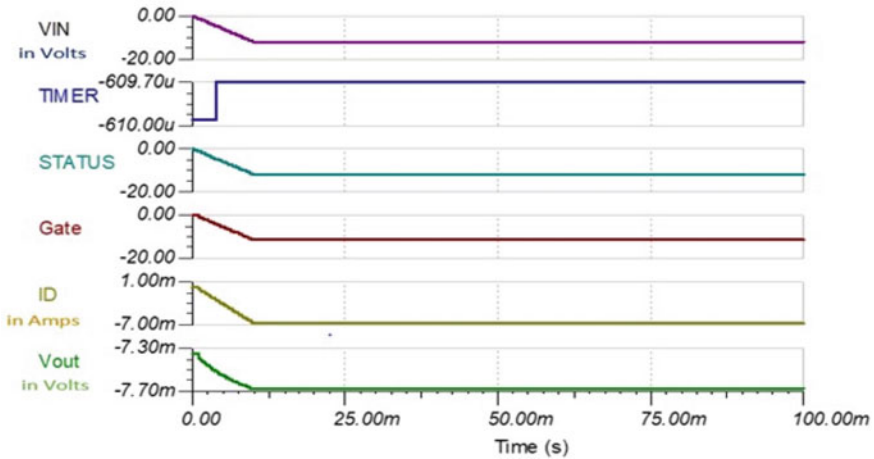


Fig. 6 Various terminal output waveforms for Reverse polarity of -12 V

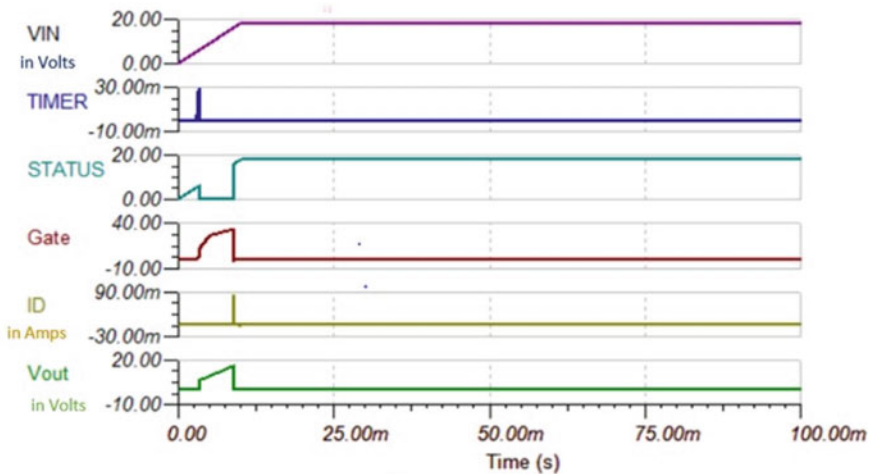


Fig. 7 Various terminal output waveforms for overvoltage of 18 V

is turned on. Here, the soft-start time of 2 ms is selected to reach the rated/steady-state current. The inductor current cannot change the path instantaneously, so the lower-side switch will provide path to flow the current through it.

This method is called synchronous rectification which reduces the voltage drop across the low-side switch and also losses. Hereby adjusting the feedback resistor and type 2 compensator value, the output voltage is regulated to constant 5 V. Figure 9 illustrates the input voltage waveform. It is observed from the waveform that the nominal input voltage of 12 V is applied to the buck converter circuit.

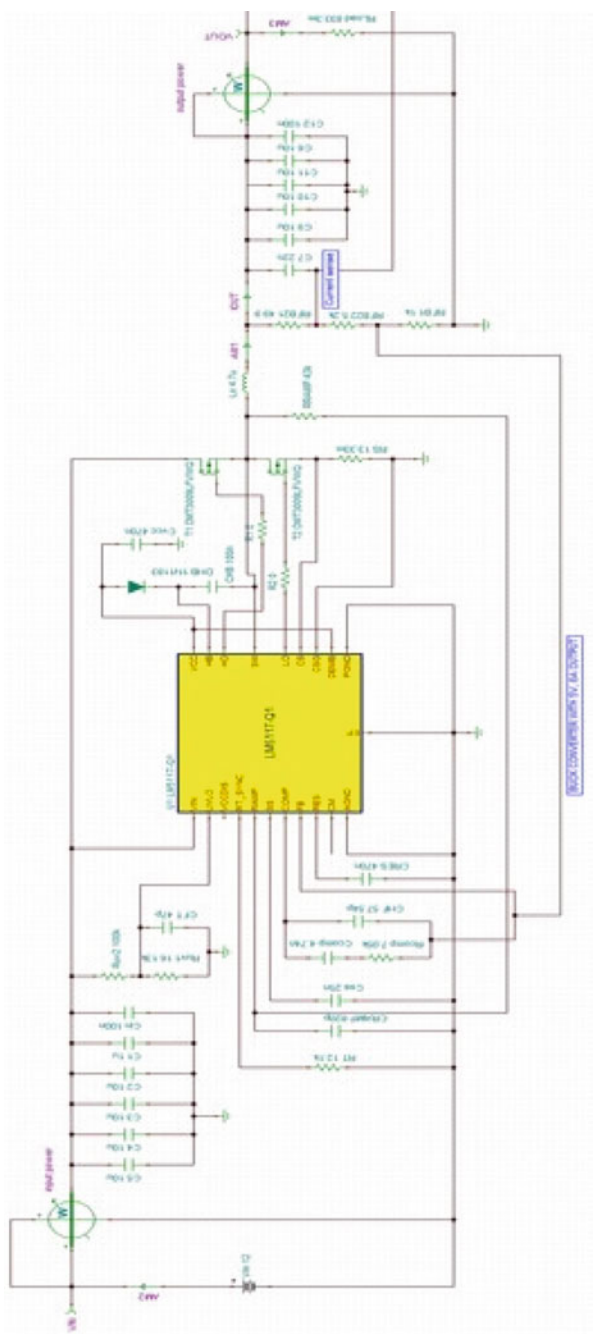


Fig. 8 Simulation circuit for synchronous buck converter using LM5117-Q1

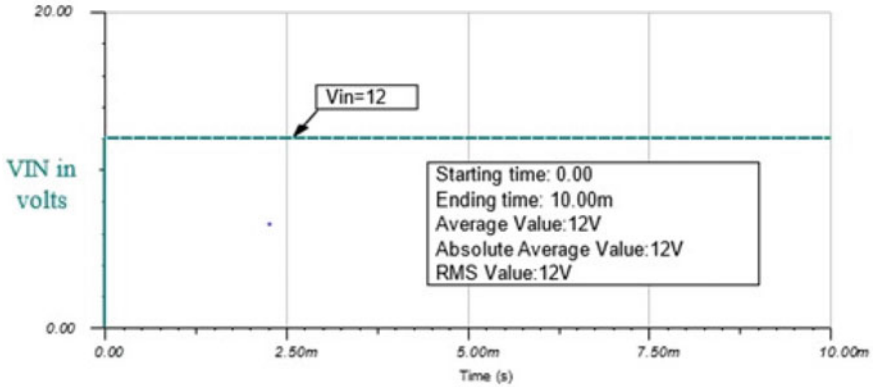


Fig. 9 Input voltage waveform

Figure 10 illustrates the input current waveform which is flowing through the battery supply. Here, it is observed that the average input current is 2.58 Amps.

Figure 11 illustrates the inductor current waveform. It is observed from the waveform that the minimum and maximum inductor current is 5.24 and 6.76 A, respectively. The average current flowing through the inductor is 6 A.

Figure 12 illustrates the output voltage waveform with 10 mV ripple. It is observed that the output voltage ripple of 10 mV which is within the specified allowable ripple limit.

Figure 13 illustrates the output current waveform. It is observed that the average output current 6 Amps is obtained from synchronous buck converter.

Figure 14 illustrates the output power waveform. The output power of 29.98 W is obtained which is the designed value based on the specifications of the buck converter.

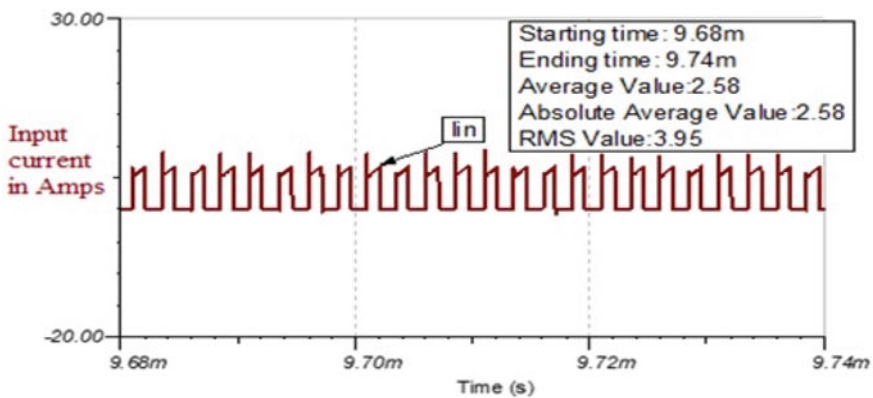


Fig. 10 Input current waveform

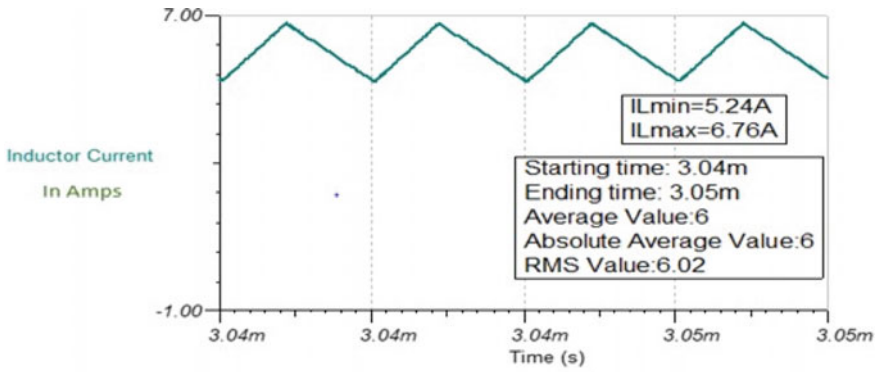


Fig. 11 Inductor current waveform

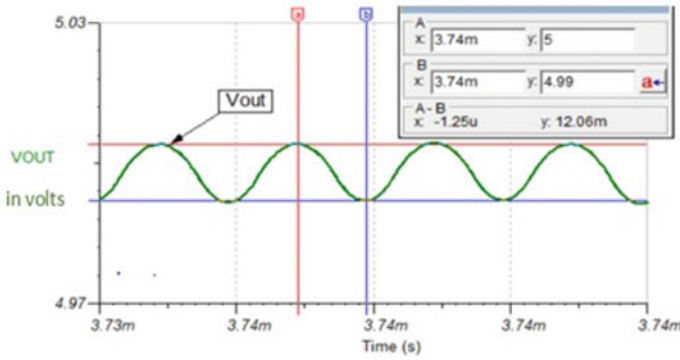


Fig. 12 Output voltage waveform with 10 mV ripple

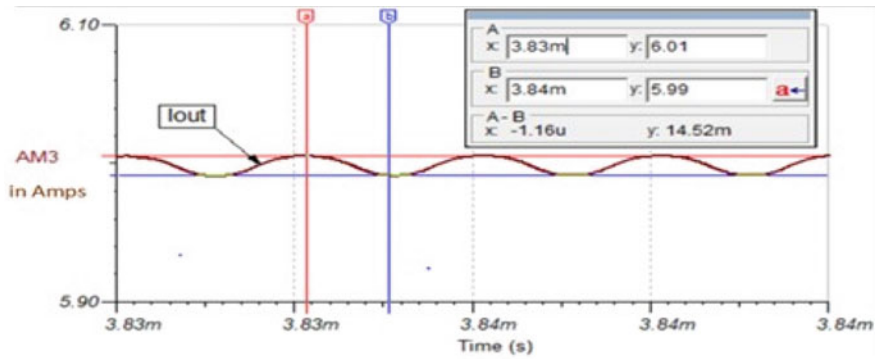


Fig. 13 Output current waveform

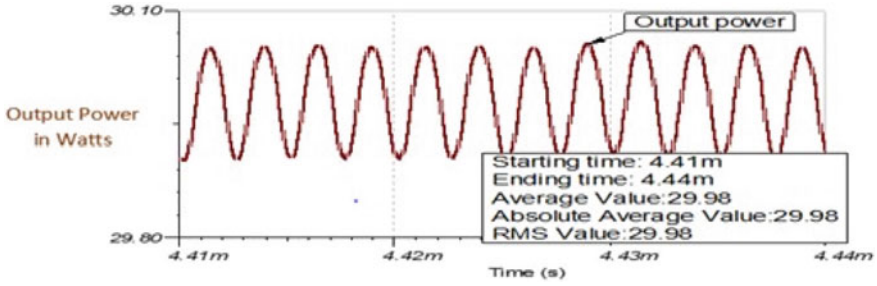


Fig. 14 Output power waveform

The output power of 29.98 W is obtained by using LM5117-Q1 for an input power of 31.15 W and obtained the efficiency of 96.24%. The simulation circuit of USB type-A controller using TPS2549-Q1 is as illustrated in Fig. 15. This in turn will monitor the CTL inputs and transitions which is required for DCP charging mode.

The simulation circuit for USB type-C controller using TPS25810-Q1 IC is as illustrated in Fig. 16.

Figure 17 illustrates the input voltage and output voltage waveforms for USB controller which is observed for the constant 5 V across VBUS1 pin.

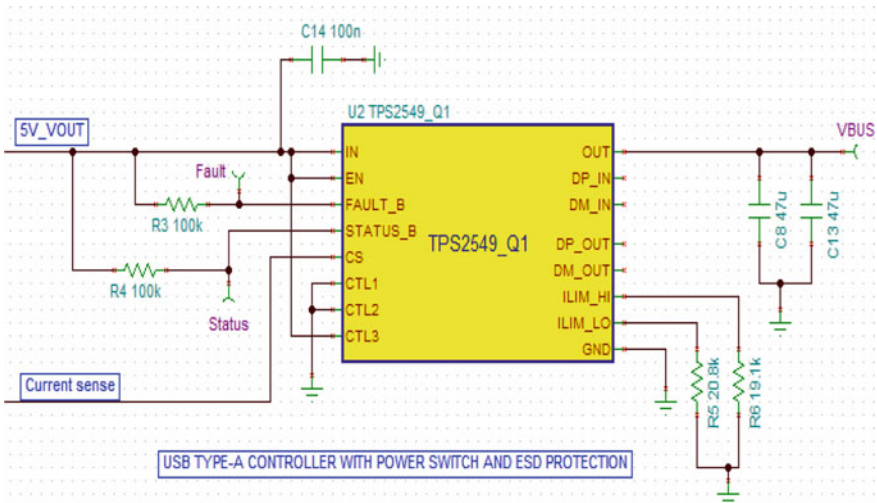


Fig. 15 Simulation circuit for USB type-A controller

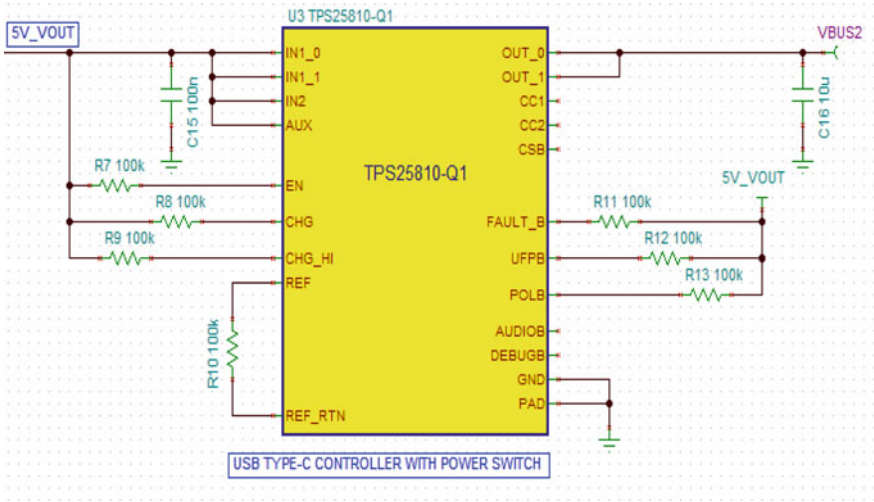


Fig. 16 Simulation circuit for USB type-C controller

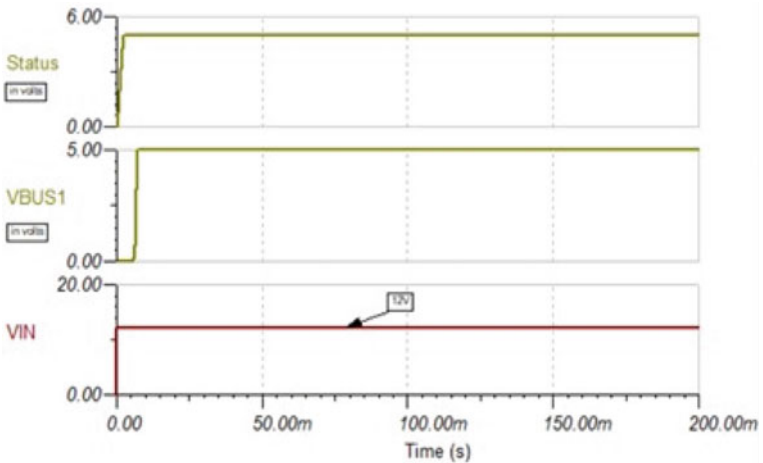


Fig. 17 Input and output voltages for USB controller

4 Hardware Setup and Results

A 30 W smart charger with USB type-A and type-C port has been implemented, and the hardware test setup is illustrated in Fig. 18. The AEC-Qualified components are used to develop the smart charger. The hardware test setup includes smart charger prototype, DC power supply, electronic load, and digital signal oscilloscope.

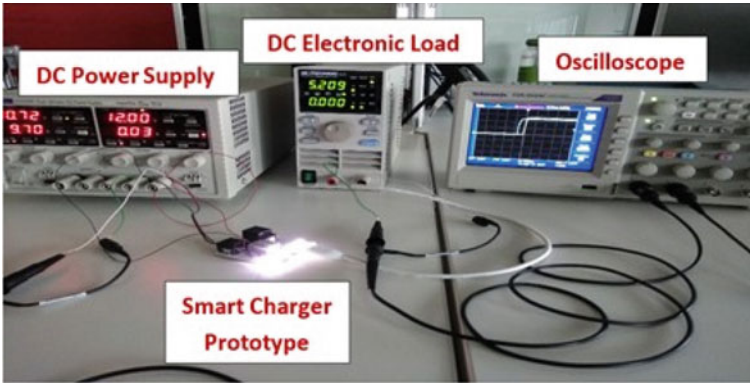


Fig. 18 Hardware setup for USB smart charger

The turn ON waveform of input and output voltage for the smart charger is illustrated in Fig. 19.

The turn OFF waveform of input and output voltage for the smart charger is illustrated in Fig. 20.

Table 2 illustrates the input current, output voltage, and efficiency of USB type-A port by fixing the input voltage at nominal 12 V and varying the output current using DC (programmable) electronic load.

Table 3 illustrates the input current, output voltage, and efficiency of USB type-A port by keeping the rated 2.4 A output current constant using DC electronic load and varying the input voltage from 9 to 16 V.

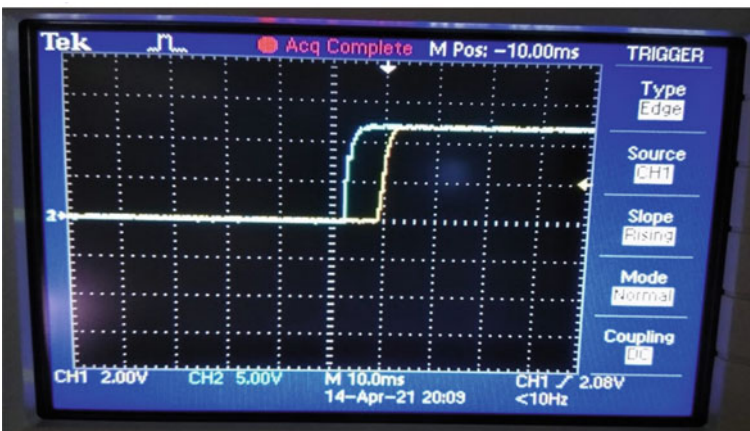


Fig. 19 Turn ON waveform



Fig. 20 Turn OFF waveform

Table 2 USB type-A port efficiency at 12 V Input voltage

Input voltage (V_{IN})	Input current (I_{IN})	Output voltage (V_{OUT-A})	Output current (I_{OUT-A})	Efficiency $\eta = \frac{(V_{OUT-A} * I_{OUT-A})}{(V_{IN} * I_{IN})} \%$
12	0.42	5.082	0.9	90.75
12	0.475	5.074	1.01	89.90
12	0.564	5.053	1.2	89.59
12	0.658	5.038	1.4	89.32
12	0.75	5.012	1.6	89.10
12	0.8431	4.999	1.8	88.93
12	0.935	4.985	2	88.85
12	1.026	4.962	2.2	88.66
12	1.12	4.949	2.4	88.37

The efficiency versus load current curve for USB type-A port is illustrated in Fig. 21. It is observed from the graph that maximum efficiency is obtained at low output current.

Table 4 illustrates the input current, output voltage, and efficiency of USB type-C port by fixing the input voltage at 12.8 V and also by varying the output current using DC electronic load.

Table 5 illustrates the input current, output voltage, and efficiency for USB type-C port by maintaining the rated 3 A output current using DC electronic load and varying the input voltage from 9 to 16 V.

Table 3 USB type-A port efficiency at 2.4 A output current

Input voltage (V_{IN})	Input current (I_{IN})	Output voltage (V_{OUT-A})	Output current (I_{OUT-A})	Efficiency $\eta = \frac{(V_{OUT-A} * I_{OUT-A})}{(V_{IN} * I_{IN})} \%$
9	1.51	4.959	2.399	87.53
10	1.35	4.957	2.399	88.08
11	1.23	4.953	2.399	87.82
12	1.12	4.949	2.399	88.33
13	1.04	4.945	2.399	87.74
14	0.96	4.942	2.399	88.21
15	0.9	4.94	2.399	87.78
16	0.84	4.938	2.399	88.14

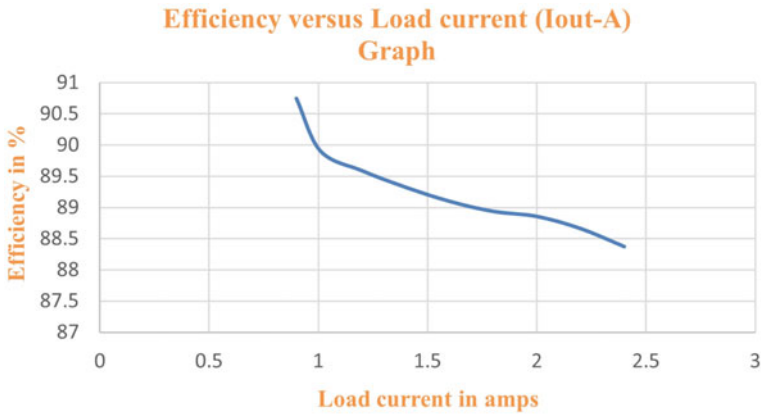


Fig. 21 Efficiency versus load current curve for USB type-A port

Table 4 USB type-C port efficiency at 12.8 V input voltage

Input voltage (V_{IN})	Input current (I_{IN})	Output voltage (V_{OUT-C})	Output current (I_{OUT-C})	Efficiency $\eta = \frac{(V_{OUT-C} * I_{OUT-C})}{(V_{IN} * I_{IN})} \%$
12.8	1.3025	5.018	3.001	90.32
12.8	1.072	5.032	2.501	91.71
12.8	0.8465	5.045	2	93.12
12.8	0.6275	5.057	1.5	94.44
12.8	0.416	5.068	1.001	95.27
12.8	0.209	5.078	0.5	94.90

Table 5 USB type-C port efficiency at 3 A output current

Input voltage (V_{IN})	Input current (I_{IN})	Output voltage (V_{OUT-C})	Output current (I_{OUT-C})	Efficiency $\eta = \frac{(V_{OUT-C} * I_{OUT-C})}{(V_{IN} * I_{IN})} \%$
9	1.8825	5.018	3.001	88.88
12.8	1.3025	5.018	3.001	90.32
13.5	1.2325	5.02	3.001	90.54
16	1.0335	5.022	3.001	91.14

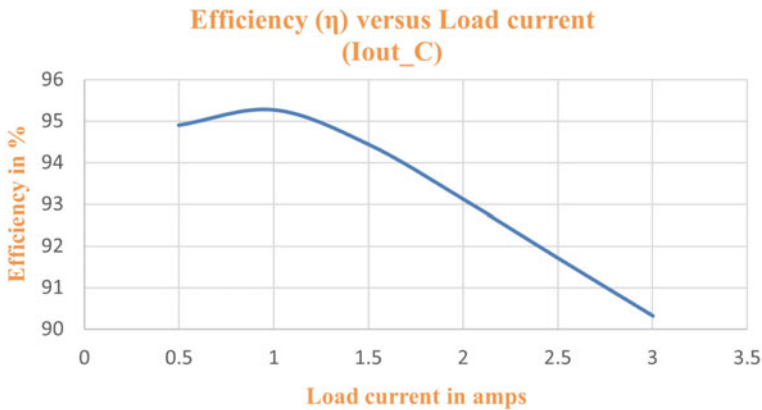


Fig. 22 Efficiency versus load current curve for USB type-C port

The efficiency versus load current curve for USB type-C port is illustrated in Fig. 22. It is observed from the graph that maximum efficiency is obtained at low output current.

5 Conclusion

The 30 W USB smart charger is effectively designed, simulated, and tested to fit into the limited space of front panel of vehicle and also expected to withstand a harsh automotive environment. Hardware setup for USB smart charger for type-A and type-C port are successfully implemented and the results are observed. Further, the hardware module will withstand all positive and negative surges which occur on the supply line. The input protection circuits for under/over voltage, reverse polarity, and overcurrent circuits are designed using LM5060-Q1 IC. The USB type-C and type-A controllers are used to monitor and control the required charging mode as per BC1.2 specifications. In this work, the designed smart charger is simulated using

TINA software and obtained the buck converter efficiency of 96.24%. Hardware prototype is developed, and functional testing is carried out on both the USB type-A and type-C port and observed the required waveforms and results. The efficiency of 88 and 90% at rated current is achieved for type-A and type-C port, respectively. The smart charger can be made to charge the laptop and other portable devices up to 100 W by adopting power delivery feature.

References

1. A. Kotlar, P. Svasta, Protection supply circuit design for power electronics in automotive, in *2017 40th International Spring Seminar on Electronics Technology (ISSE)*, Sofia, 2017, pp. 1–4. <https://doi.org/10.1109/ISSE.2017.8000975>
2. L. Fu, L. Peizhi, C. Weiyan, Design of high voltage surge suppression circuit for unmanned ground vehicle computer system, in *2017 IEEE International Conference on Unmanned Systems (ICUS)*, Beijing, 2017, pp. 539–543. <https://doi.org/10.1109/ICUS.2017.8278404>
3. P. Lopez, T.D. Pham, T.Q.V. Hoang, F. Lafon, A methodology to design an EMC filter layout providing optimal response based on simulation & considering the inter component couplings, in *2016 International Symposium on ElectroMagnetic Compatibility (CEM)*
4. B. Dragoi, On selecting a front-end DC-DC converter for automotive applications, in *2016 12th IEEE International Symposium on Electronics and Telecommunications (ISETC)*, Timisoara, 2016, pp. 29–32. <https://doi.org/10.1109/ISETC.2016.7781049>
5. Y. Liu, A. Kumar, S. Pervaiz, D. Maksimovic, K.K. Afridi, A high-power-density low-profile DC-DC converter for cellphone battery charging applications, in *2017 IEEE 18th Workshop on Control and Modeling for Power Electronics (COMPEL)*, Stanford, CA, 2017, pp. 1–6. <https://doi.org/10.1109/COMPEL.2017.8013362>
6. W. Yang, 95% light-load efficiency single-inductor dual-output DC-DC buck converter with synthesized waveform control technique for USB type-C, in *2016 IEEE Symposium on VLSI Circuits (VLSI-Circuits)*, Honolulu, HI, 2016, pp. 1–2. <https://doi.org/10.1109/VLSIC.2016.7573476>
7. C. Nan, R. Ayyanar, Y. Xi, A GaN based 2.2 MHz active-clamp buck converter for automotive applications, in *2016 IEEE Energy Conversion Congress and Exposition (ECCE)*, Milwaukee, WI, 2016, pp. 1–8. <https://doi.org/10.1109/ECCE.2016.7854683>
8. K. Kim, H. Shim, A.C. Scogna, C. Hwang, SMPS ringing noise modeling and managing methodology for RFI solutions in mobile platforms. *IEEE Trans. Compon. Packag. Manuf. Technol.* **8**(4), 554–561 (2018). <https://doi.org/10.1109/TCPMT.2018.2799330>
9. R. Goel, G. Seo, H. Le, A smart-USB-cable buck converter with indirect control, in *2017 IEEE 18th Workshop on Control and Modeling for Power Electronics (COMPEL)*, Stanford, CA, 2017, pp. 1–6. <https://doi.org/10.1109/COMPEL.2017.8013370>
10. F. He, USB port and power delivery: an overview of USB port interoperability, in *2015 IEEE Symposium on Product Compliance Engineering (ISPC)*, Chicago, IL, 2015, pp. 1–5. <https://doi.org/10.1109/ISPC.2015.7138710>
11. C. Ke, M. Ker, On-chip over-voltage protection design against surge events on the CC Pin of USB type-C interface. *IEEE Trans. Electr. Dev.* **67**(7), 2702–2709 (2020). <https://doi.org/10.1109/TED.2020.2992383>
12. M. Lai, *Battery Charging Specification Revision 1.2* (2010)

Scalability Challenges and Solutions in Blockchain Technology



K. Harshini Poojaa and S. Ganesh Kumar

Abstract Bitcoin has come out as a huge cryptocurrency success, thus changing the digital transaction (Nakamoto in *Bitcoin: A Peer-to-Peer Electronic Cash System*. *Decentralized Business Review*, p. 21260, 2008). In the earlier days, even though proof-of-work (PoW) consensus had performance-related issues, it was not considered as a major issue. The transaction processing mechanism is 3.3 to seven transactions per second with smallest 200–250 byte transactions. While this was sufficient at the beginning, the system has been overloaded over the years resulting in low transaction speed and outrageous transaction cost. Massive growth in cryptocurrency results in scalability issues in the blockchain. With more companies trying to shift their existing system to blockchain, it would be difficult to tackle with the existing PoW consensus caused by its scalability issues (Xie et al. in *IEEE Network* 33:166–173, 2019). The purpose of this research is to fix the scalability issues and increases the transaction speed by applying techniques such as lightning network, plasma cash, and hard/soft forks.

Keywords Cryptocurrency · Bitcoin · Proof-of-work · Scalability · Consensus

1 Introduction

Nowadays, the world is moving to computer-based technology because of disadvantages of typical brick-and-mortar banks. Electronic commerce is one of the main uses of the computer-based technology. The digital transactions are highly secure. Digital transactions include credit, debit card transactions, digital checks, etc. Security for these transactions is done using the digital signatures [3]. This sticks a person or an entity to a digital data. In public key cryptography, each user has a public key and a secret key. Digital signature is the secret key which is verified by the public

K. Harshini Poojaa (✉) · S. Ganesh Kumar
SRM Institute of Science and Technology, Chennai, India
e-mail: hk8268@srmist.edu.in

S. Ganesh Kumar
e-mail: ganeshk1@srmist.edu.in

key. Digital signature is used while sending payments from one party to another using the financial institution. But double-spending [4] is the risk of this. Bitcoin is the first technology to solve double-spending problem. Bitcoin is a decentralized digital currency which works on blockchain platform [1]. Since blockchain helps to reduce the risks, frauds, and brings transparency, it is a promising technology [4]. Blockchain is one of the revolutionary technologies, since it is decentralized network [3, 5]. This is one of the reasons why it is called as distributed ledger technology [6]. It distributes the data rather than copying or transferring them. Blocks, nodes, and miners are the three main concepts of blockchain. Every chain consists of multiple blocks for storing the data, and this block is created by the miner through a process called as mining. And nodes are the device which maintains the blockchain [1]. Figure 1 explains the components of a block in the blockchain network. A block contains the block header, hash value of the current block and the hash value of the previous block, timestamp, nonce, and the block data. A block is identified with the help of the block header. Since the blocks are chained to each other, the hash of the previous block is mentioned without which the chronology and the connection would get affected. The timestamp is used to determine at what time the block has been mined and added to the blockchain. The nonce is a onetime random 32-bit number which has been solved by the miners and added to the hashed block.

Gupta [7] Decentralized, peer-to-peer, transparency, and irreversibility of records are the four most alpha and omega of blockchain.

1. **Decentralized:** Decentralization makes blockchain to stand out from traditional technologies. Mostly, decentralization happens in the public domain [8]. Blockchain being a decentralized network stores the data across the network and ensures it does not get hacked or lost [9].
2. **Peer-to-peer:** The transactions happening in blockchain are always peer-to-peer [9, 10] that means at least two parties are involved. With bitcoin and other, since blockchain is transparent, anyone in the network can view all the information of that network which has not existed in financial systems.
3. **Transparency:** Since blockchain is transparent [9], anyone in the network can view all the information of that network which has not existed in financial systems.

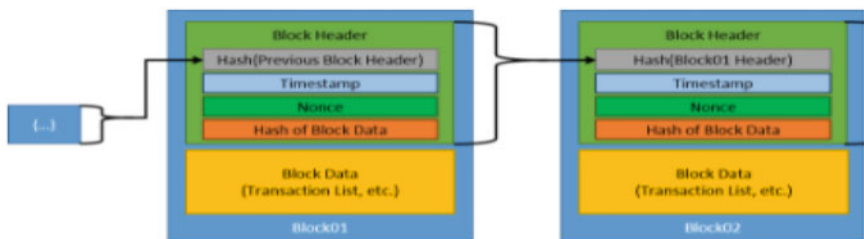


Fig. 1 Blockchain block (source <https://www.nist.gov/blockchain>)

4. **Irreversibility of records:** Once a transaction is completed and updated to the blocks, the records cannot be tampered [11]. This is mainly because all the blocks are linked to the previous block and so does the records [12]. In addition to that, a block contains the details of the previous block which makes it immutable.

With all the interesting features that blockchain provides, it has attracted a lot of users. Blockchain is the solution for many problems these days. But when the number of users has increased broadly, the scalability issues have majorly affected the blockchain systems. This paper attempts to discuss various scalability challenges and its solutions.

2 Scalability Issue in Blockchain

Bitcoin and other cryptocurrencies have attracted a lot of users. When bitcoin was introduced in 2009, the transaction speed of the block has been compromised for Deroad to centralization and security [8]. This is well explained in the blockchain trilemma. The number of transactions has grown and will continue to grow over the years [13]. The number of transactions in ethereum increased from just 3000 in Oct 2015 to over one million at some point in time in Jan 2018 (Fig. 2).

The key metrics for scalability are maximum throughput, latency, and transaction cost.

2.1 Throughput

It is one of the key metrics for finding the performance of a blockchain network. Throughput is the rate of processing. In blockchain [1], only seven transactions are processed in one second which is very low compared to Visa and PayPal. On the other hand, Visa processes around 4000 transactions per second and PayPal does 200 transactions per second.

2.2 Latency

Latency is also called as the block time is the total time that is required for the generation of next block or the total time, and a user has to wait to see the result of their transaction in the blockchain network. In bitcoin, the time taken for the confirmation of a transaction is around 10 min whereas in ethereum it is relatively fast, i.e., 15 s [14].

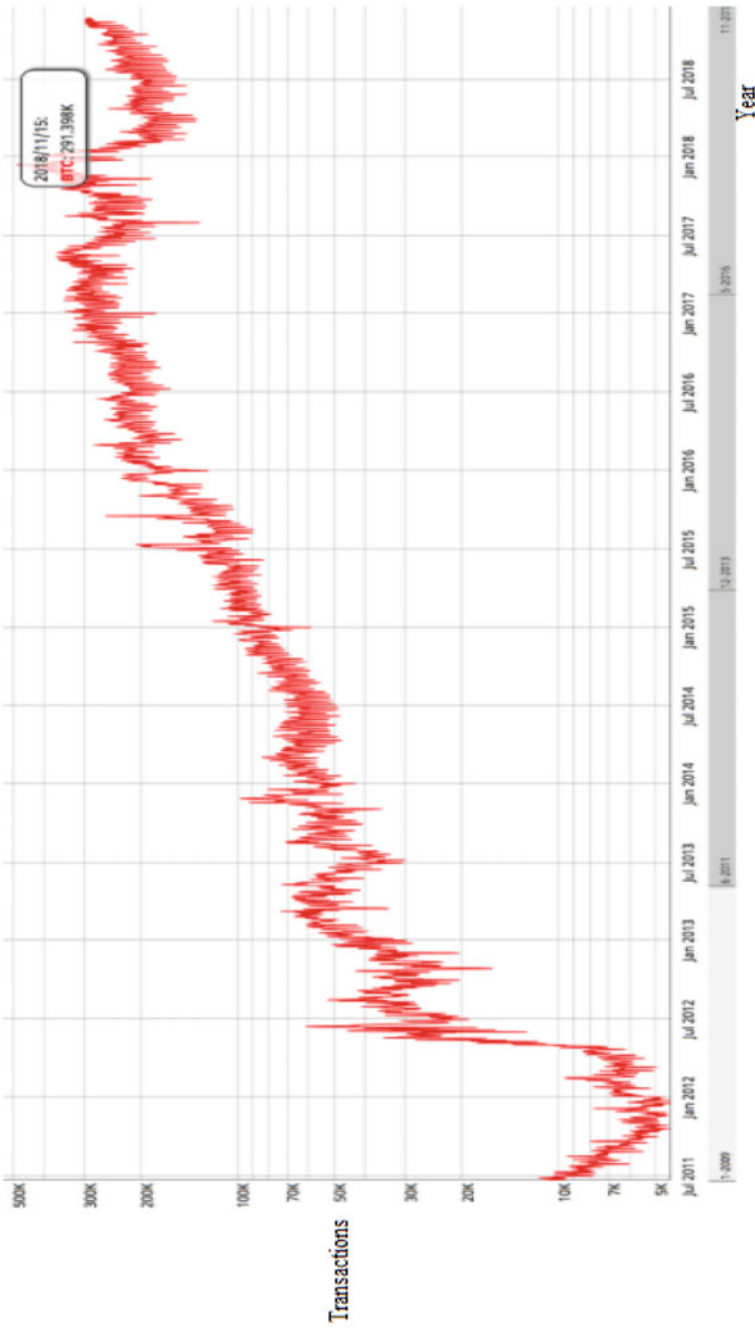


Fig. 2 Daily transactions on bitcoin since 2011 (*source* <https://www.codementor.io/blog/blockchain-scalability-5rs5ra8eej>)

2.3 Transaction Cost

The transaction fee is the price that is charged for a transaction to get into the one MB block size that happens every ten minutes. The average fee per transaction is around \$1.63.

As blockchain has attracted a lot of users, the scalability problem has started to show up and resulted in few drawbacks such as,

1. Increase in the time of confirmation for a transaction,
2. Increase in the transaction fee,
3. Increase in the computational power leading to high energy consumption, and
4. System becomes slower and unsustainable.

3 Solutions for Scalability Issue

3.1 Hard Fork

Forks are the split in the network when the participants in the network does not agree on a certain solution. Hard fork [15] is not compatible to its backward direction [6, 16]. So it creates a network on its own, and the history is stored in the newly created node. One of the most important hard forks is the bitcoin cash [17, 18]. Bitcoin cash remains one of the most successful hard forks of the cryptocurrency as of June 2021. The network was able to successfully increase the block size to 32 MB.

3.2 Soft Fork

Soft fork is compatible to its backward direction [6, 16, 19]. SegWit is the name used for the implementation of soft fork in bitcoin. SegWit is one of the most popular features of the bitcoin side chain. Through SegWit, the block size was increased up to 4 MB holding 8000 transactions. SegWit allows the data to be stored inside chain [20]. Once the transaction is completed in the main chain, the details of the transaction are sent to the side chain freeing up space for upcoming transactions in the main chain (Fig. 3).

3.3 Lightning Network

Lightning network [21] is an instant transaction channel and layer two technology applied to bitcoin to scale its blockchain. It allows the confirmation of transaction without recording them [22]. This is very comfortable way of transactions since it

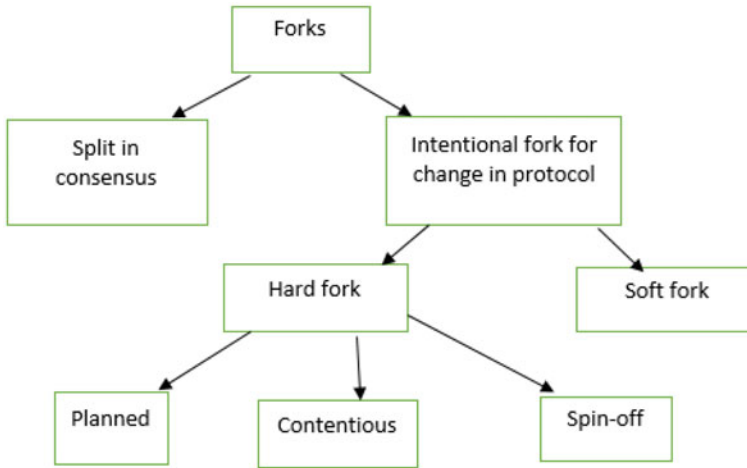


Fig. 3 Hard fork and soft fork

will not mess up the network, i.e., any participants can send or receive money from other participant and confirming the transaction without recording them in the main node. This helps solving the scalability problem using double-signed transactions.

For double-signed transaction, the parties involved in the transaction should create a channel for sending and receiving money [6]. The channel is created by creating a multisig wallet. The wallet can be accessed with their private keys. Once the channel is created, cryptocurrency should be deposited by both the parties. The parties can send money to each other multiple time after the deposit [23].

Ownership rights are also sent along with the money. The assets are distributed once the channel is closed, and data are recorded in the main chain. This ensures low fees and instant micropayments. The speed of the payment is reduced, since this is an off-chain micropayment system (Fig. 4).

3.4 Plasma Cash

Plasma cash was introduced by ethereum in 2018 to solve their scalability issues. Plasma cash required two chains such as ethereum chain and plasma side chain. Ethereum chain provides the security for the network, whereas faster and cheaper transactions are achieved by the plasma side chains [24]. Since plasma side chain is able to follow any consensus mechanism, it can potentially handle thousands of transactions per second (Fig. 5).

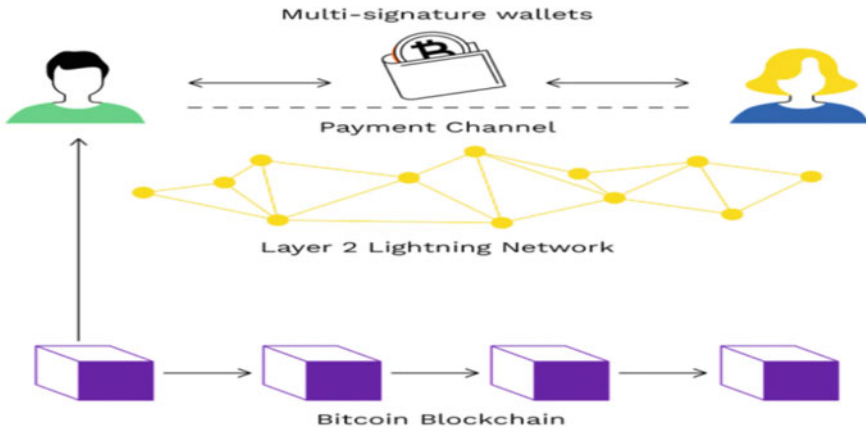


Fig. 4 Lightning network (source <https://www.bitpanda.com/>)

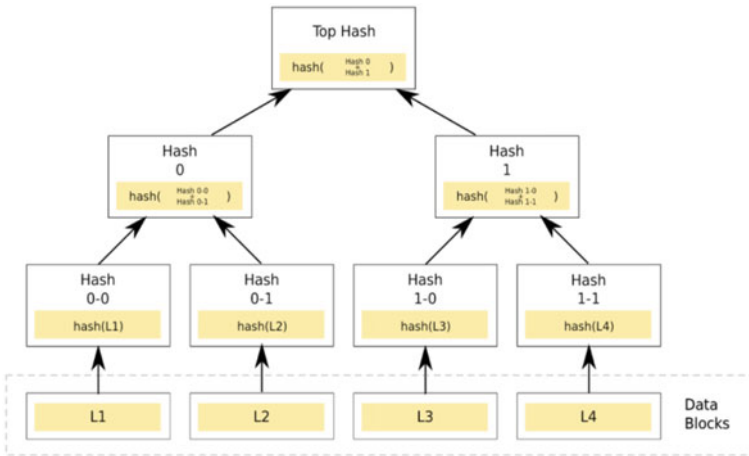


Fig. 5 Plasma cash (source <https://hackernoon.com/>)

3.5 Sharding

Sharding is one of the techniques that is used to solve the scalability issue in blockchain [25, 26]. In this technique, the network is split into smaller portions called as shards. Each shard has its own unique data. In other words, it spreads the data into single nodes through which node will be responsible for the data in it. Rather than having the transaction data in all the nodes of the network, in sharding technique, it is stored in only one node of the shard [27]. The main concern of sharding is that one shard can take over another shard resulting in security issue (Fig. 6).

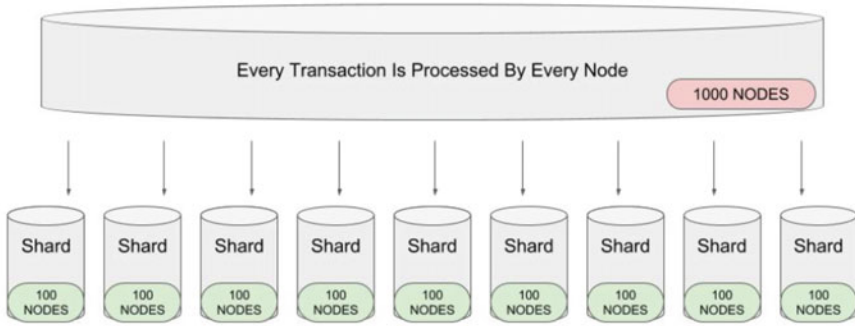


Fig. 6 1000 nodes can divide into ten shards (100 nodes each) to achieve 10 × performance (*source* <https://genesisblockhk.com>)

4 Summary of Existing Blockchain Scalability Mechanisms

Table 1 Existing scalability mechanisms and its advantages and disadvantages

Solution	Advantages	Disadvantages
Hard fork [15, 16]	1. No transaction fee 2. Has customizable and cheap transactions	1. Adjustment in network’s computing complexity
Soft fork [16, 20]	1. Reduce in transaction size 2. Reduced transaction fee	1. Not supported by all wallets
Lightning network [21–23]	1. Increase in throughput 2. Almost no transaction fee 3. Less waiting time	1. Payment channel only
Plasma cash [24]	1. Increase in throughput 2. Tree structure of parent–child blockchain	1. Expensive verification
Sharding [26, 27]	1. Quickly apparent	1. Security issue

5 Analysis of Existing Solutions

From the analysis of Table 1, it is very clear that every solution has its own advantages and disadvantages. Among all the solutions, plasma cash and lightning network have an increase in the throughput of the blockchain network whereas sharding is quick in its own ways. In addition to these solutions, a new Byzantine fault tolerance called as high performance and scalable Byzantine fault tolerance has been proposed through which the communication complexity has been reduce making it more scalable [28]. In spontaneous sharding [29], only a part of the transaction record through which the throughput is attained. MOCA consensus [30] helps to achieve the scalability by sending the information required to all the participating nodes at the start of the consensus. Another consensus called as the parallel proof-of-work [31] has been implemented through which the scalability has been increased by 34%. A secure sharding protocol with the Byzantine challenges called as ELASTICO [32] has been uniformly partition the network into smaller systems through which the scalability has been acheived. A Byzantine fault tolerant protocol called as the bitcoin-NG [33] scales, the network by performing large-scale experiments using the unchanged clients. By mixing proof-of-work and Byzantine fault tolerance, a new protocol has been formed called as PBFT protocol [34]. But the number of nodes is very critical when the both protocols get mixed up. Conflux [35] is a scalable blockchain system which decreased the throughput restriction which in turn increased the processing capability. Proof-of-property [36] is a consensus in which the transactions are validated without the complete knowledge of the address in the system.

6 Proposed Methodology

In this paper, a methodology has been proposed which will be used to reduce the scalability problem. The size of the block can be increased by using segregated witness (SegWit) protocol for reducing the size of the transaction data. An efficient hashing algorithm such as DSA must be used to generate a short signature by feeding hash value and the private key to the signature algorithm through which the size of signature is reduced which in turn decreases the size of the block. Side chain is created using the sharding technique through which the branches are created. The branch is used to store the transaction data, whereas the main block is used to store the metadata of the transaction which reduce the size in the main chain thereby achieving the scalability (Fig. 7).

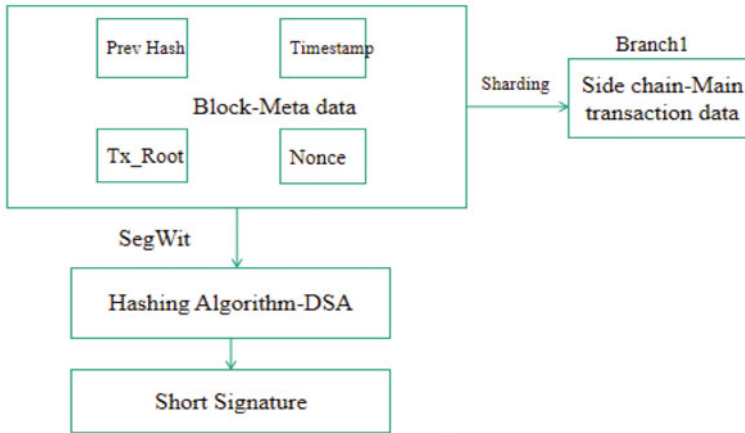


Fig. 7 Proposed methodology

7 Conclusion and Future Work

In this paper, an attempt has been made to discuss the scalability challenges and solutions for the blockchain network. Since blockchain has attracted many users over the years, it is very important to solve the scalability issues. As the number of transaction increases, it eventually increases the confirmation time. Many solutions have been addressed for solving the scalability problem, but it has its own advantages and disadvantages. Sharding and hard forks can be used to reduce the scalability issue in blockchain by reducing the burden of the main chain.

For future work, it has been planned to evaluate the proposed solution and deploys the same in the bitcoin test environment such as testnet and regression test mode to find the increase in the transaction speed through which the scalability issue will be reduced. The main focus would be on the healthcare field [37] since a private and secured data should be available [38].

References

1. S. Nakamoto, Bitcoin: a peer-to-peer electronic cash system, in *Decentralized Business Review* (2008), p. 21260
2. J. Xie, F.R. Yu, T. Huang, R. Xie, J. Liu, Y. Liu, A survey on the scalability of blockchain systems. *IEEE Network* **33**(5), 166–173 (2019)
3. A.A. Monrat, O. Schelén, K. Andersson, A survey of blockchain from the perspectives of applications, challenges, and opportunities. *IEEE Access* **7**, 117134–117151 (2019)
4. M. Di Pierro, What is the blockchain? *Comput. Sci. Eng.* **19**(5), 92–95 (2017). <https://doi.org/10.1109/MCSE.2017.3421554>

5. D. Vujičić, D. Jagodić, S. Randić, Blockchain technology, bitcoin, and ethereum: a brief overview, in *2018 17th International Symposium Infoteh-Jahorina (Infoteh)* (IEEE, 2018), pp. 1–6
6. S. Kim, Y. Kwon, S. Cho, A survey of scalability solutions on blockchain, in *2018 International Conference on Information and Communication Technology Convergence (ICTC)* (IEEE, 2018), pp. 1204–1207
7. S.S. Gupta, Blockchain. *IBM Onlone* (2017). <http://www.IBM.COM>
8. A. Chauhan, O.P. Malviya, M. Verma, T.S. Mor, Blockchain and scalability, in *2018 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C)* (IEEE, 2018), pp. 122–128
9. D. Yaga, P. Mell, N. Roby, K. Scarfone, Blockchain technology overview, arXiv preprint [arXiv: 1906.11078](https://arxiv.org/abs/1906.11078)
10. X. Zuo, A. Iamnitchi, A survey of socially aware peer-to-peer systems. *ACM Comput. Surveys* (CSUR) **49**(1), 1–28 (2016)
11. F. Hofmann, S. Wurster, E. Ron, M. Böhmecke-Schwafert, The immutability concept of blockchains and benefits of early standardization, in *2017 ITU Kaleidoscope: Challenges for a Data-Driven Society (ITU K)* (IEEE, 2017), pp. 1–8
12. E. Landerreche, M. Stevens, On immutability of blockchains, in *Proceedings of 1st ERCIM Blockchain Workshop 2018, European Society for Socially Embedded Technologies (EUSSET)* (2018)
13. S. Bano, M. Al-Bassam, G. Danezis, The road to scalable blockchain designs. *USENIX; login: magazine* (2017)
14. M. Kuzlu, M. Pipattanasomporn, L. Gurses, S. Rahman, Performance analysis of a hyperledger fabric blockchain framework: throughput, latency and scalability, in *2019 IEEE International Conference on Blockchain (Blockchain)* (IEEE, 2019), pp. 536–540
15. N. Webb, A fork in the blockchain: income tax and the bitcoin/bitcoin cash hard fork. *North Carolina J. Law Technol.* **19**(4), 283 (2018)
16. Hard Forks and Soft Forks Explained, [Online]. Available: <https://academy.binance.com/en/articles/hard-forks-and-soft-forks>
17. Bitcoin Cash, [Online]. Available: <https://www.bitcoincash.org/>
18. M.A. Javarone, C.S. Wright, From bitcoin to bitcoin cash: a network analysis, in *Proceedings of the 1st Workshop on Cryptocurrencies and Blockchains for Distributed Systems* (2018), pp. 77–81
19. F.M. Benčić, I. Podnar Žarko, Distributed ledger technology: blockchain compared to directed acyclic graph, in *2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)* (2018), pp. 1569–1570. <https://doi.org/10.1109/ICDCS.2018.00171>
20. C. Pérez-Solà, S. Delgado-Segura, J. Herrera-Joancomarti, G. Navarro-Arribas, Analysis of the SegWit adoption in bitcoin (2019). https://deic-web.uab.cat/~guille/publications/papers/2018_recsi.segwit.pdf. Visited on 06/13/2020
21. J. Poon, T. Dryja, The bitcoin lightning network: scalable off-chain instant payments (2016)
22. Lightning Network, [Online]. Available: <https://www.investopedia.com/terms/l/lightning-network.asp>
23. J. Poon, T. Dryja, Lightning network (2015)
24. G. Konstantopoulos, Plasma cash: towards more efficient plasma constructions (2019), arXiv preprint [arXiv:1911.12095](https://arxiv.org/abs/1911.12095)
25. Q. Zhou, H. Huang, Z. Zheng, J. Bian, Solutions to scalability of blockchain: a survey. *IEEE Access* **8**, 16440–16455 (2020)
26. X. Feng, J. Ma, Y. Miao, Q. Meng, X. Liu, Q. Jiang, H. Li, Pruneable sharding-based blockchain protocol. *Peer-to-Peer Network. Appl.* **12**(4), 934–950 (2019)
27. S.S.M. Chow, Z. Lai, C. Liu, E. Lo, Y. Zhao, Sharding blockchain, in *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)* (2018), pp. 1665–1665. https://doi.org/10.1109/Cybermatics_2018.2018.00277

28. Y. Jiang, Z. Lian, High performance and scalable Byzantine fault tolerance, in *2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)* (IEEE, 2019), pp. 1195–1202
29. Z. Ren, K. Cong, T. Aerts, B. Jonge, A. Morais, Z. Erkin, A scale-out blockchain for value transfer with spontaneous sharding, in *2018 Crypto Valley Conference on Blockchain Technology (CVCBT)* (2018), pp. 1–10. <https://doi.org/10.1109/CVCBT.2018.00006>
30. Z. Wang, MOCA: a scalable consensus algorithm based on cellular automata, in *2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS)* (2018), pp. 314–318. <https://doi.org/10.1109/ICSESS.2018.8663808>
31. S.S. Hazari, Q.H. Mahmoud, A parallel proof of work to improve transaction speed and scalability in blockchain systems, in *2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)* (2019), pp. 0916–0921. <https://doi.org/10.1109/CCWC.2019.8666535>
32. L. Luu, V. Narayanan, C. Zheng, K. Baweja, S. Gilbert, P. Saxena, A secure sharding protocol for open blockchains, in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security* (2016), pp. 17–30
33. I. Eyal, A.E. Gencer, E.G. Sirer, R. Renesse, Bitcoin-NG: a scalable blockchain protocol, in *13th {USENIX} symposium on networked systems design and implementation ({NSDI} 16)* (2016), pp. 45–59
34. M. Vukolić, The quest for scalable blockchain fabric: proof-of-work versus BFT replication, in *International workshop on open problems in network security* (Springer, Cham, 2015), pp. 112–125
35. C. Li, P. Li, D. Zhou, W. Xu, F. Long, A. Yao, Scaling nakamoto consensus to thousands of transactions per second (2018), [arXiv:1805.03870](https://arxiv.org/abs/1805.03870). [Online]. Available: <https://arxiv.org/abs/1805.03870>
36. C. Ehmke, F. Wessling, C.M. Friedrich, Proof-of-property: a lightweight and scalable blockchain protocol, in *Proceedings of the 1st International Workshop on Emerging Trends in Software Engineering for Blockchain* (2018), pp. 48–51
37. R. Saranya, A. Murugan, A systematic review of enabling blockchain in healthcare system: analysis, current status, challenges and future direction. *Mater. Today: Proc.* (2021)
38. M. Mohideen AbdulKader, S. Ganesh Kumar, Privacy challenges and enhanced protection in blockchain using erasable ledger mechanism, in *Expert Clouds and Applications*, ed. by I. Jeena Jacob, F.M. Gonzalez-Longatt, S. Kolandapalayam Shanmugam, I. Izonin. *Lecture Notes in Networks and Systems*, vol. 209 (Springer, Singapore, 2022). https://doi.org/10.1007/978-981-16-2126-0_16

An Open-Source Framework Unifying Stream and Batch Processing



Kiran Deshpande and Madhuri Rao

Abstract Log monitoring and analysis plays critical role in identifying events and traces to understand system behaviour at that point in time and to ensure predictive, corrective actions if required. This research is centered towards modelling open-source framework meant for real-time and historical log analytics of IT infrastructure of an educational institute consisting of application servers hosted over Internet and Intranet, peripheral firewalls and IoT devices. Modelling such framework has not only enhanced processing speed of real-time and historical logs through stream processing and batch processing, respectively, but also facilitated system administrators with critical security incidents monitoring and analysis in near-real time. It also allowed forensic investigations on indexed historical logs stored after stream processing by using batch processing. The modelled framework provides open-source, efficient, user-friendly, enterprise-ready centralized heterogeneous log analysis platform with fast searching options. Open-source tools like Apache Flume, Apache Kafka, ELK Stack and Apache Spark are used for log ingestion, stream processing, real-time search and analytics and batch processing, respectively, in this work. Arriving at a novel solution to unify big data processing paradigms stream and batch processing for log analytics, we propose an approach that can be extrapolated to a generalized system for log analytics across a large infrastructure generating voluminous heterogeneous logs.

Keywords Big data · Batch processing · Stream processing · Heterogeneous log analytic · Apache Spark · ELK Stack · Apache Kafka · Performance evaluation

1 Introduction

Taking into consideration the increase in demand for large-scale data processing, there is enhancement in researcher's interest in batch processing, which involves offline processing of large volumes of data at rest, as well as stream processing,

K. Deshpande (✉) · M. Rao

Thadomal Shahani Engineering College, University of Mumbai, Mumbai, Maharashtra, India
e-mail: madhuri.rao@thadomal.org

which involves online processing of large quantities of data in motion. Since era of big data is upon us, applications requiring support for both stream and batch processing are increasing, there is a requirement to develop common framework which will unify these both big data processing methods and offer support for both. Significant research has been observed in development of extended functionalities for batch processing frameworks for offering support for stream processing due to popularity of batch processing frameworks. But since current era of big data is more focused towards stream processing, there is requirement of novel framework, where data in motion is processed through stream processing framework at foundation, and data at rest will be processed through batch processing framework built on top of stream processing framework [1, 2].

Recent research in big data has been not only focused towards in-depth analysis of the data from past to find useful insights so that decision making capability will become more precise, but also focused towards handling the notion of real-time big data [3, 4]. Using big data generated smartly to gain valuable decision support becomes important. This would only happen if we analyse all the data we have and get important insights and directions. The main bottleneck in the big data analysis is to process data as soon as it is generated if possible in real time to gain important insight [4]. Big data analysis solution which will address this bottleneck needs to be very adaptable because of information technology evolution. Increasing demand for big data analytics has given rise to development of plethora of data processing systems in recent years, offering a broad range of features and capabilities [4]. The data processing systems developed can be categorized into either a batch processing which focuses on processing data at rest or stream processing system that processes data in motion [5]. However, in current era of big data, applications that can support both processing paradigms will provide more benefits to organizations [2]. For example, the detection of abnormal and malicious events through real-time monitoring of stream-based applications. Classification of the event through offline analysis, its correlation and taking appropriate actions can be initiated [6, 7]. The framework offering both batch and stream processing needs to be modelled to provide execution environment to develop applications using both real-time and offline analysis [2]. Such framework can serve need of next generation big data analysis in huge way.

Every organization's working is heavily dependent on Internet. It becomes very important to analyse Internet and Intranet traffic so that abnormal and malicious events can be identified which may hamper organization's security and reputation [6, 7]. Monitoring the logs and system performance of critical applications like Web server, proxy server, peripheral routers and firewalls which contains crucial events related to Intranet and Internet activity [8–11]. Log data is often voluminous and is continuously growing [7, 9]. In order to have forensic analysis of events, real-time as well as historical data is required to find out in-depth correlation. In such scenario, for real-time log ingestion, processing, faster storage, retrieval and search of such big data technologies like Apache Flume, Apache Kafka, ELK Stack, ES-Hadoop, Hive, Apache Hadoop and Apache Spark can prove efficient.

Major security disasters consist of a series of steps. So if we can identify preceding steps to major security disasters, at the first occurrence itself, we may avoid them from happening [6]. Managing organization information technology infrastructure securely is critical. Until and unless respective traffic is monitored, analysed and correlated, the security flaws present in system, network and application servers deployed cannot be identified. In any medium-scale organization with IT infrastructure-dependent services, log generation is voluminous which needs to be collected and analysed in real-time and to be stored for historical analysis to keep the security posture of the organization in sync with security policy of the organization [7]. Here, stream and batch analytics of logs can play critical role and can help administrators to keep track of various security events and initiate proactive actions [6].

Logs generation is continuous process resulting in voluminous log data with different formats and rates and can be used for obtaining meaningful insights through proper and effective analysis. Introduction of Cloud computing and parallel computation frameworks supporting much required log analysis can help in managing and analysing data of large size with a high production rate [12, 13]. The distributed computing paradigm can address need of resources required for on-demand storage and computing of log mining [8, 9]. Taking into consideration current era of big data and significance of log analysis, several open-source and commercial solutions were designed for implementing log collection, storage, search, analysis and visualization. Taking into consideration advantages and disadvantages of Spark [14, 15], Kafka [16] and ELK Stack [17, 18] regarding log mining, integrating Spark and Elasticsearch capabilities with open-source tools like Apache Flume, Apache Kafka can be advantageous for creating an efficient framework for scalable log management and analysis [16, 19]. Such integration can also prove as milestone for unifying much required stream processing and batch processing capabilities considering current need of big data log analytics.

In this paper, we propose a centralized heterogeneous log analysis framework integrating Apache Flume, Apache Kafka, ELK Stack and Spark. The framework aims at presenting platform unifying stream and batch processing and providing real-time search and analytics by ensuring full use of the functionalities offered by all these platforms listed earlier. It also can serve as a guide for implementing Elasticsearch-based data analysis after performing stream processing and batch processing to cater the current need of log analytics. Thus, the proposed framework will provide open-source, enterprise-ready platform and solution for heterogeneous real-time log monitoring, analysis which will provide better insights for faster trouble shooting and can be widely used across multiple enterprises and domains. To the best of our knowledge, this is the first effort which integrates capabilities of Apache Flume, Apache Kafka, ELK Stack and Apache Spark for big data log analytics to address most of the big data log analytics requirements and challenges listed in Sects. 1.2 and 1.3, respectively.

The rest of the paper is organized as follows. In addition to Introduction (Sect. 1), Sect. 2 discusses related work, Sect. 3 includes major objectives of this research. Section 4 introduces the proposed system architecture for log analytics. Section 4

includes details of technological stack used in proposed framework. Section 5 delineates implementation architecture through FOSS Tools. Section 6 details the experimental environment as well as interprets and evaluate results achieved. The last section, Sect. 7, summarizes conclusions delivered with a brief discussion of future work.

1.1 Note on Big Data Processing Paradigms

Big data analytics is the process of using analysis methods running on powerful supporting platforms to discover potential insights hidden in big data, such as unseen patterns or unknown correlations. Big data analytics can be categorized into two varying paradigms as per their processing time requirements [1, 5].

- **Streaming Processing:** Streaming processing paradigm works on the assumption that the potential insights of data depends on data freshness [20]. Thus, the streaming processing paradigm ensures analysis of data as soon as possible to derive insights. In this paradigm, data arrives in a stream with continuous arrival. To find approximation results, one or few passes over the stream are made. Representative open-source systems including Storm and Kafka [3] can be used to see impact and application of streaming processing paradigm in online application which require data processing at the second, or even millisecond, level [1, 5].
- **Batch Processing:** Basis for batch processing paradigm is that data is first stored and then used for analysis [4]. MapReduce has become most popular and dominant batch processing model. MapReduce divides data into small chunks; then these chunks are processed in parallel and in a distributed manner to generate intermediate results. All the intermediate results are aggregated to derive final result. Batch processing MapReduce model schedules computation resources close to data location to avoid the communication overhead during data transmission [1, 5]. Wide adoption of MapReduce model is seen in bioinformatics, Web mining and machine learning [21].

1.2 Why Log Analysis Is Required

Log analysis is the process of finding meaningful insights from system, network and application server generated logs refereed as log events. Log analysis helps in providing useful metrics by which administrators can know events happened across the infrastructure and can use this information in order to improve or solve performance issues within an application or infrastructure. To ensure mitigation of risks, comply with security policies and understand online user behaviour, proactive and reactive log analysis can help organizations. Log information is much needed in different security perspective. Log analysis is required for:

- **Debugging Information:** Enabling logging in applications and devices can help in checking for a specific error message or event occurrence which may help in debugging.
- **Performance Evaluation:** Logs information can play very important role in optimizing resource utilization as well as in finding out performance related issues. To understand system, application, network health and performance over date and time information written in logs is crucial.
- **Security Evaluation:** To manage the system, network and application security of any organization log analysis play crucial role. To detect security breaches, application misuse and malicious events it can be helpful.
- **Predictive analysis:** For futuristic analysis of threats, flaws, traffic patterns, security policy design, resource optimization information gathered thorough log analysis can help administrators.
- **Internet of Things and Logging:** Log analysis is also very important to know the status and health of the IoT devices for their uninterrupted operation since effective management of such devices is dependent on logs and alerts generated by them.

1.3 Challenges in Big Data Log Analysis

Even if log analysis is much useful in current era taking into consideration advantages discussed earlier, but it imposes lot of challenges which needs to be resolved. Searching, analysing, correlating and visualizing system, application and network log generated by the IT and network infrastructure will play crucial role to gain meaningful insights. Manual log analysis goes beyond human capabilities. Log analysis involves following major challenges:

- **Heterogeneous log format:** Every application and device has different log format. Understanding different logs formats and searching across different format can be time consuming.
- **Different time format:** Every log record has date and time which is crucial for log analysis. Correlation of log events interpreting incorrect date and time becomes difficult.
- **Decentralized log:** Application servers, devices are distributed over the network, it is difficult to monitor, handle logs of application servers, devices until centralization of the logs is done.
- **Storage and retrieval:** Voluminous nature of logs makes storage, retrieval and processing difficult. Secure storage of logs generated is also crucial since it contain lot of security information and details.

2 Related Work

With the prospective of finding value from big data in hand, research of big data analysis means and tools is increasingly gaining popularity. The survey paper [1] is an attempt to analyse the definition, framework and typical big data processing systems of big data. This especially focuses not only on conceptual illustration and comparison of batch data processing system and stream data processing system but also focuses on hybrid processing system. There has been a lot of work on both batch and stream processing over the last decade. The seminal paper [2] presents unification of stream processing and batch processing through common computing framework. It discusses development of middle layer between MapReduce applications and the streaming platform so that benefits of stream processing can be combined with the ease of programming and familiarity of MapReduce batch processing [2]. In Li et al. [7] presented a cloud-based log-mining framework using Apache Spark and Elastic-search to speed up log analysis process of HTTP and FTP access logs. The paper [3] represents evaluation of distributed stream processing platforms like Apache Storm, Apache Spark and Apache Flink for IoT applications with their advantages and disadvantages. With respect to issues identified in the survey paper [20], streaming analytics can be considered as an emerging research area focusing on key issues like scalability, integration, fault tolerance, timeliness, consistency, heterogeneity and load balancing. In [8] work presented by Deepak Mishra et al. on batch processing through popular big data frameworks, Apache Hadoop and Apache Spark, concludes Spark performs better than Hadoop after comparing Hadoop and Spark's performance. The seminal paper [22] discusses Apache Spark-based Analytics of Squid Proxy Logs for studying traffic behaviour and identifying threats by generating Internet traffic statistics like top domains accessed and top users. The review paper [14] discusses use of Apache Spark for big data analytics focusing the key components, abstractions and features of Apache Spark. The conference paper [23] presents a HPC log data analytics framework that is based on a distributed NoSQL database technology and the Apache Spark framework for extracting precise insights useful for system administrators and end users. Experimental analysis presented by Haoxiang and Smys in paper [21] outperforms as compared to other data mining algorithms used for privacy preservation in terms of attack resistance, scalability, execution speed and accuracy.

Few researchers proposed [24, 25] usage of Elastic Stack for easy and rapid management of big data problem of stream processing. In Liu et al. [26] presented cyberattack detection model by making use of ELK Stack for network log analysis and visualization. In this study, network log analysis and management system is designed for providing functions to filter, analyse and present network log data for further processing. DevOps teams are using Docker container technology not only to ensure faster software delivery cycle to boost operational efficiency but also to ensure application portability. In paper, Chen et al. [27] designed Docker container log collection and analysis system by using open source log collection platform ELK, lightweight log collector Filebeat and distributed message queue Kafka.

The seminal paper [28] demonstrated the use of open-source platforms showcasing the feasibility of it by comparing the performance of log analysis between commercial solutions and open-source solutions. Platform that collects [29] the variety of logs using Logstash for identifying the malicious activity in the network proposed by Sanjappa and Ahmed. Author demonstrated use of ELK ecosystem clubbed together for effective analysis of log files in order to get easily understandable insights [25, 29]. Wang et al. [30] presented work aiming to develop a monitoring system using ELK stack to address weakness or unavailable Wi-Fi signal problems.

Wide adoption of log analysis is seen in literature for addressing IT infrastructure security issues. Debnath et al. [31] have implemented machine learning to discover patterns in application logs and used these discovered patterns along with the real-time log parsing for designing advanced log analytic applications. He et al. [32] have proposed system which ensures wide use of logs for managing systems for guaranteed reliability by applying data mining methods. Son and Kwon [28] have emphasized on the reasons because of which the use of open-source tools is preferred over commercial security log analysis systems with huge pricing, complexities and resource requirement. Due to big data properties like volume, variety and rate, it is very difficult to detect the attacks using traditional detection system. More et al. [33] has represented big data technologies use for threat detection.

Data analysis performance can be enhanced through the parallel computation frameworks like Apache Spark and Apache MapReduce using data parallelism. A unified cloud platform with batch analysis and in-memory computing capacity by combining capabilities of Hadoop and Spark [8, 9] was proposed by Lin et al. Large-scale log analytics for detecting abnormal traffic from voluminous logs efficiently by using Hadoop was demonstrated by Therdphapiyanak and Piromsopa.[10]. The ELK stack, i.e. Elasticsearch, Logstash and Kibana, can play important role in developing scalable heterogeneous log analytic platform through its capabilities of automatically collecting, indexing, aggregating and visualizing log data [17, 18]. Efficient geo-identification of website user traffic through generated logs using ELK stack was orchestrated by Prakash et al. [34]. Bagnasco et al. had demonstrated effective use of the Elasticsearch ecosystem for monitoring the infrastructure as a service (IaaS) and scientific application deployed on the cloud [35]. Metha et al. articulated a streaming architecture based on ELK, Spark, and Hadoop for anomaly detection from network connection logs in near-real time [11]. Li et al. introduced a method to speed up log analysis with Elasticsearch and Spark through independent use of Elasticsearch and Spark in their framework [36].

3 Objectives

The Work in this paper is intended towards achieving following objectives through real time and offline log analysis IT Services of an educational Institute. Real-time logs are analysed by stream processing framework Apache Kafka, analysed in near real time by ELK Stack and processed logs will be made available to batch processing

framework Apache Spark for offline analysis as well as forensic investigation of historical logs. Detailed discussion of the same is done in next section.

- To ensure predictive maintenance, troubleshooting of Internet- and Intranet-hosted application servers, peripheral firewall and routers in case of critical security events identified through real-time log analysis.
- To ensure mitigation of malicious security events.
- To ensure real-time performance monitoring of data sources in consideration.
- To conduct forensic investigation through historical data analysis to crosscheck compliance with internal security policies.
- To implement IoT-based environment monitoring system that supports continuous tracking of temperature and humidity levels required to be maintained in centralized server room which hosts the log analytics setup proposed through stream processing of IoT logs of IoT network deployed.

4 Proposed System Architecture

Computing framework is one of the key aspects in improving data analytics and processing efficacy. Since era of big data is upon us, applications requiring support of both stream and batch processing are increasing, and there is a requirement to develop common framework which will unify these both big data processing methods and offer both [2]. Apache Spark Layer at top for batch processing and Apache Kafka, Elastic Stack as bottom layer in proposed framework for real-time log processing, exploration, analytic and search better analysis of security metrics can be done. In general, big data analysis framework can be represented in terms of layered structure, as shown in Fig. 1. It can be categorized into three layers, including device layer, data collection and processing layer and application layer [3–5]. This layered view can help in providing a conceptual clarity to understand working of proposed model and understand complexity of a big data system. The proposed framework can be a part of data collection and processing layer and can be used for data collection and processing through which online and offline analysis of real-time log data generated through data sources in consideration can be done.

- **Device layer:** Device layer includes sources of big data required for analysis, coming in from all heterogeneous sources like application servers, Web access, IoT devices and network. This layer serves as the foundation for the entire real-time big data processing and analysis in consideration.
- **Data Collection and Processing:** Data collection and processing is responsible for receiving data from the data sources and ensures its conversion into a format that is in sync with data analysis in consideration. Data collection layer is not only needed because of increase in data sources but also to ensure integration of multi-source, structured and unstructured data for further analysis. In this layer, streaming data will be send for processing, and the accumulated historical data will be stored so that it can be further analysed with analytical tool based on the requirements of the

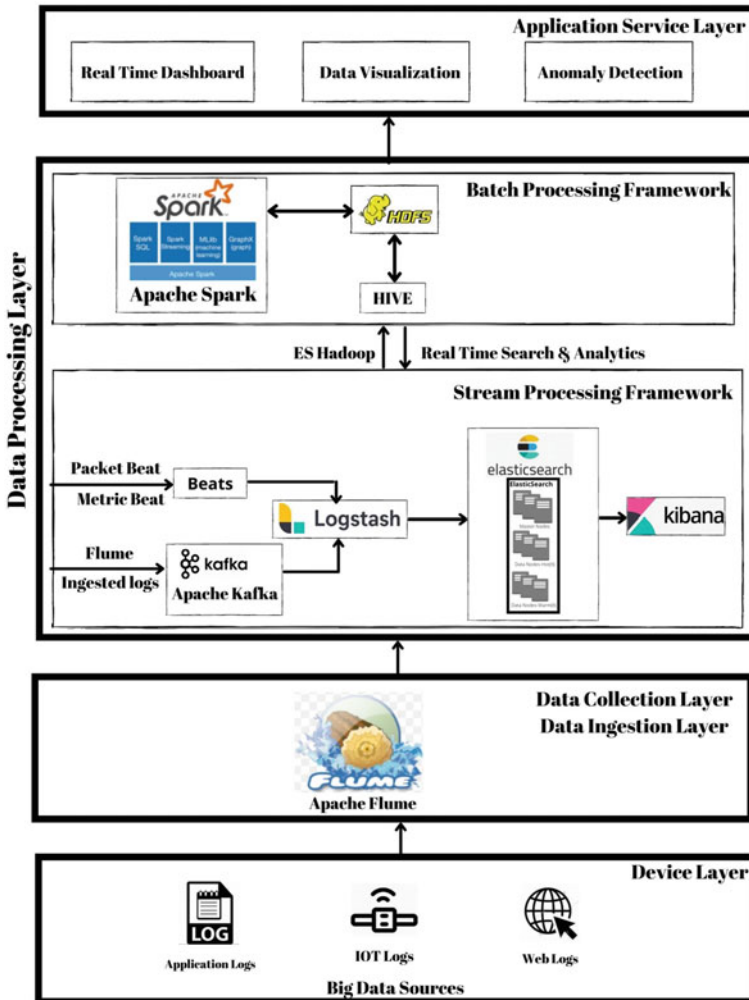


Fig. 1 System architecture

application layer. This layer is the core in processing both big data at fly and big data at rest. Since in current era of big data, sources are extremely heterogeneous in structure and content, the data processing layer ensures parallel computing, data cleansing, data integration, data indexing and so on [3, 5].

- Application Layer: Application layer is the highest layer that uses the interface provided by the data processing layer to perform various data analysis functions like querying, statistical analysis, clustering and classification [5]. It combines basic analytical methods to develop various real-time dash boards required for real-time analysis and performance monitoring of log data in consideration. Along with

the previously discussed layers, application layer can build various applications like recommendation and alert system which can be useful for end users.

5 Implementation Architecture

Figure 2 represents implementation architecture of proposed framework. This section explains working process sequence in detail for proposed framework.

1. Apache Flume is the data ingester. It collects live logs from Intranet, peripheral firewall and application servers as a source and creates a memory-based pipeline to Apache Kafka as a sink.
2. Apache Kafka gets log data from Flume and collects that log data into topics. Kafka uses java instances to manage these topics (jps). Zookeeper works with Apache kafka at the backend for replication, recovery and management of these topics. Here, Apache Kafka uses producer and consumer process (jps). Kafka producer gets log data from Flume and Kafka consumer transfers that log data to logstash for parsing.
3. Logstash is responsible for parsing and filtering of raw log data from Apache Kafka. It reads topics from Kafka consumer and does parsing of that data by creating attribute names. These attribute names are useful for creation of Elastic-search index. At the same time, logstash perform filtering of content by applying filter plugin like grok which matches specific patterns and eliminate data out of that pattern. This helps making data structured and creation of index in Elastic-search easy.

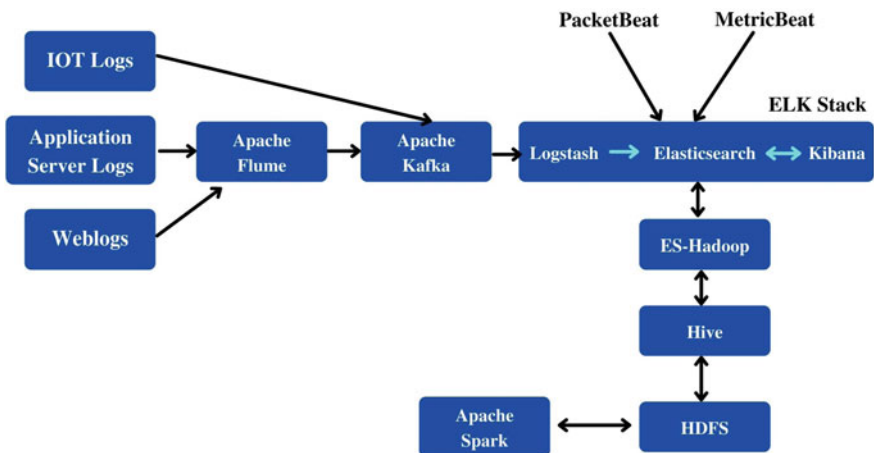


Fig. 2 Implementation architecture

4. Elasticsearch creates index and maps structured data into it. Timestamp-based index mapping can be done over log data to create visualization in Kibana. Visualizations based on Elasticsearch indices helps in creating interpretation of live data in form of bar chart, line graph, pie chart, heat map, gauge etc.
5. ES-Hadoop connector is used to transfer indexed data from Elasticsearch to HDFS. Apache hive is the interface used by Hadoop. Hive provides SQL like interface for data manipulation.
6. Hadoop architecture uses MapReduce algorithm to perform batch processing over HFDS. MapReduce has limitations of processing due to file system read–write operations latency. Here, Apache spark provides 100 times faster solution to MapReduce latency during batch processing. Apache spark architecture makes use of RDD (resilient distributed datasets), data frames and perform transitions on HDFS data by fetching it inside main memory.
7. Processed data is stored back in HDFS by spark-scala user interface. This batch-processed optimum data is again transferred to Elasticsearch through ES-Hadoop connector using hive interface. Finally, Elasticsearch can create visualizations required for deep low latency analytic of batch processed historical data using Kibana.
8. Packetbeat and Metricbeat are responsible for real-time performance monitoring of IT resources, services and Intranet by shipping related metrics to ELK Server.
9. MQTT Broker: The Message Queuing Telemetry Transport (MQTT) is a lightweight, publish–subscribe network protocol that transports messages between IoT devices. The protocol usually runs over TCP/IP; however, any network protocol that provides ordered, lossless and bidirectional connections can support MQTT. Here, MQTT broker is a server that receives all messages from the IoT clients and then routes the messages to Apache Kafka for further analysis through proposed framework.

6 Experimental Environment and Result Analysis

In this section, hardware and software specifications used in proposed framework as well as experimental environment and result analysis are explained. All server machines used for creating experimental environment are physical machines. Ubuntu 14.04 and Centos 7 with 64 bit is adopted as our operating system.

6.1 Experimental Environment

IT infrastructure network architecture of an educational organization in consideration providing services like DNS, Internet, Web, Squid, NIDS and Firewall Security which generates lot of logging and traffic data is represented through Fig. 3. Infras-

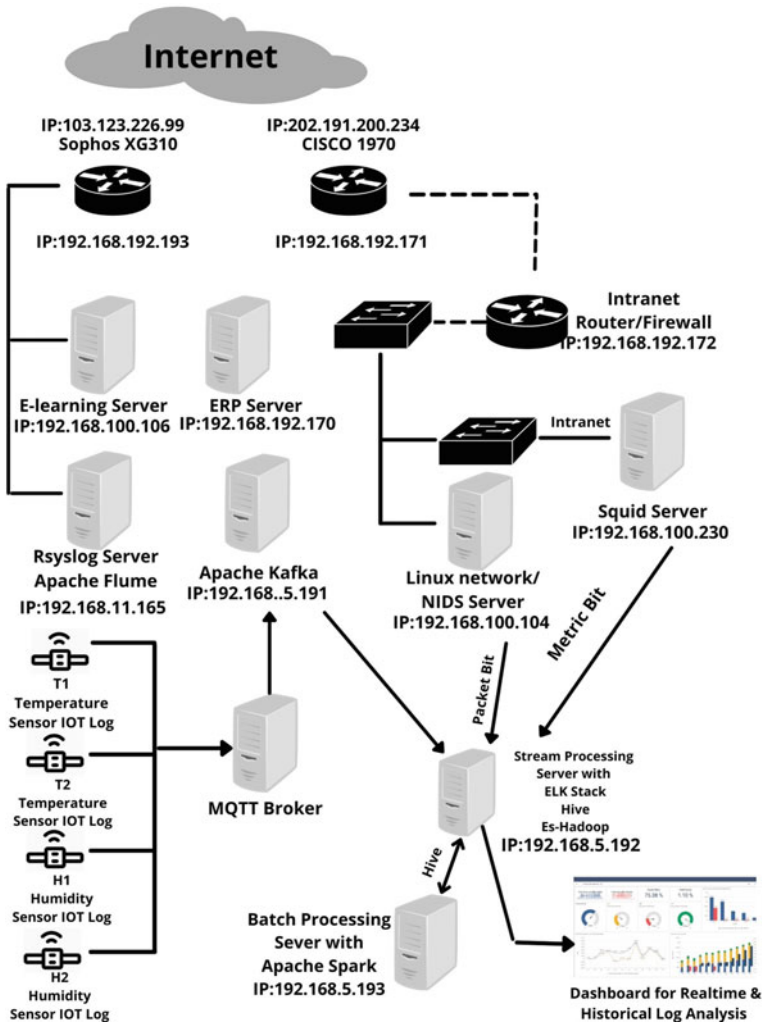


Fig. 3 Block diagram of network architecture setup used for modelling novel framework

tructure also has IoT network deployed through MQTT Broker and Kafka for creating environment monitoring system that supports continuous tracking of temperature and humidity levels required to be maintained in centralized server room which hosts the log analytics setup proposed. All these components are distributed over Local Area Network (LAN). Manually checking logs of each and every server and device on daily basis is not feasible. Syslog has been configured on critical servers and devices like peripheral firewall and Web servers to forward logs to central Rsyslog server so that logs can be made available to Apache Flume for ingesting into framework. Packetbeat and Metricbeat are configured on NIDS, Squid Server to ensure shipping their

logs to ELK Server for real-time network monitoring of Intranet and performance monitoring of Squid Server, respectively.

In implementation scenario, Apache Flume, Apache Kafka and ELK stack have been configured for real-time stream processing, analysis and centralized log management. ES-Hadoop and Hive installed on ELK Server ensure bidirectional movement of indexed logs between ELK and Apache Spark through HDFS which is used as storage for in-depth low latency analytics. This subsection also explains hardware configurations of servers, software configurations of FOSS tools and operating system environments that are used in proposed framework with its purpose.

1. Web Servers: Intel (R) Xeon (R) CPU E5 2603 v3 1.6GHz 48GB Memory with Centos 7 Server Operating System. This is used as source of real-time logs of Apache through which Moodle E Learning Platform, ERP Server is made accessible to 3000 users of an educational institute. Log forwarding is done through rsyslog to server with Apache Flume.
2. Squid Proxy Server with Metricbeat: Intel (R) Xeon (R) CPU X3220 2.4GHz 8GB Memory with centos 7 Server Operating System. Squid Proxy server is meant for providing secure Internet access in Intranet.
3. Network Intrusion Detection (NIDS) Server with Packetbeat: Intel (R) Xeon (R) CPU X3220 2.4GHz 8GB Memory with centos 7 Server Operating System. NIDS Server is meant for monitoring all network traffic of Intranet including switches and routers.
4. Rsyslog Server with Apache Flume: Intel (R) Xeon (R) CPU E5 2603 v3 1.6GHz 48GB Memory with Centos 7 Server Operating System. Apache Flume ingests Apache logs of Web servers, peripheral firewall to ELK Stack. This server can be used as sink for multiple heterogeneous log sources.
5. Server with ELK Stack, Apache Spark and Hive Instances deployed: Intel (R) Xeon (R) CPU E5 2603 v3 1.6GHz 48GB Memory with Ubuntu 18 Server Operating System. ELK Stack is used for real-time log processing, and Spark is used for offline processing of historical logs. Here, hive uses ES Hadoop connector to transfer indexed data to HDFS from Elasticsearch. Similarly, hive gets batch processed data from HDFS and inserts back to Elasticsearch for in depth analytics.
6. Sophos XG310 (SFOS 18.0.5 MR-5-Build586) Firewall: It also serves as log source for Rsyslog Server along with Web server logs through Syslog service configured.

In implementation architecture for log shipping, stream processing, real-time search and analytic, batch processing and unifying stream processing framework to batch processing network following open-source tools are used.

1. Log Ingestion: Apache Flume 1.5.0.1
2. Data Pipeline Tool: Apache Kafka 2.7.0
3. Batch Processing of Historical Logs: Apache Spark 3.0.1 with Prebuilt for Apache Hadoop 3.1
4. Real-time search and analytic: Elastic Stack 7.7.0
5. Connector of ELK Stack and Apache Spark: ES Hadoop 7.11

6. Lightweight shipper for network data: Packetbeat 7.13.1
7. Lightweight shipper for performance metrics: Metricbeat 7.13.1.

6.2 *Result Analysis*

In this section, we will discuss some precise insights through IT Infrastructure log analysis of an engineering institute through proposed framework. This Engineering Institute is A. P. Shah Institute of Technology located at Mumbai, Maharashtra, India. There are 3000 active users including students, faculties and administrative staff who accesses the IT Infrastructure in consideration resulting in 10 million logs per day, approximately. This voluminous log analysis is intended to ensure security of critical infrastructure components like routers, firewall, Proxy server, e-learning server and ERP server of this institute as well as identifying the malicious activities. Log analysis represented in this section helps institute to keep its security posture in sync with security policy framed at institute level which will be discussed later in this section. Big data log analytics of unstructured heterogeneous logs represented here helps administrators to keep track of various events related to security and take proactive actions on basis of that. Also, along with real-time log analysis managed through Kafka and ELK Stack, analysis of historical logs through batch processing implemented through Apache Spark can form basis to revise, develop future security policies required for institute. In addition to log analysis, the proposed framework also ensures real-time monitoring of performance metrics related to IT resources which helps in shifting from reactive to proactive monitoring.

Result analysis presented in this section focuses on identifying following security breaches on Apache Web Service configured on Application Servers in consideration. Following are the Apache Service-related security events in consideration around which result analysis will revolve.

1. Using blocked request method: In most cases, GET and POST are the only request methods required to operate a dynamic website. Allowing more request methods than are necessary increases site's vulnerability. Thus, finding request methods which are blocked through real-time log analysis can help administrator.
2. Using blocked user agents: Blacklisting user-agents revolves around the idea that every browser, bot and spider that visits server identifies itself with a specific user-agent character string. Thus, user-agents associated with malicious, unfriendly or otherwise unwanted behaviour may be identified and blacklisted in order to prevent against future access. Thus, finding malicious user-agents which are blocked through real-time log analysis can help administrator.
3. Identifying and tracing access request to Web server with HTTP request result status code 404: Attempts to access resources not hosted on the Web server are indication of trying to run malicious code on Web server to gain access of the server. Identifying and tracing such events through real-time log analysis can help administrators in redesigning security polices of Web servers.

4. Identifying and tracing access request to Web server with HTTP request result status code 403: Attempts to access prohibited documents on the Web server are indication of trying to run malicious code on Web server to gain unauthorized access of the server. Identifying and tracing such events through real-time log analysis can help administrators in redesigning security polices of Web servers.
5. Identifying access requests from poor reputation IPs: Poor reputation IPs are responsible for sending high level of spam and viruses. To identify and block accesses from such IPs can help administrators in preventing malicious activities and enhance security of IT Infrastructure.
6. Identifying Probing on ports of Perimeter Firewall: The perimeter firewall is first level of defence for organization. Identifying open ports is prior step to perform DOS attacks. Identifying and tracing attempts to gain access into organization network through open ports of SSH and telnet through real-time log analysis can help in enhancing security polices of network.

Result analysis presented through this paper also focuses on real-time performance monitoring of Apache Service of application servers through ELK Stack. Result analysis also discusses performance monitoring of Squid Proxy Service through Metricbeat which is a lightweight shipper that can be installed on servers to periodically collect metrics from the operating system and services running on the server. It also focuses on real-time performance monitoring of network traffic collected through Packetbeat which is a lightweight network packet analyser that sends network log data from NIDS Server to ELK Server. Analysis represented through Fig.4 helps

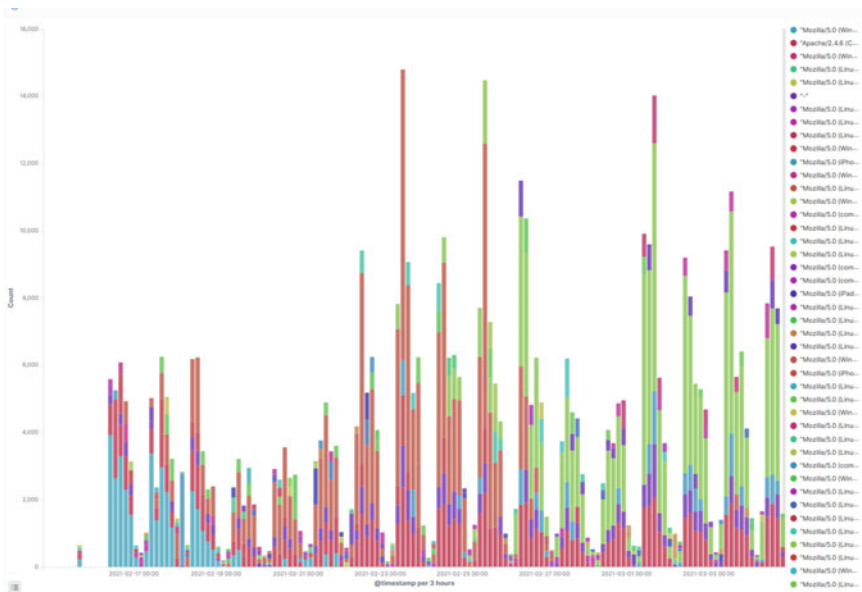


Fig. 4 Analysis of user agent data through batch processing

in identifying user-agents associated with malicious, otherwise unwanted behaviour and blacklisting them such user agents for future access through secure configuration of Apache Service. Analysis also represents the fact that most of user agents that are listed are the user agents white-listed for Apache access through its configuration. It can also help in identifying sources in terms of hosts which are initiating accesses through malicious user agents and categorizing them for further security analysis. Through the analysis depicted through Fig. 5, it seems that Apache Status code 200 was seen most of the time which represents normal working of Apache service on application servers. It also depicts very less percentage of HTTP status codes like 403, 404 which is indication of secure configuration of Web Server. With Apache monitoring, administrators can ensure whether service is configured to sufficiently handle the current scale of access requests. In order to prevent slowdowns which may affect end user experience routine, Apache monitoring can play crucial role in solving performance-related issues in near-real time. Real-time performance monitoring of Apache service deployed on application servers represented through Fig. 6 helps in tracking key Apache monitoring metrics to prevent fatal problems in the scenarios where server load and access requests scale up. With clear and real-time dashboard, represented through [18] Fig. 6 will help administrators to view real-time changes in core metrics and provide stable and efficient server work through proactive actions. Proposed framework also ensures collection, correlation and analysis of squid proxy service logs responsible for providing Internet service over Intranet. Squid logs serve as valuable source of information about Squid workloads and performance. Figure 7

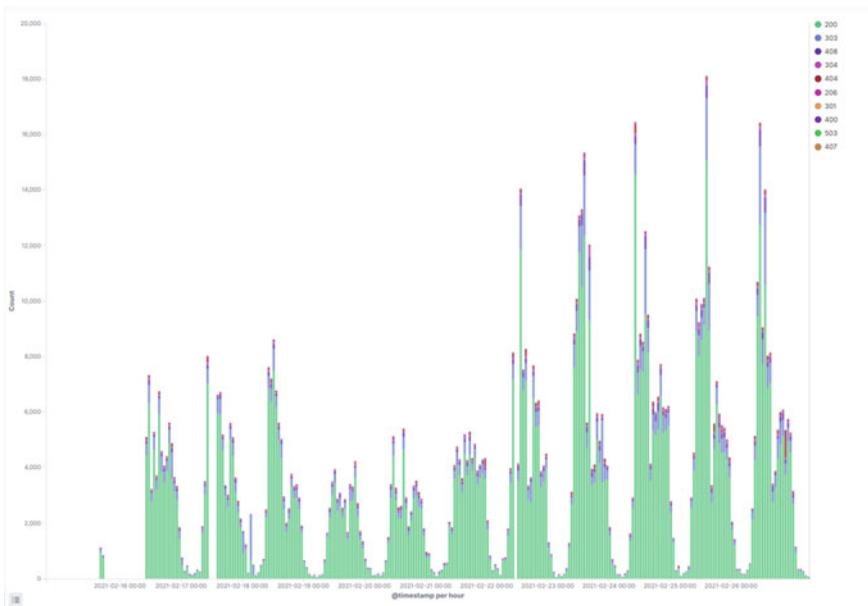


Fig. 5 Analysis of Apache server status code

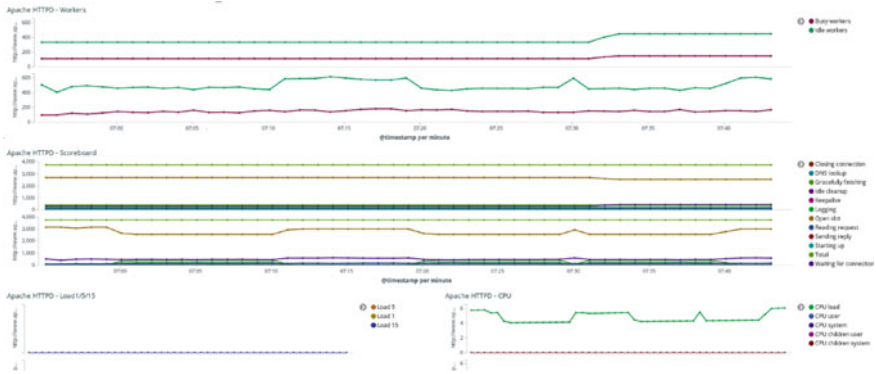


Fig. 6 Real-time performance monitoring of Apache service

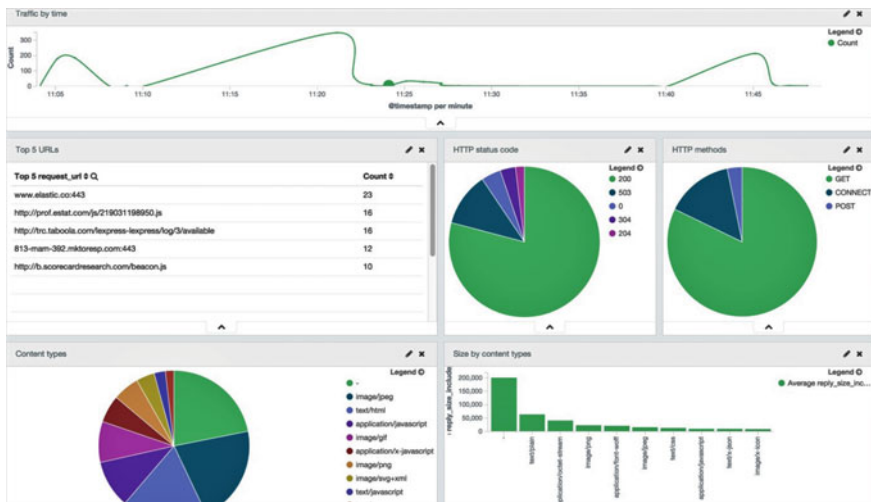


Fig. 7 Real-time performance and analysis of Squid Proxy service

[18] represents not only real-time traffic load on Squid Proxy service but also precisely visualize top users usage analysis, top domains accessed, HTTP methods and status code analysis, content types accessed through in-depth analysis of Squid logs. Application server performance monitoring is really crucial. Using tools like Metricbeat and ELK to monitor, visualise performance metrics makes it convenient to provide stable and error-free working of application servers. Figure 8 [18] represents real-time analysis of application servers in terms infrastructure specific metrics like memory usage, CPU usage, disk usage, network bandwidth and system load through log information shipped through Metricbeat configured on Squid Proxy server.

Packetbeat which is part of the Elastic Stack can be integrated seamlessly with Logstash, Elasticsearch and Kibana in order to transform network data with Logstash,



Fig. 8 Real-time system performance monitoring through Metricbeat

real-time search and analytic in Elasticsearch, review data through precise visualizations in Kibana dashboards [18]. Network performance monitoring can help administrators in ensuring whether network is designed properly and network devices are configured to sufficiently handle the current scale of access. In order to prevent slowdowns in accessing Intranet and Internet contents which may affect end user performing critical tasks, routine real-time network performance monitoring can play crucial role in solving performance related issues in near-real time. Figure 9 [18] depicts real-time Intranet network statistics collected on NIDS server deployed and its analysis by shipping network data collected through Packetbeat to Elasticsearch in terms of precise visualizations through Kibana dashboards. Analysis is represented in terms of connections over time, top Intranet hosts creating and receiving traffic, network traffic statistics between hosts.

IoT devices generate large amounts of logging and events data, monitoring and analysing the same can help both for troubleshooting purposes and as part of predictive maintenance as well. Keeping track of every different activity of IoT devices is impossible without having scalable framework which can address sheer volume of IoT logs. Although the Internet of things (IoT) is getting more and more attraction by researchers, integrating devices and machines to process the data in real time and

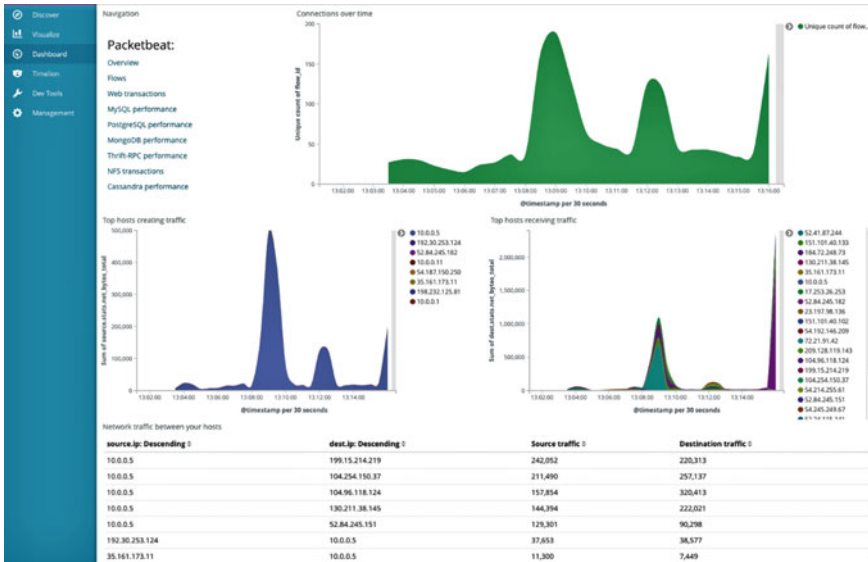


Fig. 9 Real-time network performance monitoring through Packetbeat

at scale is a challenge. To reliably operate with IoT setup, a comprehensive view of the device health in aggregate will prove beneficial.

Graphical analysis represented through Fig. 10 showcases analysis of MQTT Broker successful connection requests received over IoT network discussed earlier through ELK stack. Such analysis can help in tracing vulnerabilities like Ping

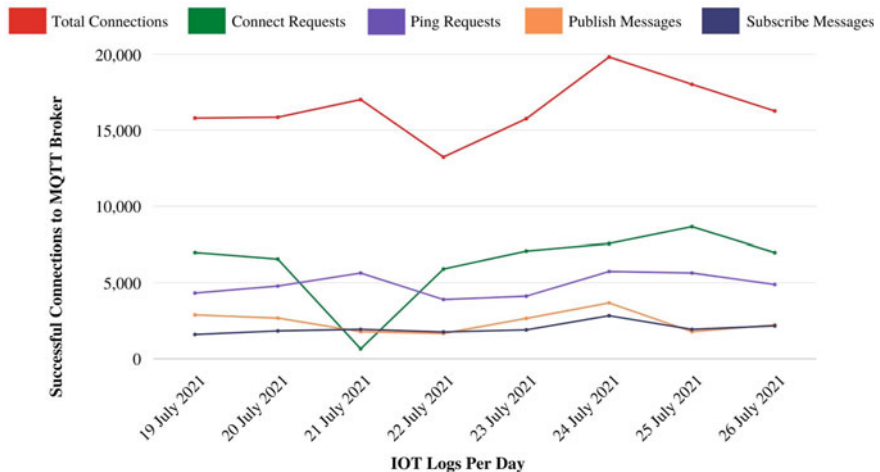


Fig. 10 IoT connection analysis through modelled framework

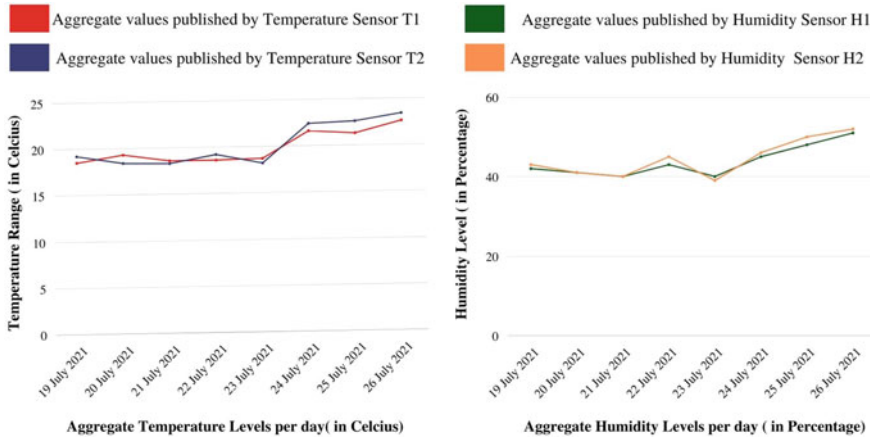


Fig. 11 Aggregate temperature and humidity level analysis through modelled framework

flooding and SYN flooding through which DOS attack is possible. Figure 11 depicts aggregate temperature and humidity level analysis on the values published through IoT network deployed for continuous tracking of temperature and humidity levels required to be maintained in centralized server room which hosts the log analytic setup proposed.

6.3 Performance Evaluation of Framework

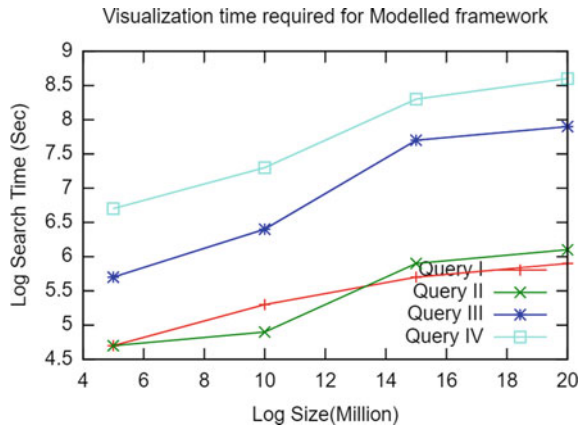
This section discusses performance evaluation of proposed framework in analysing voluminous logs generated through sources discussed in previous subsection with respect to the search queries listed below. In order to compare the performance 5 million Logs (715MB), 10 million logs (1.5 GB), 15 million logs (2.1 GB), 20 million logs (2.7 MB) were used for analysis. Time required to search and analyse specific log data pattern from tagged and classified logs is noted down in following table taking into the size of log volumes mentioned earlier. Proposed framework outperforms for different log volume sizes taking into consideration targeted queries listed below.

1. Query I: HTTP Status Code Analysis to get distinct HTTP Status Codes.
2. Query II: Analysing Frequent Hosts to identify and list hosts with poorly rated IPs from access logs.
3. Query III: Analysing the Top ten hosts who have received 404 and 403 HTTP Response Code over month.
4. Query IV: Analysing time period with maximum 404 and 403 HTTP Response Codes per day over month.

Table 1 Visualization time of modelled framework for queries under consideration

Log size	Query I	Query II	Query III	Query IV
5 million	4.7 s	4.7 s	5.7 s	6.7 s
10million	5.3 s	4.9 s	6.4 s	7.3 s
15million	5.7 s	5.9 s	7.7 s	8.3 s
20million	5.9 s	6.1 s	7.9 s	8.6 s

Fig. 12 Visualization time of modelled framework for queries under consideration



5. Query V: Listing probing events on perimeter firewall with respect to ssh port 22 and telnet port 23. Since both these services are configured on nonstandard ports identifying this probing can give important security related insights.

Table 1 lists time required for graphical visualization of specific log data pattern from tagged and classified log data through Kibana over different log sizes. Figure 10 shows graphical analysis of values listed in Table 1. It is observed that Kibana visualization time required for Query III and IV is significantly more to Query I and II which portrait requirement of implementing scalable cluster for Elasticsearch (Fig. 12).

7 Conclusion

Taking into consideration the speed, nature and volume of logs, real-time monitoring of logs is infeasible to get meaningful insights. The setup collects heterogeneous logs to a central location by using Rsyslog server that ingests logs to Apache Kafka and then to ELK Stack for log analysis and correlation. Modelled framework will be helpful in real-time analysis of heterogeneous logs by combining use of open-source tools at different stages of log analysis so that identification of events that might represent threats can be automated. Modelled framework also ensures forensic investigations of historical logs through Apache Spark-based batch processing

framework implemented. Packetbit and Metricbit are used for ingesting network and system logs of NIDS server and Squid server through which real-time network and system performance monitoring can be done. It helps in giving insights of the logged events through graphical visualization which is easy, quick and efficient way to understand and correlate log events. Modelled framework is a much required attempt to unify stream processing implemented through Apache Kafka, ELK Stack and batch processing implemented through Apache Spark. Modelled framework designed and developed using best open-source tools available will provide cost effective, open-source, enterprise-ready platform and solution for online and offline big data log analytics to provide better insights for faster trouble shooting. It can be widely used across multiple enterprises and domains to identify events that might hamper security of their IT services.

Acknowledgements I would like to express our thanks of gratitude to my Guide Prof. Madhuri Rao for guiding me during this work. Lastly, we would like to thank my Research Center Thadomal Shahani College of Engineering for providing me continuous support whenever required.

References

1. S. Yu, Data processing and development of big data system: a survey, in *Advances in Artificial Intelligence and Security. ICAIS 2021*, ed. by X. Sun, X. Zhang, Z. Xia, E. Bertino. Communications in Computer and Information Science, vol. 1423 (Springer, Cham, 2021), p. 34. <https://doi.org/10.1007/978-3-030-78618-2>
2. M. Harvan, T. Locher, A.C. Sima, Cyclone: unified stream and batch processing, in *2016 45th International Conference on Parallel Processing Workshops (ICPPW)* (2016), pp. 220–229. <https://doi.org/10.1109/ICPPW.2016.42>
3. H. Nasiri, S. Nasehi, M. Goudarzi, Evaluation of distributed stream processing frameworks for IoT applications in Smart Cities. *J. Big Data* **6**, 52 (2019). <https://doi.org/10.1186/s40537-019-0215-2>
4. Z. Lv, H. Song, P. Basanta-Val, A. Steed, M. Jo, Next-generation big data analytics: state of the art, challenges, and future research topics. *IEEE Trans. Ind. Inf.* **13**(4), 1891–1899 (2017). <https://doi.org/10.1109/TII.2017.2650204>
5. H. Hu, Y. Wen, T.-S. Chua, X. Li, Toward scalable systems for big data analytics: a technology tutorial. *IEEE Access* **2**, 652–687 (2014). <https://doi.org/10.1109/ACCESS.2014.2332453>
6. S. Chaudhari, V.K. Maurya, V. Singh, S.S. Tomara, A. Rajana, A. Rawata, Real time logs and traffic monitoring, analysis and visualization setup for IT security enhancement, in *Next Generation Computing Technologies (NGCT-2019)* (2019)
7. Y. Li, Y. Jiang, J. Gu, M. Lu, M. Yu, E.M. Armstrong, T. Huang, D. Moroni, L.J. McGibbney, G. Frank, C. Yang, A cloud-based framework for large-scale log mining through Apache Spark and elasticsearch. *Appl. Sci.* **9**(6) (2019)
8. I. Mavridis, H. Karatza, Performance evaluation of cloud-based log file analysis with Apache Hadoop and Apache Spark. *J. Syst. Softw.* **125**, 133–151 (2017). ISSN 0164-1212. <https://doi.org/10.1016/j.jss.2016.11.037>
9. X. Lin, P. Wang, B. Wu, Log analysis in cloud computing environment with Hadoop and Spark, in *2013 5th IEEE International Conference on Broadband Network and Multimedia Technology* (2013), pp. 273–276. <https://doi.org/10.1109/ICBNMT.2013.6823956>

10. J. Therdphapiyanak, K. Piromsopa, Applying Hadoop for log analysis toward distributed IDS, in *Proceedings of the 7th International Conference on Ubiquitous Information Management and Communication (ICUIMC'13)* (Association for Computing Machinery, New York, NY, USA, 2013), Article 3, pp. 1–6. <https://doi.org/10.1145/2448556.2448559>
11. S. Mehta, P. Kothuri, D.L. Garcia, Anomaly Detection for Network Connection Logs (2018). [arXiv:1812.01941](https://arxiv.org/abs/1812.01941)
12. C. Yang, M. Yu, F. Hu, Y. Jiang, Y. Li, Utilizing cloud computing to address big geospatial data challenges. *Comput. Environ. Urban Syst.* **61**, Part B, 120–128 (2017). ISSN 0198-9715
13. C. Yang, Q. Huang, Z. Li, K. Liu, F. Hu, Big data and cloud computing: innovation opportunities and challenges. *Int. J. Digital Earth* **10**(1), 13–53 (2017). <https://doi.org/10.1080/17538947.2016.1239771>
14. S. Salloum, R. Dautov, X. Chen et al., Big data analytics on Apache Spark. *Int. J. Data Sci. Anal.* **1**, 145–164 (2016). <https://doi.org/10.1007/s41060-016-0027-9>
15. <https://spark.apache.org/>
16. <https://kafka.apache.org/>
17. S. Chhajed, *Learning ELK Stack* (Packt Publishing Ltd., Birmingham, UK, 2015)
18. <https://www.elastic.co/>
19. <https://flume.apache.org/>
20. T. Kolajo, O. Daramola, A. Adebisi, Big data stream analysis: a systematic literature review. *J. Big Data* **6**, 47 (2019). <https://doi.org/10.1186/s40537-019-0210-7>
21. W. Haoxiang, S. Smys, Big data analysis and perturbation using data mining algorithm. *J. Soft Comput. Paradigm (JSCP)* **3**(01), 19–28 (2021)
22. D.D. Mishra, S. Pathan, C. Murthy, Apache Spark based analytics of Squid Proxy Logs, in *IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, vol. 2018 (2018), pp. 1–6. <https://doi.org/10.1109/ANTS.2018.8710044>
23. B.H. Park, S. Hukerikar, R. Adamson, C. Engelmann, Big data meets HPC Log analytics: scalable approach to understanding systems at extreme scale, in *IEEE International Conference on Cluster Computing (CLUSTER)*, vol. 2017 (2017), pp. 758–765. <https://doi.org/10.1109/CLUSTER.2017.113>
24. M. Bajer, Building an IoT data hub with elasticsearch, Logstash and Kibana, in *2017 5th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW)* (2017), pp. 63–68. <https://doi.org/10.1109/FiCloudW.2017.101>
25. I.Y.M. Al-Mahbashi, M.B. Potdar, P. Chauhan, Network security enhancement through effective log analysis using ELK, in *International Conference on Computing Methodologies and Communication (ICCMC)*, vol. 2017 (2017), pp. 566–570. <https://doi.org/10.1109/ICCMC.2017.8282530>
26. J.C. Liu, C.T. Yang, Y.W. Chan et al., Cyberattack detection model using deep learning in a network log system with data visualization. *J. Supercomput.* (2021). <https://doi.org/10.1007/s11227-021-03715-6>
27. L. Chen, J. Liu, M. Xian, H. Wang, Docker container log collection and analysis system based on ELK, in *International Conference on Computer Information and Big Data Applications (CIBDA)*, vol. 2020 (2020), pp. 317–320. <https://doi.org/10.1109/CIBDA50819.2020.00078>
28. S.J. Son, Y. Kwon, Performance of ELK stack and commercial system in security log analysis, in *2017 IEEE 13th Malaysia International Conference on Communications (MICC)* (2017), pp. 187–190. <https://doi.org/10.1109/MICC.2017.8311756>
29. S. Sanjappa, M. Ahmed, Analysis of logs by using Logstash, in *Proceedings of the 5th International Conference on Frontiers in Intelligent Computing: Theory and Applications*, ed. by S. Satapathy, V. Bhateja, S. Udgata, P. Pattnaik. *Advances in Intelligent Systems and Computing*, vol. 516 (Springer, Singapore, 2017). <https://doi.org/10.1007/978-981-10-3156-4>
30. Y.T. Wang, C.T. Yang, E. Kristiani, Y.W. Chan, The implementation of Wi-Fi Log analysis system with ELK Stack, in *Frontier Computing. FC 2018*, ed. by J. Hung, N. Yen, L. Hui. *Lecture Notes in Electrical Engineering*, vol. 542 (Springer, Singapore, 2019). <https://doi.org/10.1007/978-981-13-3648-528>

31. B. Debnath et al., LogLens: a real-time log analysis system, in *2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)* (2018), pp. 1052–1062. <https://doi.org/10.1109/ICDCS.2018.00105>
32. P. He, J. Zhu, S. He, J. Li, M.R. Lyu, Towards automated log parsing for large-scale log data analysis. *IEEE Trans. Dependable Secure Comput.* **15**(6), 931–944 (2018). <https://doi.org/10.1109/TDSC.2017.2762673>
33. R. More, A. Unakal, V. Kulkarni, R.H. Goudar, Real time threat detection system in cloud using big data analytics, in *2017 2nd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology (RTEICT)*, Bangalore (2017), pp. 1262–1264
34. T. Prakash, M. Kakkar, K. Patel, Geo-identification of web users through logs using ELK stack, in *Proceedings of the 2016 6th International Conference Cloud System and Big Data Engineering (Confluence)*, Noida, India, 14–15 Jan 2016, pp. 606–610
35. S. Bagnasco, D. Berzano, A. Guarise, S. Lusso, M. Masera, S. Vallero, Monitoring of IaaS and scientific applications on the cloud using the elasticsearch ecosystem. *Proc. J. Phys.* **608**, 012016 (2015)
36. Y. Li, Y. Jiang, F. Hu, C. Yang, Armstrong, T. Huang, D. Moroni, C. Fench, Leveraging cloud computing to speedup user access log mining, in *Proceedings of the OCEANS 2016 MTS/IEEE Monterey*, Monterey, CA, USA, 19–23 Sept 2016
37. C.T. Yang, E. Kristiani, Y.T. Wang et al., On construction of a network log management system using ELK stack with Ceph. *J. Supercomput.* **76**, 6344–6360 (2020). <https://doi.org/10.1007/s11227-019-02853-2>
38. M. Podhoranyi, A comprehensive social media data processing and analytics architecture by using big data platforms: a case study of twitter flood-risk messages. *Earth Sci. Inform.* **14**, 913–929 (2021). <https://doi.org/10.1007/s12145-021-00601-w>
39. F. Firouzi, B. Farahani, Architecting IoT cloud, in *Intelligent Internet of Things*, ed. by F. Firouzi, K. Chakrabarty, S. Nassif (Springer, Cham, 2020), p. 4. <https://doi.org/10.1007/978-3-030-30367-9>
40. W. Xie, P. Li, H. Xu, Architecture and implementation of real-time analysis system based on cold chain data, in *Complex, Intelligent, and Software Intensive Systems. CISIS 2018*, ed. by L. Barolli, N. Javaid, M. Ikeda, M. Takizawa. *Advances in Intelligent Systems and Computing*, vol. 772 (Springer, Cham, 2018), p. 44. <https://doi.org/10.1007/978-3-319-93659-8>
41. <https://hive.apache.org/>
42. <http://hadoop.apache.org/>

Smart Mirror Information System Using Iot



B. Praveena, K. R. Chairma Lakshmi, S. Vijayalakshmi, and K. Vijay Anand

Abstract Technological advancements motivated to develop smart mirrors designed with Raspberry Pi. This paper focuses on the application of smart mirror in home automation, notice board for displaying the news, schedule for the day, weather updates, and room temperature. Home automation includes controlling of electric appliances by voice control using the microphone fitted with the smart mirror, and remote access is also possible by means of Adafruit cloud and Blynk app. PIR sensor is used, and hence, whenever there is no motion detected near to the mirror, the screen gets turned off and thereby increasing the power saving capability. Screen casting can be done using which YouTube videos can be casted on the mirror. Gas leakage in the room can be monitored by the smart mirror in which the gas sensor has been integrated and buzzer gives an alert whenever it is crossing the threshold value. MySQL is used for storing the sensor data which can be used for future analysis. With the help of an own server-based management program wireless devices, the communication between the microcontrollers was done successfully. Big data analytics is integrated in the proposed methodology, and the Grafana is used for data visualization and connecting IoT devices. In practice, the benefits of IoT were demonstrated to bring down the barriers and create a pathway to the mainstream adaptation of IoT smart devices.

Keywords Raspberry Pi · NodeMCU · Smart mirror · IoT · PIR · Home automation · Temperature and humidity monitoring · Voice control

1 Introduction

Smart products like television, smart watches, etc., are emerging due to the advancements in the technology. The survey says that on an average a men spends a minimum of 18 min looking into the mirror. This time can be productively used if it turns out to be smarter [1]. Smart mirror is a glass made up of 70% reflective and 30% transparent

B. Praveena (✉) · K. R. Chairma Lakshmi · S. Vijayalakshmi · K. Vijay Anand
Department of Electronics and Instrumentation Engineering, R.M.K. Engineering College,
Chennai, India

and hence used it instead of traditional mirror. It functions through NodeMCU which is integrated with the smart technology. Smart mirror functioning through Raspberry Pi foreshows features like weather, temperature, calendars, daily news feeds, to do lists, etc. [2–4]. The proposed technology makes our home or office smarter toward the future development. Smart mirror can be interfaced with both existing and new sensors and products [5]. Philips HomeLab incorporated the interactive mirrors which creates fun in the bathroom by playing cartoons to the children and news to the elders [6]. Home appliances are controlled by voice recognition [7, 8].

Face recognition can be implemented with deep learning algorithms for making the device much more smarter [9]. The server used is free as well as secure, and database is created with SQL. Fog computing proven to be efficient than cloud computing in terms of latency and total network usage [10]. The smart phones can get paired with the smart mirror by means of a Bluetooth facility available in the Raspberry Pi [11]. Akshaya et al. have integrated the two possible ways of accessing the smart mirror by means of Website and mobile application which creates an user-friendly environment [12]. Yusri et al. [13] developed a smart mirror specifically for disabled persons by controlling the home appliances by means of the intelligent mirror. Jin et al. [14] have developed a secure smart mirror which alerts the user when other persons try to access the mirror and biometric authentication is also an added feature. Besserer et al. [15] designed a model which recognizes the user and their emotions and motivates them to do their exercises and shows their happiness level after doing exercise. Rahman et al. [16], Nguyen and Liu [17] proposed a model which recognizes the face and suggests the type of makeup suitable to that particular person. Raspberry Pi enables the smart mirror technology more efficiently [18, 19] and it can be used for edge computing.

2 Materials and Method

2.1 Proposed Methodology

The idea behind our sensible product is to show data like time, date, weather, and an inventory of tasks to be done on a mirror show. This can be the essential data which we want to know during the morning hours to plan for the day. Smart mirror is a technology which will make our lives easier and ease our daily routines. The idea of this product has been around for many years. The inspiration came from the Internet of things (IoT) idea, which can be represented as an associate to form everything sensible. At its core, the Internet of things is regarding connecting devices over the net during a method that permits communication between users and applications on such devices. In the Following, this idea hardware elements that area unit necessary for the product to be purposeful are noninheritable. These include a Raspberry Pi controller board, monitor, and two-way mirror. Instead of third-party APIs an open-source software called Node-RED is used. All the GET and POST methods are managed

using this software. With the help of this software, a process flow has been created. Using which all the client devices, i.e., IoT devices are accessible. Figure 1 depicts the hardware and software integrated with the smart mirror. The home automation performed remotely is represented in Fig. 2.

In existing a single Wi-Fi router used in the home can hold up to 30 devices. When an individual products are added, it will go beyond its limit so we need more routers. Instead the mesh networking concept is used where one NodeMCU will access all the client devices. Here, this NodeMCU is alone connected to the router. This NodeMCU will act as master, around ten slave devices can be connected to this master. In existing products, the system cannot be configured. But in the proposed model, the system can be configured by the clients themselves based on their purpose. For increasing the security of the proposed product, a random IP address will be generated during each time of login. This random IP address can be checked from a remote through an application. This strengthens the privacy of the data of a proposed model. The size of a database in MySQL server is 524272 terabytes, and file size for data is 16

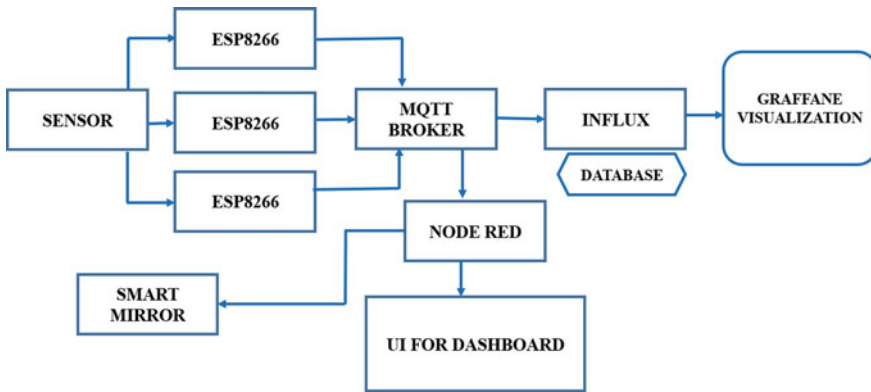


Fig. 1 Block diagram of proposed method

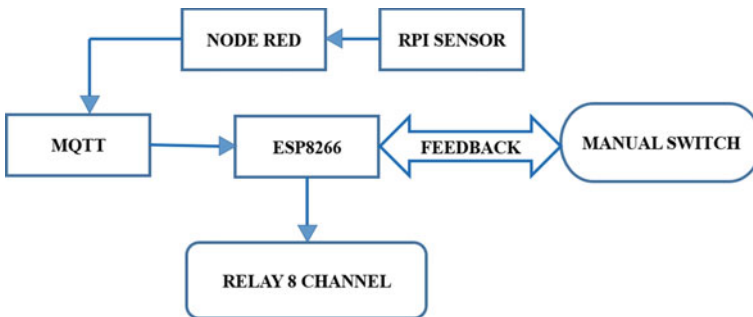
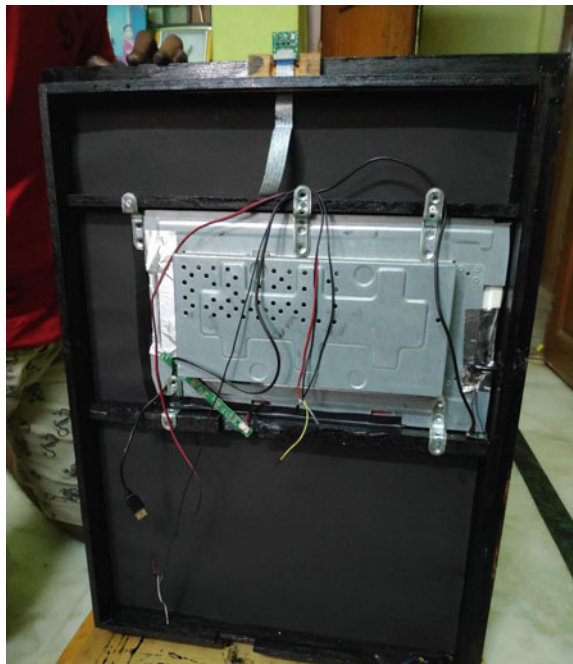


Fig. 2 Block diagram of home automation

terabytes and for the log is 2 terabyte. Hence, huge amount of data can be stored securely.

The main theme of proposed methodology is to create or develop a more sustainable technology in this growing world. The product is out of the box co-engineered with a powerful computer and most advanced IoT devices at a most moderate price. The target audience is people who are planning to construct modern homes with voice assistant combined with artificial intelligence and in industries and colleges as an attractive dashboard which displays all the necessary data which will be a replacement of old style notice boards combined with google calendar, drive access for displaying some pictures, etc. Smart mirror has all necessary ports such as Ethernet port, Wi-Fi, and Bluetooth for connectivity purposes. In order to connect existing appliances with the Internet, an ESP8266 microcontroller is used. In order to increase the efficiency, a variety of sensors and transducers is used so that the efficiency of a technology and home assistant modules is increased. Some of the sensors used in the proposed method are PIR for motion detection for light intensity and DHT11 for temperature and humidity monitoring. Figure 3 is the photographic image of the proposed model.

Fig. 3 Photographic image of the prototype



2.2 Open-Source Software

In the proposed method, the open-source software [OSS] is used. The tools used here are Node-RED, Mosquitto, MQTT, InfluxDB, Grafana, and remote framework.

Node-RED is a programming tool that can be connected with hardware devices through the Internet. It is an open-source tool that was originally developed by IBM technologies. Later, it was developed by JS Foundation. The platform used in this tool is Node JS, and the language used is JavaScript. Node-RED cannot be directly implemented with IoT. It is a generic event processing engine. Without writing any program, it can collect data from Websites like Twitter and Web tools like WebSockets and HTTP and store it in its database. Node-RED provides a Web browser-based flow editor that helps to create JavaScript functions. Instead of low-level coding tasks, Node-RED allows users to connect Web services along with hardware. This can be done through a visual drag-drop interface (Fig. 4).

MQTT means message queuing telemetry transport. It is used to exchange data between the devices and the cloud server. The bandwidth requirements for MQTT are at the absolute minimum, and it has the capability to handle unreliable networks. Hence, it can be used for machine-to-machine [M2M] communication. This protocol usually runs over TCP/IP.

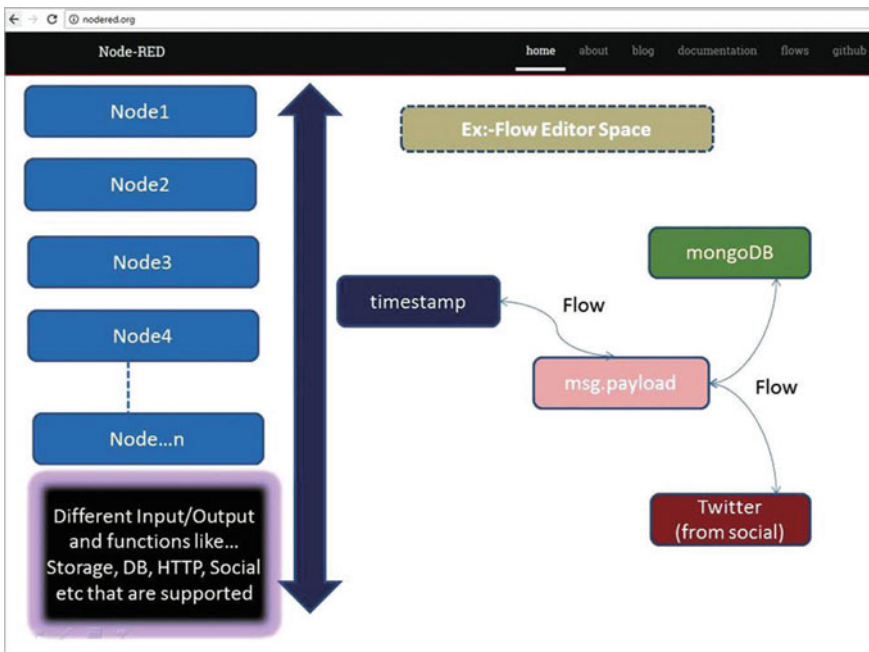


Fig. 4 Node-RED flow

InfluxDB is an open-source time series database that was originally developed by Influx Data. It is optimized for quick, excessive storage, and retrieval of time series data in fields like operations, observations, application metrics, IoT sensing element information, and period analytics. It also has the support for processing data from Grafana. InfluxDB is designed to handle high write and load query.

Grafana is multi-platform open-source analytics and shows an interactive visualization for Web applications. It displays charts, graphs, and gives alerts for the Web when connected to the supported data sources. As a visualization tool, Grafana is a standard element in observing stacks and is usually employed in combination with statistic databases like InfluxDB. Using this tool, the data can be monitored with a customizable dashboard. The main advantage of using Grafana is it will collect data for each and every millisecond, whereas other tools have time intervals. Using Grafana, the data can be monitored even after a week or a month.

3 Algorithm

Step 1: Installing OS in Raspberry Pi.

Step 2: Installing Node-RED in local.

Step 3: An IP address will be generated.

Step 4: There is a port number for each OSS.

Step 5: The port number for each portal is entered in the IP format: Port number.

Step 6: After accessing this port number, all portals will be opened at once.

Step 7: In Node-RED, each node has individual functions. Using these functions, the module can be connected to Google Assistant, UI dashboard, etc.

Step 8: All these nodes are interconnected. During this interconnection, a flow will be generated. In this flow, using JavaScript, a connection is established.

Step 9: Before establishing a connection, MQTT brokers have to be installed. In MQTT, for each device, a device ID is generated. This device ID is entered in Node-RED. For all load devices, an MQTT ID is given. If anyone of the ID pushes the data, that data will be captured by Node-RED. Based on the condition given, a flow will be produced using this captured data.

Step 10: From Node-RED, all data are pushed to InfluxDB. This database will collect each and every data per second. The data present in this InfluxDB will be used by Grafana.

Step 11: Based on these data, Grafana will produce flowcharts and graphs.

Step 12: All the above steps are done on the local server. To establish all these procedures in the cloud, we use a framework called remote framework.

Step 13: The static IP address is generated by InfluxDB and Grafana to Raspberry Pi.

Step 14: This static IP address is merged with the router present in the home.

Step 15: On opening each type of this framework, a dynamic IP is generated.

Step 16: This dynamic IP gets matched to static IP which is connected to the router present in the home.

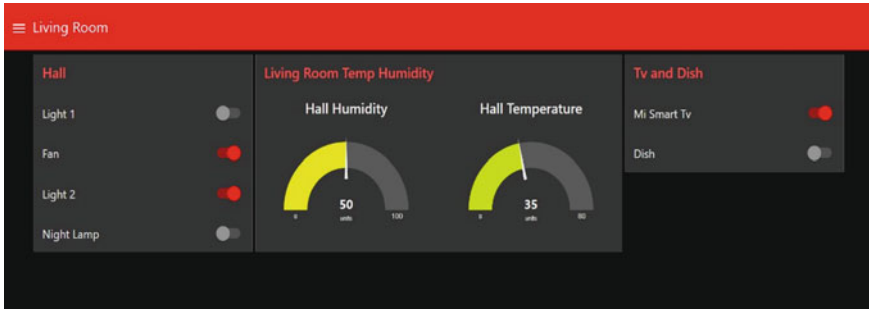


Fig. 5 Temperature and humidity status of living room

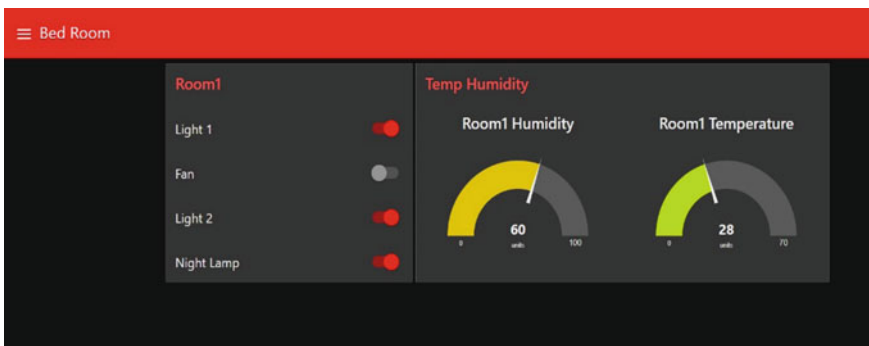


Fig. 6 Temperature and humidity status of bed room

4 Result and Analysis

The status of the temperature and humidity in the living room and bed room is monitored in the dashboard, and it is given in Figs. 5 and 6. The water tank level measurement is monitored remotely, and it is shown in Fig. 7.

The weather update and video demonstration for makeup are displayed in smart mirror, using which a person can do their work in a stage by stage manner and is shown in Fig. 8. The news updated is given in Fig. 9.

5 Conclusion

The smart mirror uses artificial intelligence which plays a major role in showing the user’s notifications and data. A GSM-based automation system is also mannered, according to this technology users can control and monitor the appliances, sensors by Blynk app from the mobile phone. The ultimate goal of Jarvis technology is to

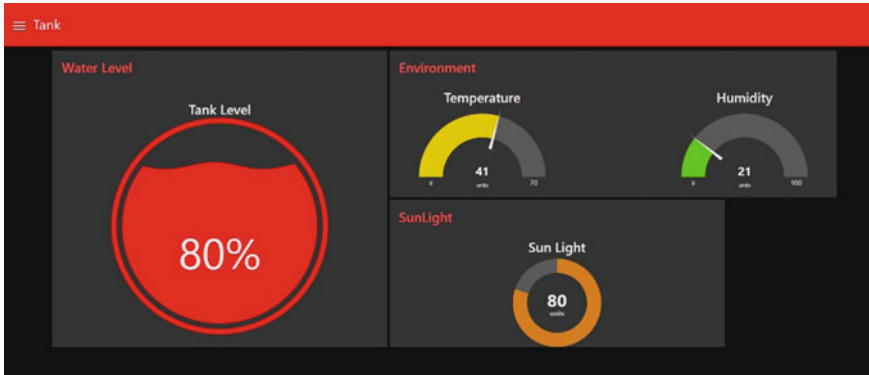


Fig. 7 UI dashboard for tank level measurement



Fig. 8 Smart mirror displaying makeup video

interface all the automated systems into a single interface. This innovation started as a trail to furnish in bringing smart home environment and working place. Ultimately, we all are well pleased with our project. As a further development, extra features can add to this smart mirror, which becomes more customizable and user-friendly.



Fig. 9 Smart mirror update us the news

References

1. A. Johri, S. Jafri, R. N. Wahi, D. Pandey, Smart mirror: a timesaving and affordable assistant, in *2018 4th International Conference on Computing Communication and Automation (ICCCA)* (2018), pp. 1–4
2. Y. Sun, L. Geng, K. Dan, Design of smart mirror based on Raspberry Pi, in *2018 International Conference on Intelligent Transportation Big Data and Smart City (ICITBS)* (2018), pp. 77–80
3. S.C.V.S.L.S. Ravi Kiran, N.B. Kakarla, B.P. Naik, Implementation of home automation system using smart mirror. *Int. J. Innovative Res. Comput. Commun. Eng.* **6**(3) (2018)
4. K. Jin, X. Deng, Z. Huang, S.C. Chen, Design of smart mirror based on Raspberry Pi, in *2018 2nd IEEE Advanced Information Management Communicates Electronic and Automation Control Conference* (2018)
5. P. Henriquez, B.J. Matuszewski, Y. Andreu-Cabedo, L. Bastiani, S. Colantonio, G. Coppini, M. D’Acunto, R. Favilla, D. Germanese, D. Giorgi et al., Mirror mirror on the wall ... an unobtrusive intelligent multisensory mirror for well-being status self-assessment and visualization. *IEEE Trans. Multimedia* **19**(7), 1467–1481 (2017)
6. J.A. Pateljayshri, T. Sadgir Sonal, D. Sangaleharshada, A. Dokhale, A review paper design and development of a smart mirror using Raspberry Pi. *Int. J. Eng. Sci. Invention (IJESI)* **7**(4), 40–43, ISSN (Online): 2319-6734 (2018)
7. B.P. Kulkarni, A.V. Joshi, V.V. Jadhav, A.T. Dhamange, IoT based home automation using Raspberry Pi. *Int. J. Innovative Stud. Sci. Eng. Technol. (IJISSET)* **3**(4) (2017)
8. J. Jose, R. Chakravarthy, J. Jacob, M.M. Ali, S. Maria D’souza, Home automated smart mirror as an internet of things (IoT) implementation—survey paper. *Int. J. Adv. Res. Comput. Commun. Eng.* **6**(2) (2017)
9. Y.B. Hamdan, Smart home environment future challenges and issues—a survey. *J. Electron.* **3**(01), 239–246 (2021)

10. S. Shrestha, S. Shakya, A comparative performance analysis of fog-based smart surveillance system. *J. Trends Comput. Sci. Smart Technol. (TCSST)* **2**(02), 78–88 (2020)
11. M. Ghazal, T. Al Hadithy, Y. Al Khalil, M. Akmal, H. Hajjiab, A mobile-programmable smart mirror for ambient IoT environments, in *2017 5th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW)* (2017), pp. 240–245
12. R. Akshaya, N.N Raj, S. Gowri, Smart mirror-digital magazine for university implemented using Raspberry Pi, in *2018 International Conference on Emerging Trends and Innovations in Engineering and Technological Research (ICETIETR)* (IEEE, Ernakulam, India, 2018), pp. 1–4
13. M.M. Yusri, S. Kasim, R. Hassan, Z. Abdullah, H. Ruslai, K. Jahidin, M.S. Arshad, Smart mirror for smart life, in *2017 6th ICT International Student Project Conference (ICT-ISPC)* (IEEE, Skudai, Malaysia, 2017), pp. 1–5
14. K. Jin, X. Deng, Z. Huang, S. Chen, Design of the smart mirror based on Raspberry Pi, in *2018 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)* (IEEE, Xi'an, China, 2018), pp. 1919–1923
15. D. Besserer, J. Bäurle, A. Nikic, F. Honold, F. Schüssel, M. Weber, Fitmirror: a smart mirror for positive affect in everyday user morning routines, in *Proceedings of the Workshop on Multimodal Analyses enabling Artificial Agents in Human–Machine Interaction* (2016), pp. 48–55
16. A.M. Rahman, T.T. Tran, S.A. Hossain, A. El Saddik, Augmented rendering of makeup features in a smart interactive mirror system for decision support in cosmetic products selection, in *2010 IEEE/ACM 14th International Symposium on Distributed Simulation and Real Time Applications* (2010), pp. 203–206
17. T.V. Nguyen, L. Liu, Smart mirror: intelligent makeup recommendation and synthesis, in *Proceedings of the 25th ACM International Conference on Multimedia* (2017), pp. 1253–1254
18. M.B.N. Siripala, M. Nirosha, P.A.D.A. Jayaweera, N.D.A.S. Dananjaya, S.G.S. Fernando, Raspbian magic mirror—a smart mirror to monitor children by using raspberry pi technology. *Int. J. Sci. Res. Publ.* **7**(12), 335, ISBN 2250-3153-2017 (2017)
19. V. Khanna, Y. Vardhan, D. Nair, P. Pannu, Design and development of a smart mirror using Raspberry Pi. *Int. J. Electr. Electron. Data Commun.* **5**(1) (2017)

A Hybrid Model for Prediction and Progression of COVID-19 Using Clinical Text Data and Chest X-rays



Swetha V. Devan and K. S. Lakshmi

Abstract COVID-19 is an infectious disease caused by a virus known as novel corona virus or severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). Since December 2019, the world is dealing with a pandemic as a result of this sickness. In addition to the medical field, technologies such as deep learning and machine learning aid in the fight against COVID-19. With the use of image or textual data, these technologies can anticipate the existence of disease. The suggested method is a hybrid model that predicts COVID-19 along with illness development using both machine learning and deep learning. Patients' entire medical records (clinical text data) and chest X-rays are regarded significant data for the suggested method. Because this disease mostly affects the respiratory system, X-rays of the chest are utilized to determine how far the sickness has progressed. A logistic regression model is trained with the clinical text data to classify them as COVID or not. Since the classification is binary, logistic regression is an efficient and easy to implement method. Then a VGG-16 model, which is considered as one of the best vision models, is trained by using several chest X-Rays. The trained model is then used to predict the patient's status and the progression of disease. A GUI is also developed for a user-friendly experience so that user can directly input the data. The overall output of the proposed model includes the COVID-19 status, the percentage of progression and the masked image of chest X-ray. The combined classifications using clinical data and chest X-Ray improve the effective disease progression detection of COVID-19.

Keywords COVID-19 · Machine learning · Deep learning · Clinical text · Chest X-Rays

S. V. Devan (✉) · K. S. Lakshmi

Department of Information Technology, Rajagiri School of Engineering and Technology, Ernakulam, Kerala, India

K. S. Lakshmi

e-mail: lakshmiks@rajagiritech.edu.in

1 Introduction

In December 2019, the first case of COVID-19 was reported in China. Since then, the disease has posed numerous hazards and challenges. According to WHO, there are about 1.63 billion reported cases till May 2021 and about 3.3 million deaths were reported. This disease affects various persons in different ways. Some people may not need hospitalization to recover and experience only mild symptoms. Others, on the other hand, have suffered from serious health problems. COVID-19 is characterized by a dry cough, fever and lethargy. In addition, the majority of persons have had mild to moderate respiratory disease [1]. The virus that causes COVID-19 can be detected using the real-time reverse transcription-polymerase chain reaction (RT-PCR) test which is the most preferred method of illness detection. But there are several false negative and false positive results for RT-PCR. It also takes 2–4 h to get the results. Chest X-rays (CXRs) and computed tomography (CT) scans can also be used to screen for COVID-19 infection and evaluate disease progression in hospital admitted cases. Despite the fact that they are not officially suggested [2] as primary diagnostic procedures, they can be utilized as a secondary test to confirm the existence of disease and track disease development.

The application of artificial intelligence to medical diagnostics has a number of benefits for the evolution of the healthcare business. Artificial intelligence-based software [3] can tell if a patient is sick even before symptoms appear. Software's like this can assist doctors in making decisions. These systems work with digital data like texts or photos to deliver results in seconds. Among the textual information is clinical text, which covers the patient's medical data, comprises patient history and assessments, as well as a wealth of information for clinical decision making. Image includes CT scans or X-Rays. Although CT has a higher sensitivity [4] for pulmonary illness, it comes with several drawbacks. It includes the difficulties of sanitizing the room and equipment after each patient, as well as the time it will take to do so after each person. On the other hand, X-Rays of the patients can be collected with less complications since the sanitization of equipment for collecting X-Ray is not a big deal when compared to CT.

The proposed model is a system which uses both machine learning and deep learning technologies for predicting COVID-19 and its progression. The system receives text data as well as chest X-Rays as input and outputs. The logistic regression technique is utilized to handle text data, while the VGG-16 architecture is used to process CXRs. A logistic regression model can effectively handle the binary classification problems. In this case, the logistic regression model will analyse the clinical data and will return yes or no based on the findings. The VGG-16 model is used to detect COVID-19 patients in the case of CXR. The probability value used to calculate the progression is returned by VGG16. The CXR may contain the whole thoracic region and our region of interest is just the lung areas. So, to segment out the lung areas, the lung masks are used. Lung masks are generated by using a pre-trained UNet model. The developed masks are then merged with the X-rays before they are introduced into VGG-16 model. The masked X-rays highlight the lung areas. The

proposed model intends to be a helping hand to medical practitioners to confirm the COVID-19 case and to show how much it has progressed within the lung.

The rest of the sections are organized as follows. Section 2 will give the brief idea about past related works. Section 3 describes about the architecture and techniques involved in the proposed methodology. Then, Sect. 4 deals with results and discussion. Finally, Sect. 5 will conclude the study.

2 Related Works

The literature review reveals more about the previous works in detecting COVID-19 which uses deep learning or machine learning algorithms.

The author of [4] proposed a deep learning model for detecting pulmonary manifestations of COVID-19 with chest X-rays. This method includes a convolutional neural network (CNN) and several pre-trained ImageNet model. The model is trained to learn the features while using each ImageNet with the CNN. The learned knowledge is then used to find the relative performance and the best performing models are iteratively pruned. Author collected several CXR datasets and segmentation is performed at the pre-processing stage. The process of segmenting separates the region of interest from the rest of the picture. In [4] for segmenting the lung regions, the author used a pre-trained UNet model. The size of the dataset and its inherent uncertainty, as well as the computing resources required for effective implementation and usage, are two major determinants of this approach's performance.

In [5], the author explains a neural network architecture that can be trained with a small amount of data while still producing radiologically interpretable results in finding COVID-19. The author has used FC-DenseNet103 to segment the lung areas from the Chest X-Rays. Then a patch-by-patch training approach is used for classifying the CXR as normal, bacterial pneumonia, tuberculosis (TB) or viral pneumonia which includes the pneumonia caused by COVID-19 infection. The segmented images were cropped randomly with a size of 224×224 in the classification network, and the resulting patches were used as network inputs. The centres of patches were randomly selected within the lung areas to avoid cropping the patch from the empty area of the segmented image. Several patches were randomly acquired during the inference, with the number of patches chosen to cover all lung pixels several times, allowing each image to represent the entire attribute of the entire image. The patches were then fed into the network to generate the required output. The classification algorithm's backbone is ResNet-18 model. When a model is overly complex for a limited set of data, overfitting may occur. The ResNet architecture would aid in the prevention of overfitting. The performance of this model slightly affected when reducing the patch size, and there was also no benefit with increasing the patch size. So, the patch size must be maintained as 224×224 .

The author of [6] proposed a weakly supervised deep learning framework to detect COVID-19 infected regions fully automatically using chest CT data acquired from multiple centres and multiple scanners. Based on the CT radiological features, the

disease classifies COVID-19 cases from community-acquired pneumonia (CAP) and non-pneumonia (NP) scans using the developed deep neural networks. The author used TCIA dataset for the proposed model. They trained a multi-view UNet model for the segmentation task. For the classification, they developed a network which was inspired by VGG-16 architecture. In the architecture, configuration of CNN depth increased using small convolution filters stacked with non-linearity injected in between them. All convolution layers consisted of 3×3 kernels, batch normalization and rectified linear units. The proposed CNN was fully convolutional, consisting of five convolutional blocks. Then, a multi-scale learning scheme is adapted to cope with variations of the size and location of the lesions. To implement this, the intermediate CNN representations, i.e. feature maps, at third, fourth and fifth convolution layers were fed into the weakly supervised classification layers. A 1×1 convolution was applied to mapping the feature maps down to the class score maps. Though this model is not discriminative enough when it comes to separate the community-acquired pneumonia from COVID-19, this model can pinpoint the regions of inflammation or lesions within the lung effectively.

Babukarthik et al. [7] is about the COVID-19 detection from CT scans and CXRs. The author of [7] proposed a model which consist of CNN models such as VGG16, ResNet50, DenseNet121, InceptionResNetV2 and several machine learning methods. The CNNs are used for extracting features from the CXRs and CT scans. Then, COVID-19 is identified from the extracted features by using various machine learning algorithms and statistical modelling techniques. In the feature extraction phase, each model is implemented in a hierarchical fashion so that to ensure obtained features are finely refined. The classification logic of the proposed model uses several algorithms such as k-nearest neighbors (kNNs), support vector machine (SVM), Gaussian process (GP), random forest (RF), multilayer perceptron (NN) and Adaboost to process the features. Each algorithm is implemented for different purposes. This method is too complex since it incorporates multiple deep learning and machine learning algorithms.

Shamsi et al. [3] explains a model which helps in classifying the lungs as COVID-19 affected and healthy lungs (normal person) using CXR images. This method proposes an independent and continuous learning algorithm for generating a DCNN architecture spontaneously. The process includes the operations of partitioning DCNN into numerous weighted fully connected and meta-convolutional block. Each block possesses the operations like pooling, convolution, batch normalization, dropout, fully connection and activation operation. The genetic operations such as selection, crossover and mutation process are performed to evolve the population for DCNN architectures. The fitness value will be generated after these processes which will be the prediction results. The model is trained by using about 5000 CXRs. Due to storing and evaluating a huge amount of DCNN structure, GDCNN has high computation and space complexity.

3 Methodology

The proposed methodology consists of two phases such as text processing and image processing. The first phase is detecting COVID-19 from clinical text data using logistic regression model. Second phase is finding COVID-19 from the chest X-Rays and there by calculating the progression, by using pre-trained ImageNet model. The backbone of classification architecture is VGG-16 model. The system architecture is shown in Fig. 1. The proposed methodology consists of the following steps within the upcoming sections. Section 3.1 describes about the datasets that are used; Sect. 3.2 describes about the procedures involved in text data pre-processing. Section 3.3 is the machine learning classification; Sect. 3.4 describes about pre-processing of images; and finally, Sect. 3.5 is the section which deals with the classification of CXRs and progression detection.

3.1 Data Collection

The proposed system requires two types of data, clinical text and chest X-Rays. The clinical text data [8] can be collected from Kaggle repository. The available dataset included 1057 entries of details of patients having symptoms of COVID-19 or other viral diseases. The dataset consists of several attributes including the patient’s gender, age, label which specifies the disease, other details of symptoms and tests of the patients.

For the image processing part, multiple CXR datasets were used which were also collected from Kaggle repository; this includes COVID-19 chest X-Ray Data and CoronaHack—Chest X-Ray dataset [9]. These datasets include chest X-Rays of normal people, COVID-19 patients and people with viral pneumonia. Then, lung segmentation from chest X-Ray dataset [10] is also used which consist of CXRs of

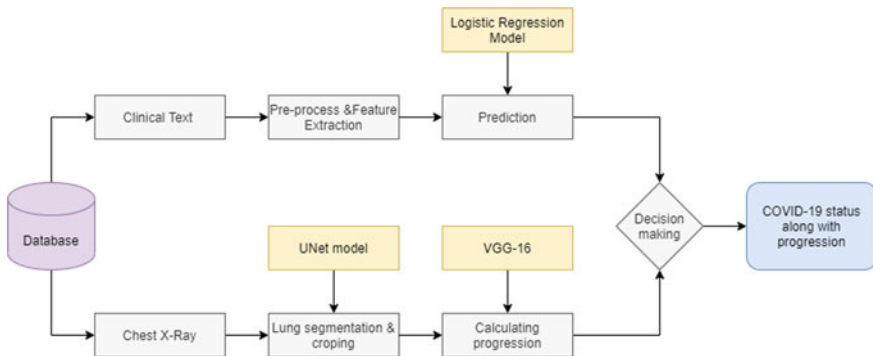


Fig. 1 Architecture of the proposed methodology

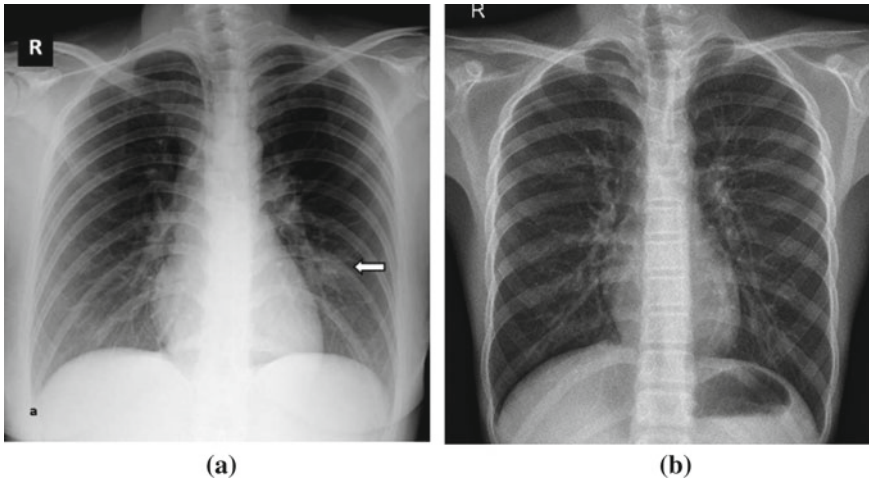


Fig. 2 Images from CXR dataset **a** COVID-19 X-Ray. **b** Normal X-Ray

normal people and CXRs showing pneumonia and non-pneumonic related abnormalities [11]. In Fig. 2a is the X-Ray of COVID patient and b is that of a normal patient.

3.2 Clinical Text Pre-processing

The first step is to prepare the clinical texts to perform machine learning algorithm [3]. The raw data were subjected for several pre-processing methods. Data cleaning is done to remove unwanted texts which includes special characters, white spaces, etc. Then, from the pre-processed data the features can be extracted. For feature extraction, the TF-IDF technique is used. The TF-IDF [12] stands for **term frequency-inverse document frequency**. The TF-IDF technique will identify the relevant features and will convert into the vectorized form. The TF-IDF technique helps in identifying how important a particular word or phrase is to a given document in the process of feature extraction. The basic working of this technique is based on two statistical concepts such as term frequency and inverse term frequency. The term frequency refers to the number of times a term t appears in the document. Then, inverse document frequency will measure the importance or relevance of a particular word in the overall document. TF-IDF value can be simply computed by multiplying both or can be found out by using Eq. (1).

$$W_{i,j} = tf_{i,j} \times \log \frac{N}{df_i} \quad (1)$$

where $tf_{i,j}$ referred to the number of occurrence of term i in document j , df_i is the number of documents containing term i and N is the total number of documents. Here, each record will be considered as different documents. Then, by applying weight function to the extracted features, the vectorized input will be given into the logistic regression model.

3.3 Clinical Text Classification

For classifying the clinical data as COVID positive or negative, a classification algorithm is required. Since the data is text data, it can be classified by using a machine learning algorithm. Here, the classification is a binary classification, so the logistic regression [13] is one of the best candidates for this job. The supervised machine learning technique logistic regression can accurately predict the tags for a binary classification task. The algorithm will return 1 if the test case is of a COVID-19 patient or else it will return 0. The class membership probability can be calculated by the Eq. (2). In which P stands for probability which can have values from 0 to 1; a and b are independent variables, which will vary according with the extracted features.

$$p = \frac{e^{a+bx}}{1 + e^{a+bx}} \quad (2)$$

Primarily, our main focus is on the clinical notes section of the data set which consist of description about the character of patient regarding the symptoms or likely causes, etc. Basically, the primary observations about the patient is included in the clinical notes. Then, there is a column named findings which consist of class label. If the finding is COVID, it will be set as 1 and all other findings such that other viral diseases are set as 0. Logistic regression is used for classification [14]. In the training phase, the logistic regression model is set to train with the vectorized tokens and the findings which will be either 0 or 1 so that the machine can learn what to return when a text is given. In the testing phase, the pre-processed vectorized clinical texts can be classified by the trained model.

3.4 Chest X-Ray Pre-processing

The second phase of the proposed methodology is to identify the CXRs with COVID-19 and to compute the progression. Before implementing the classification algorithm, the raw X-Ray images need to be pre-processed [4]. The fully connected layers in convolutional neural networks, for example, demanded that all images be of the same size arrays. Image pre-processing can also speed up model inference and reduce

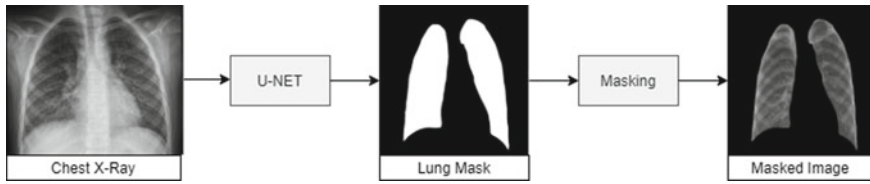


Fig. 3 The CXR segmentation

model training time. If the input photographs are very huge, shrinking them will significantly reduce model training time without compromising model performance.

One of the most important pre-processing is the image segmentation. This is highly applicable in case of using deep learning techniques in medical imaging processes. Separating the foreground from the background, or clustering pixels based on colour or shape similarity are examples of image segmentation. Here, image segmentation is performed on the CXR data sets to visualize the required regions of the data. For this purpose, lung masks will be generated. A pre-trained UNet model is used for segmenting lung areas. The dataset in [10] is a collection of CXRs along with lung masks, so this dataset is used to retrain the UNet architecture so that it could return masks of size 512×512 pixels. These masks are then placed on the CXR images to create a bounding box that contains the lung pixels. Figure 3 shows the mask generation process.

After segmentation the image, further pre-processing techniques includes pixel rescaling and edge preservation are done to maintain the picture quality. The masked images are given into the classification unit for further proceedings.

3.5 CXR Classification

The masked CXR images are fed into the classification network along with the label and progression. The classification network is a VGG-16 model. The VGG-16 [15] model is one of the best architectures in classifying images. There are only two class labels here, Yes and No. Yes, if COVID-19 is identified, else No. A data frame has been created which consist of the class label and progression for each image in training set. This data frame is used to train the network. The network takes images of size 256×256 . So, the segmented CXR are reshaped into 256×256 pixels. The number of neurons in hidden layers is set in to 512. The classification logic is based on the two activation functions, rectified linear activation function (ReLU) and sigmoid function. ReLU activation is applied to the hidden layers, and Softmax activation is applied to the output layer. The ReLU activation will output the value if it is positive, and output zero if the value is negative. Whereas sigmoid activation function will always keep the values between 0 and 1. The sigmoid function can be calculated by the Eq. 3.

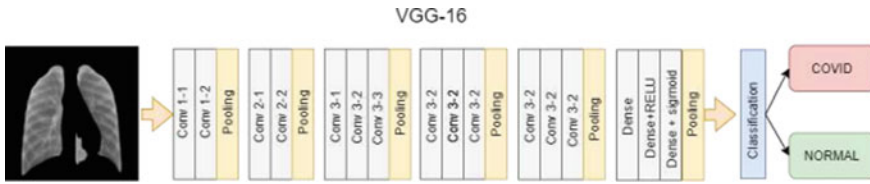


Fig. 4 Classification using VGG-16

$$f(x) = \frac{1}{1 + e^{-(x)}} \tag{3}$$

After setting the values and activation function, the model is trained with the segmented images and the model will be saved. The VGG-6 architecture is shown in Fig. 4. In the architecture, Maxpool layer of 2×2 filter of stride 2 and convolution layers of 3×3 filter with stride 1 and always used identical padding. Throughout the architecture, the convolution and max pool layers are arranged in the same way. It has two FC (completely connected layers) in the end, followed by a Softmax for output. The 16 in VGG16 alludes to the fact that it contains 16 layers with different weights. In the testing phase, the saved model can be used for classifying the CXR. If the CXR is identified as of a COVID-19 patient, then the percentage of progression will be returned.

4 Results and Discussion

As the final step, a graphical user interface has been built with which can join the two phases as well as a user-friendly approach. In the GUI, we can upload clinical texts and once the COVID is detected from text, we can upload CXR. By uploading these data there, we can see the steps by step processes. The output of the model includes the masked image, the two messages such that findings from both text and CXR, then the percentage value of progression which shows how severe is the case. Figures 5 and 6 show the screenshot from the GUI. The first page in the application is a login window where the user can enter userID and password. The login credentials will be checked in the background and if it is matched, the home page will appear. In the home page, there provided three buttons “upload image”, “Start Processing” and “Cancel”. The user has to upload the X-Ray of the patient and the clinical text data, which will be a clinical note consisting the details, symptoms etc. When the processing begins, the system will call the past modules and step by step process will be carried out. The steps will be listed in the window itself. After the processing, the output will be printed.

The proposed system is developed in a windows system having a 4 GB RAM and 2.20 GHz processor. Several tools and libraries are used throughout the system. The text processing phase is implemented with the help of the natural language toolkit

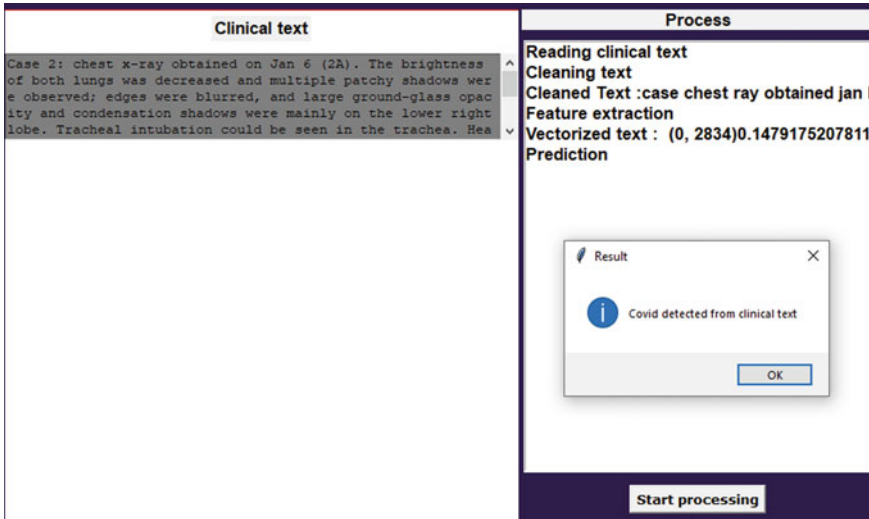


Fig. 5 Confirmation from clinical text

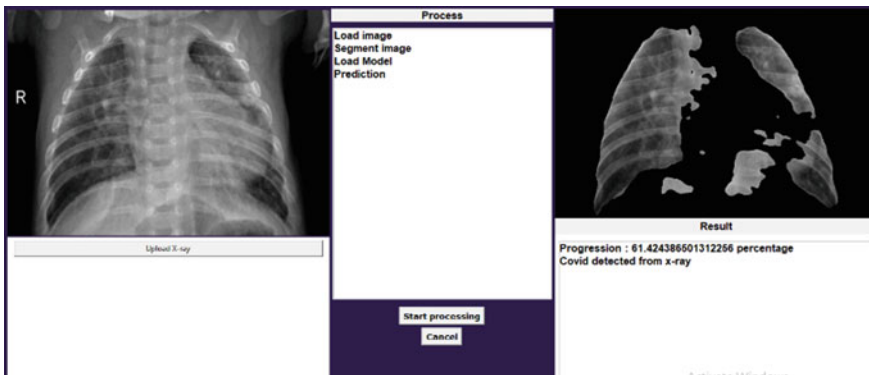


Fig. 6 Progression of COVID-19

(NLTK) which consists a group of libraries and packages that can help in processing natural language. In the second phase, Keras library functions are used which is a well-known deep learning framework. At the end, a GUI has been developed with the help of Tkinter toolkit. The required dataset includes both text and image data. The text data is split into a 70:30 ratio, with 70% of the data being used to train the model, whereas the remaining 30% being used to test the model. The clinical data for COVID-19 is less available. There is a probability for improving the performance if more data is available [16]. The system can be updated according to the availability of more data. Even though this system can aid support in analysing clinical data. For

Table 1 Accuracy

	Phase 1	Phase 2
Accuracy	91	97.7%

X-Ray classification, the text to train ratio is as 80:20, i.e. 80% of data were used for training and 20% being used for testing.

The accuracy of the model can be calculated by the Eq. (4). In the proposed model, we use two algorithms such as logistic regression and VGG-16; the accuracy in each case is given in the Table 1.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FN} + \text{FP}} \tag{4}$$

TP is the number of positive classes predicted as positive. TN is the number of negative classes predicted as negative. FN is the number of positive classes predicted as negative. FP is the number of negative classes predicted as positive.

The phase 1 of the model has been tested in two steps to determine its true accuracy. It employed 75% of the available data that were taken manually in the first stage, which results in lower accuracy than the stage where the entire data were used for experimentation. As a result, it can deduce that if more data is provided to these algorithms, performance may improve. The graph plotted with the obtained value is shown in Fig. 7.

Then in phase 2, the epoch value is given as 7 since the data set is images; in this case, the size of the data set is greater and the time taken for training is more. The least number of topically relevant images are used though; it is recognized that the training time and memory limits are required for practical deployment using computer resources. The model achieved 97.2% precision and *F1* score is 97%. The graph plotted with the obtained value is shown in Fig. 8.

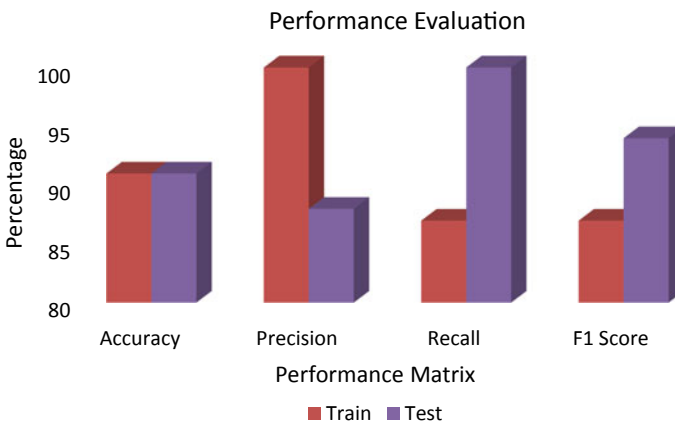


Fig. 7 Graph of phase 1

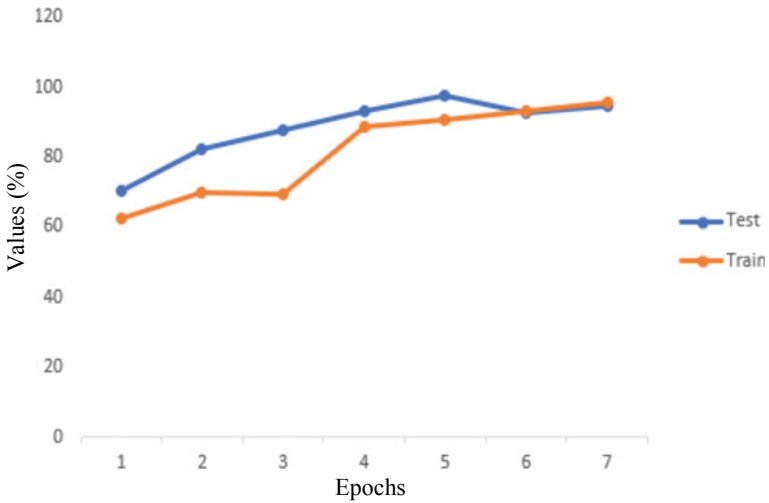


Fig. 8 Graph showing accuracy of phase 2

Table 2 Comparative study

	Type of data	Architectures used	Accuracy obtained (%)
Proposed model	<ul style="list-style-type: none"> Clinical text CXR 	<ul style="list-style-type: none"> Logistic regression UNet VGG-16 	95
[4]	<ul style="list-style-type: none"> CXR 	<ul style="list-style-type: none"> UNet VGG-16 VGG19 InceptionV3 	90.01
[5]	<ul style="list-style-type: none"> CXR 	<ul style="list-style-type: none"> FC DenseNet103 ResNet-18 	91.9
[6]	<ul style="list-style-type: none"> CT scans 	<ul style="list-style-type: none"> Multiview UNet CNN 	89.2
[7]	<ul style="list-style-type: none"> CXR 	<ul style="list-style-type: none"> CNN 	94.84

A comparative study can be done with other similar works Table 2 shows the comparative analysis of the proposed model with other works.

5 Conclusion and Future Scope

The COVID-19 disease had a negative impact on almost every industry. Not only the ones infected, but also all the people had to face a lot of difficulties due to lockdown and all. Many researches are carrying out in defending against this disease. Here, a system is proposed to identify the disease and progression of COVID-19 disease. The system uses clinical texts as well as chest X-Rays for finding the disease. A dataset consists of about 1057 entries of clinical texts and a group of chest X-ray datasets which contain X-Rays of normal lungs and the lungs showing abnormalities such as pneumonic related or COVID-19-related abnormalities. These were used to train the proposed system. While testing the system effectively identifies and returns the image as either normal or as COVID-19 along with the progression. Machine learning algorithm is applied for classifying clinical text data and deep learning algorithm in CXR processing. The system resulted in an overall accuracy of 95%. Increasing the quantity and quality of data can improve the efficiency of the model. The dataset for clinical text is a growing data repository, so the dataset can be updated eventually according with the availability of data. For CXRs, more X-Ray images can be included for training the VGG-16 architecture. In the future, we would like to add a gradient feature and heat maps so that we can pinpoint and visualize the disease-affected regions within the X-Ray.

References

1. World Health Organization Corona Virus Informations [Online]. <https://www.who.int/emergencies/diseases/novel-coronavirus-2019/>
2. COVID-19 and imaging: an article on the limited role for CT and CXR in diagnosis of COVID-19 [Online]. Available: <https://blog.radiology.virginia.edu/covid-19-and-imaging>
3. A. Shamsi, H. Asgharmezahad, S.S. Jokandan, A. Khosravi, P.M. Kebria, D. Nahavandi, S. Nahavandi, D. Srinivasan, An uncertainty—aware transfer learning-based framework for COVID-19 diagnosis. *IEEE Trans. Neural Networks Learn. Syst.* **32** (2021)
4. S. Rajaraman, J. Siegelman, P.O. Alderson, L.S. Folio, L.R. Folio, S.K. Antani, Iteratively pruned deep learning ensembles for COVID-19 detection in chest X-rays. *IEEE Access* **8** (2020)
5. Y. Oh, S. Park, J. Chul Ye, Deep learning COVID-19 features on CXR using limited training data sets. *IEEE Trans. Med. Imaging* **39** (2020)
6. S. Hu, Y. Gao, Z. Niu, Y. Jiang, L. Li, X. Xiao, M. Wang, E.F. Fang, W. Menpes-Smith, J. Xia, H. Ye, G. Yang, Weakly supervised deep learning for COVID-19 infection detection and classification from CT image. *IEEE Access* **8** (2020)
7. R.G. Babukarthik, V. Ananth Krishna Adiga, G. Sambasivam, D. Chandramohan, J. Amudhavel, Prediction of COVID-19 using genetic deep learning convolutional neural network (GDCNN). *IEEE Access* (2020)
8. COVID-19 clinical text data [Online]. Available: <https://www.kaggle.com/bachrr/covid-chest-xray/metadata.csv>
9. COVID-19 X-ray dataset—corona hack dataset [Online]. Available: <https://www.kaggle.com/praveengovi/coronahack-chest-xraydataset>
10. Lung segmentation from chest X-ray dataset. Available: <https://www.kaggle.com/nikhilpandey360/lung-segmentation-from-chest-x-ray-dataset>

11. T. Zebin, S. Rezy, COVID-19 detection and disease progression visualization: deep learning on chest X-rays for classification and coarse localization. *Appl. Intell.* **51**, 1010–1021 (2021)
12. Description for TF/IDF technique [Online]. available: <https://www.geeksforgeeks.org/sklearn-feature-extraction-with-tf-idf/>
13. S.S. Aljameel, I.U. Khan, N. Aslam, M. Aljabri, E.S. Alsulmi, Machine learning-based model to predict the disease severity and outcome in COVID-19 patients—5587188 2021/04/20
14. A.M.U.D. Khanday, S.T. Rabani, Q.R. Khan, N. Rouf, M.M.U. Din, Machine learning based approaches for detecting COVID-19 using clinical text data. *Int. J. Inf. Technol.* (2020)
15. VGG-16 architecture explanation [Online]. Available: <https://towardsdatascience.com/step-by-step-vgg16-implementation-in-keras-for-beginners-a833c686ae6c>
16. E. Sogancioglu, E. Çalli, B. Ginneken, K. Leeuwen, K. Murphy, Deep learning for chest X-ray analysis: a survey (2021)

High-Precision Indoor Tracking Using Ultra-Wide Band Devices and Open Standards



K. Deepika and B. Renuka Prasad

Abstract Indoor tracking requires precise localization with the use of short-range radio technology. Tracking the position of humans in an indoor environment is accomplished using Ultra Wide Band communication technology to achieve high accuracy. Ultra Wide Band (UWB) assists in positioning a user in an indoor environment. UWB technology-based devices obtain the position and monitor the movements of a human in an indoor environment. Ultra Wide Band (UWB) technology positions a user with x , y coordinates obtained from timespan and frequency of communication. Positioning with UWB technology is implemented with transit time methodology—Time of Flight (ToF) to measure the running time of light between the tag and anchors. UWB based positioning of an object requires 3 fixed nodes (anchors) to implement the trilateration algorithm. The direct line-of-sight between the tag and nodes is required to achieve high accuracy. UWB technology uses the Deccawave DWM1001C module to identify the location of a user. The system locates the position values of a user in x , y coordinates using UWB technology with the DWM1001C module in Matplotlib and draw the actual location with visualizations using Grafana.

Keywords Grafana · InfluxDB · MQTT · Node-RED · TCP/IP

1 Introduction

Indoor tracking implements location tracking of a user in an indoor environment to achieve real-time position coordinates and represents the current position of a user with indoor positioning technologies. Ultra Wide Band (UWB) technology [13] is IEEE 802.15.4a and IEEE 802.4z compliant suitable for tracking in an indoor environment as the signal frequency penetrates through thin walls. UWB technology [6] positions a user with the Cartesian coordinates system (x , y) obtained from timespan and frequency of communication. The technology tracks using Time Distance of Arrival (TDoA) [20], Time of Flight (ToF) and Two Way Ranging (TWR). The

K. Deepika (✉) · B. Renuka Prasad
Department of MCA, RV College of Engineering, Bengaluru, Karnataka, India
e-mail: deepikak@rvce.edu.in

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
S. Smys et al. (eds.), *Inventive Computation and Information Technologies*, Lecture Notes in Networks and Systems 336, https://doi.org/10.1007/978-981-16-6723-7_48

655

security at the Physical layer is achieved with Distance–Time bounded protocol and latency is less than 1ms to obtain x and y coordinates.

UWB technology uses Deccawave (now Qorva) DWM1001C module to identify the location of a user. The technology helps to determine the actual position of a user and plots the location as x and y coordinates. UWB based indoor positioning system [19] with high precision is achieved with DWM1001C module. DWM1001C module is configured either as an anchor or as a tag. A tag is also configured with RPi Zero with DWM1001C module.

The objectives of Real-Time Indoor Positioning Systems (IPS) with UWB are

- Implement a human positioning system in an indoor environment with DWM1001C module using UWB technology
- Obtain the position in x, y coordinates and communicated to the time-series database in a remote system
- Display the position with the x, y coordinates on a scale map and path movements in an interactive visualization graph.

The DWM1001C module is programmed using Python. The location coordinates are communicated to InfluxDB to store the sensor data using an optimized time-series database (TSDB). The coordinates are visualized using an interactive visualization tool, Grafana to fetch the sensor data from InfluxDB and visualize the location coordinates in interactive dashboards. The anchors and tag communicate using UWB frequency and use Time Division Multiple Access (TDMA) [10]. The anchor is positioned in Line of Sight (LoS) propagation. The real-time system is implemented using UWB technology by configuring the DWM1001C module in two methods

- DWM1001C module with RPi Zero configured as tag and DWM1001C modules as anchors
- DWM1001C module configured as tag and anchors

Indoor Positioning Systems (IPS) is a real-time positioning of a human with Ultra Wide Band (UWB) technology using the devices programmed with DWM1001C module. The module DWM1001C combines DW1000 chipset, nRF52832 Micro-Controller Unit (MCU) and 3-axis accelerometer. The module enables Real-Time Locating Systems (RTLS) [11] integrated with DWM1001C Integrated Chip (IC), antenna and power control. The module supports Time of Flight (ToF) and Time Difference of Arrival (TDoA) location algorithms. The source code editors like Visual Studio Code and visualization Integrated Development Environments (IDE) platforms like Grafana and Jupyter are licensed under Free Software. The high-precision indoor tracking using Ultra-Wide Band devices and open standards aims to build a system using Free Software to provide reliability and privacy to the user [14].

Several research works have been executed in indoor tracking using UWB devices to achieve high precision over the last few decades. The literature review is achieved to overview, outline, analyze and classify the state-of-the-art research in this domain. The contributions of the works implemented using UWB are presented in Sect. 2. The experimental setup of the DWM1001C module with RPi Zero as tag and DWM1001C

module configured as anchors and DWM1001C module as tag and anchors is detailed in Sect. 3. The methodology of the DWM1001C module configured as tag and anchors are elaborated in Sect. 4. In Sect. 5, the methodology of the DWM1001C module is deliberated. Future works and conclusions are given in Sect. 6.

2 Literature Review

The signal technologies for indoor localization means solutions providing the position of mobile objects or people in indoor environments (e.g., hospitals, malls, etc.), is one of the most cutting-edge services with growing demand in smart applications such as robotics for care and pedestrian navigation. The challenge of indoor technology is to position a person in a bounded environment surrounded by obstacles [12]. The experimental analysis of extensively deployed using UWB technology are discussed in Table 1.

The signal technologies for indoor localization means solutions providing the position of mobile objects or people in indoor environments (e.g., hospitals, malls, etc.), is one of the most cutting-edge services with growing demand in smart applications such as robotics for care and pedestrian navigation. The challenge of indoor technology is to position a person in a bounded environment surrounded by obstacles [12]. The experimental analysis of extensively deployed using UWB technology are discussed in Table 1.

Ultra Wide Band (UWB) is a long-range radio technology that communicates using LoS and NLoS propagation. UWB technology supports high bandwidth, multipath technique [7]. The technology provides an accuracy of 1 cm with a frequency range of 70–300m. Positioning in an indoor location is achieved with Two Way

Table 1 Real-time indoor positioning systems with UWB technology

Name	Year	Pulse duration	Accuracy	Principle	Application
Nakano et al. [8]	2018	750 ps	2.1 mm	Distance resolution management	3D positioning
Keefe et al. [9]	2017	200 ps	9.5 cm	AoA, TDoA	Local positioning system
Sato et al. [11]	2016	Psuedo noise	1 m	AoA	Indoor flying robot
Flores et al. [1]	2016	2 ns	10 cm	ToA, AoA	Indoor object positioning
Writrsal et al. [16]	2016	Very short	10 cm	TWR, MIMO, PDoA	Accurate indoor positioning

Range (TWR) [2], Time Difference of Arrival (TDoA) [3], Phase Difference of Arrival (PDoA) algorithms. Smart homes, warehouse management applications are implemented using UWB technology. 3D positioning is achieved by implementing the distance resolution management principle estimating the accuracy of 2.1 mm. Angle of Arrival (AoA) [20] and Time Difference of Arrival (TDoA) [20] algorithms obtain the accuracy of 9.5 cm in local positioning systems. Indoor flying robots are implemented with AoA algorithms obtaining an accuracy of 1m. Indoor positioning is accomplished with ToA [17], AoA, TWR [15], MIMO, PDoA algorithms with the accuracy of 10 cm.

3 Experimental Setup

The setup is configured by installing the anchor devices at the height of 2.5–3 m. A minimum of 3 anchors is required to implement a trilateration algorithm. The devices operate using a repeating superframe structure of 100 ms duration. SuperFrame structure has 30 slots and is numbered as Beacon Message Numbers (BCN) slots. Each anchor is assigned with each slot and is called a seat number and varies from 0 to 29. The first slot is assigned for the initiator. The count of anchors is not limited to 30 and the count can be increased. BCN numbers can be increased accordingly. Each anchor is assigned with BCN slots and a seat number. The additional anchor after BCN 30 will reuse the slots but it should not connect the anchor with the same seat number. In case the same seat number is allotted, the system will disconnect from the network and reconnect to the same network until no conflicts occurs with the seat number.

The tag is connected to the processor which reads the location (x, y) serially and communicates to the time-series database system. Initially, the tag sleeps and periodically wakes up to listen to the beacon of the anchor and Almanacs messages. Anchor listens to the period of 5 SuperFrames before returning to sleep for the specified interval. The anchor will automatically wake up and resume the process again. The sleep period will initially be 10s and will be extended to 60s. The tag is configured with two modes of operation—responsive and low-power mode operations. Responsive mode follows (Two Way Range) TWR exchange for scheduling the listening period. TWR exchange schedules the next listening period to listen to the beacons during the SuperFrame reserved the TWR exchange slot. The DW1000 chipset will remain idle in this period and nRF52832 will be in sleep mode. The low-power mode operation puts the DW1000 chipset in the sleep mode until the following TWR exchange.

In low-power mode, TWR exchange the DW1000 will be put to Deep Sleep and will be sent to a responsive state in the next TWR exchange. The microcontroller will be put to sleep with other components of the module except the Real-Time Clock (RTC) and accelerometer. The module is in the lowest power consumption mode and will not be able to listen to beacons. The module needs to moves out of the area and initiates communication with the anchors in range, the tag will proceed with TWR slot reservation. The tag collects the ranging and data slot maps to show the slot

utilization from the beacon messages of all the anchors in range and combines to select a free-ranging slot in the SuperFrame to assign the anchor in the range. The system faces few technical issues as the ranging slots are occupied in the SuperFrame. Firstly, the system tries to establish communication every 60 s to receive the incoming data and reserve a TWR slot. Every 100 ms, the SuperFrame contains 15 ranging slots which provide sufficient time to the tag to perform TWR with 4 anchors and giving a maximum location rate capacity 150 Hz. Secondly, the system capacity is and new tags will not be able to start ranging until the existing tags move out of the area or give up the slots.

Additional tags can be added to the system to high-precision indoor tracking using UWB devices. The system is designed to have 150 Hz system capacity and system expansion is achieved with the following tag count and location rate.

- 15 tags @ 10 Hz (max. location rate)
- 150 tags @ 1 Hz
- 300 tags @ 0.5 Hz
- 9000 tags @ 0.01667 (min. location rate).

The TWR internal location engine estimates the position of the tag with the known position of the anchors in range. The location estimate is calculated with the three or four ranging results. The location estimate manages one or two missing responses from the anchors and estimates the location of the tag. The position of the tag is estimated with the static position (x, y) . The three-axis accelerometer—STM LIS2DH12TR detects the orientation of the device. The devices operate at low power mode and the supply voltage is about 2.8 to 3.6 V. The technology communicates between 3.5 and 10 GHz frequency and achieves a precision of 10–30 cm. The range of the devices is 250 m² maximum. The data communications are achieved up to 27 Mbps. The system is capable of cluster communication as it communicates 750

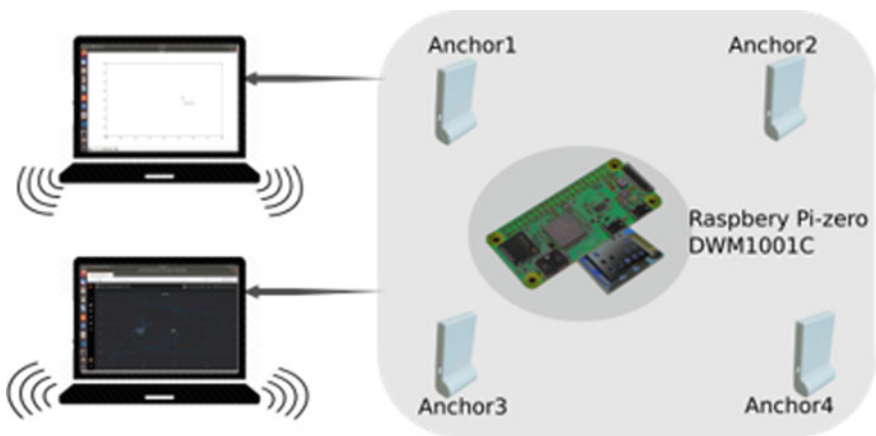


Fig. 1 Experimental setup of real-time DWM1001C module with RPi Zero as tag and DWM1001C as anchors

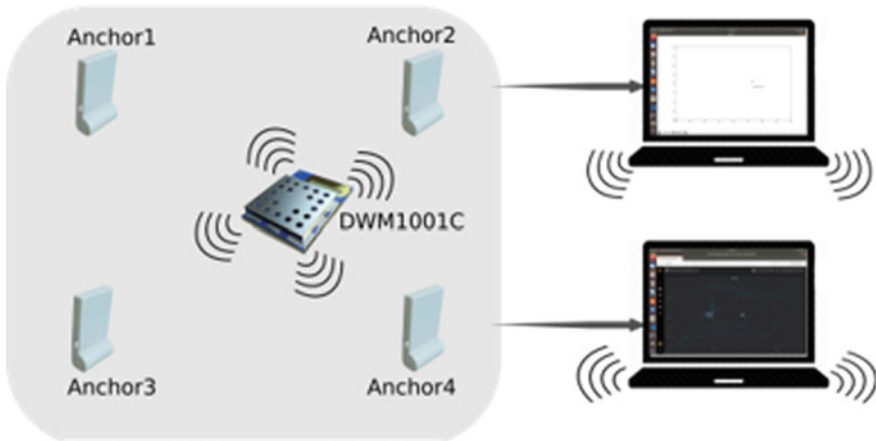


Fig. 2 Experimental setup of real-time DWM1001C module as tag and anchors

tags in 0.2Hz, 150 tags 1 Hz and 15 tags 10Hz. The experimental setup to obtain the location from the DWM1000 chipset configured with RPi Zero is shown in Fig. 1 and DWM1001C module configured as tag and anchors is represented in Fig. 2.

4 Methodology of Real-Time DWM1001C Module with RPi Zero as Tag and DWM1001C as Anchors

Real time indoor tracking with UWB using DWM1001C module is a human tracking system implemented for indoor environments. The following assumptions are made in implementing the system. The identity of the person is already known to the system. The timestamp information along with the location is communicated to the system. The location tracking details of a particular user is stored in the database for a maximum period of two days. The position is obtained in x and y coordinates. It is assumed that the user carries the device at all times. The user is located in the range of anchors fixed in the indoor area [18]. The methodology of DWM1001C Module with RPi Zero is represented in Fig. 3.

4.1 Step 1—Hardware Circuitry

The hardware setup is made as depicted in Sect. 3 in Fig. 1. The system is setup with DWM1001C enabled with DW1000 chipset, nRF52832 MCU and 3-axis accelerometer device. The anchor is fixed in equal distances and height with the Line of Sight (LoS). The tag is constructed by coupling DWM1001C module with RPi Zero. The Pin diagram of the Raspberry Pi Zero with DW1001C module is given in Fig. 4.

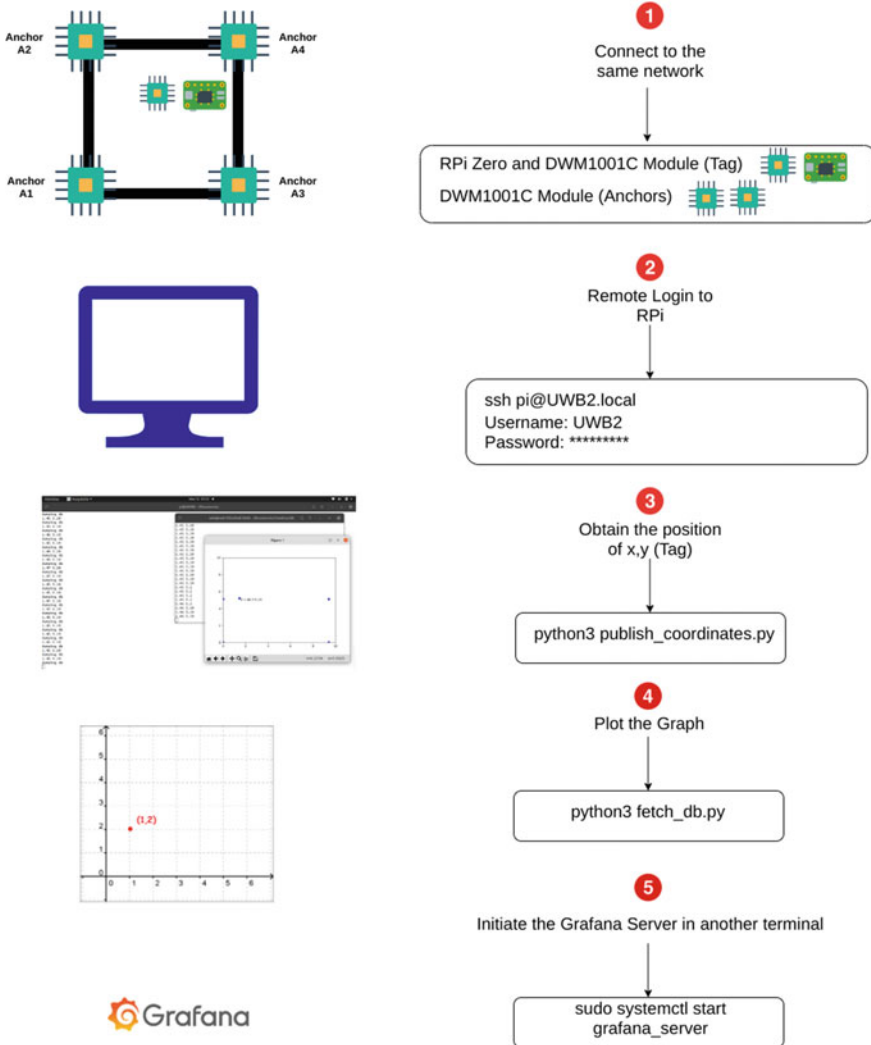


Fig. 3 Methodology of DWM1001C module with RPi Zero

4.2 Step 2—Software Environment

The anchor and DWM1001C module with RPi Zero tag are programmed using Python in Visual Studio Code. The code is divided into three sections—fetch, dump and plot. The fetch section retrieves the coordinates of the tag by calculating the time period of the message transfer and response obtained from the anchor. The dump section obtains the position (x, y) of the tag from the anchor to the remote system. The plot section plots the graph from the obtained coordinates on two types of graphs—

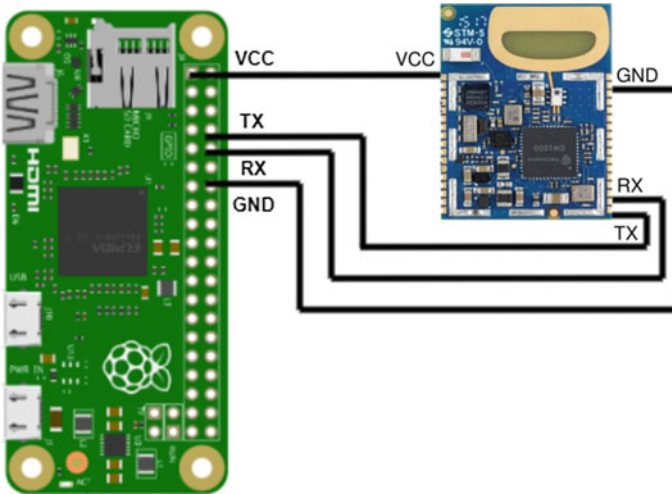


Fig. 4 Pin diagram of RPi Zero with DW1001C module

internal and external graphs. The internal graphs are plotted using matplotlib, Python graph library. The external graphs are represented over the cloud using Grafana, a visualization tool.

4.3 Step 3—Initialization Phase

The anchors have a static position and initialize the communication to the tag. The anchor and tag devices communicate at 3.3GHz. Each anchor positioned in the indoor environment communicates to the tag by sending a message. The tag obtains the message from the nearest anchor faster than the distant anchor. The tag sends a response to every anchor in the order of message obtained and is called a Two Way Range (TWR). The fetch section calculates the position (x, y) with time period and frequency.

4.4 Step 4—Communication Phase

The tag communicates the position coordinates (X, Y) to the dump section of the remote system via local network or by Cloud. The system receives x and y values depicting the position in Universal Asynchronous Receiver Transmitter (UART) mode.

4.5 Step 5—Decoding Phase

The system obtains the position coordinates (x, y) values and calculates the position of the tag with the known static positions of the anchors. The position coordinates are sent to the plot section with the anchor positions.

4.6 Step 6—Positioning Phase

The plot sections plots the graph internally and externally. Matplotlib tool represents the graph marking the anchor points and displaying the tag movement. Grafana visualizes the tag and positions the movement line [16].

5 Methodology of Real-Time DWM1001C Module

DWM1001C module is preloaded with firmware to assist system developers to deploy RTLS systems for required applications at ease. The module is programmed to behave as an anchor as one of the fixed nodes and tag as mobile located nodes. DWM1001C module is configured using a UART connection from an external host to implement indoor tracking. Some assumptions are made for implementing the system to retrieve the position coordinates of the user in a bounded environment. The system assumes that the identity of the person is already known. The timestamp information is retrieved with the position coordinates. The data is stored for a maximum period

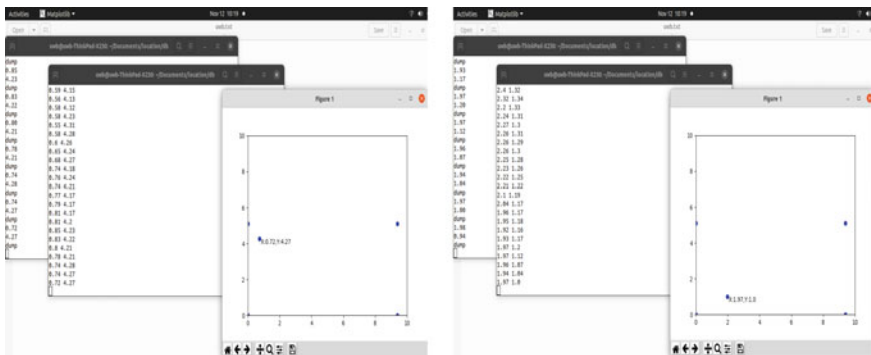


Fig. 5 Using matplotlib to plot the position with DWM1001C module and RPi Zero

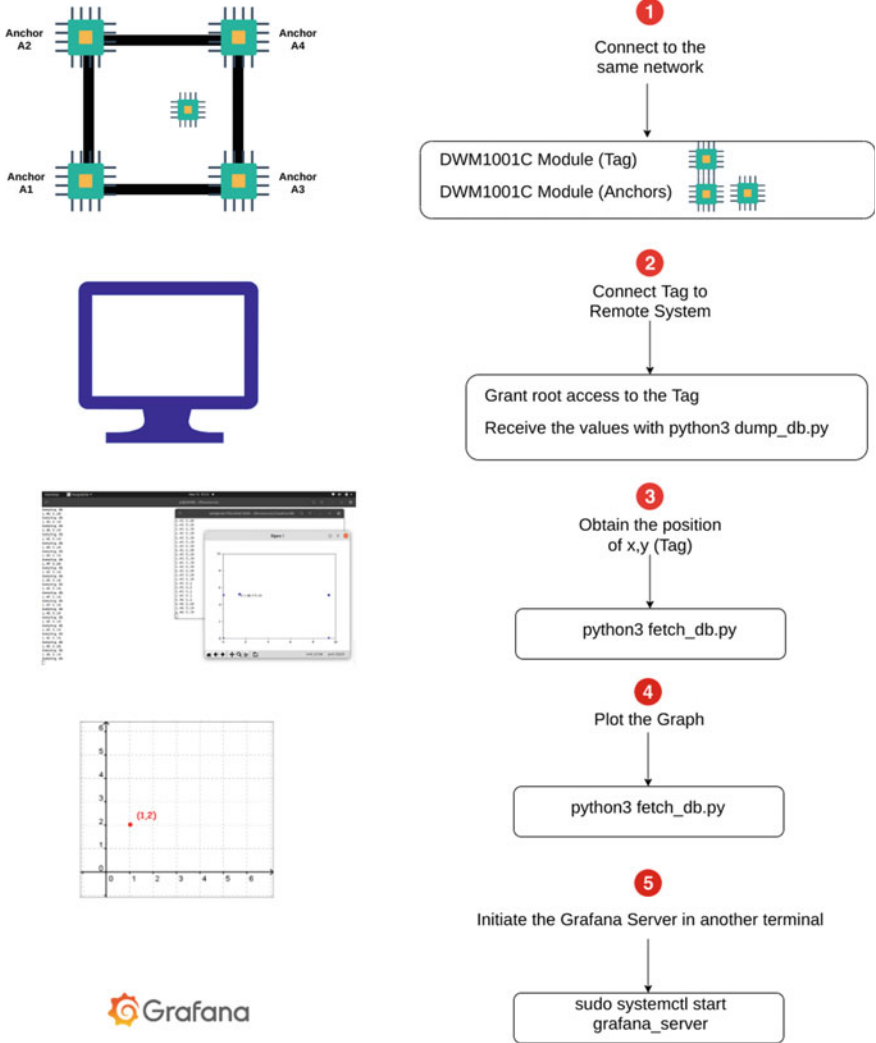


Fig. 6 Methodology of DWM1001C module

of two days. The position is obtained in x and y coordinates (Fig. 5). The device—DWM1001C module is carried by the user at all times and the user is located in the range of anchors in a bounded environment. The methodology of DWM1001C module is represented in Fig. 6.

5.1 Step 1—Hardware Circuitry

The hardware setup of DWM1001C Module programmed as anchors and tags is shown in Sect. 3 in Fig. 2. DWM1001C configured with DW1000 chipset, nRF52832 MCU and 3-axis accelerometer device serve as anchors and tag. The anchor is fixed in equal distances and height with the Line of Sight (LoS).

5.2 Step 2—Software Environment

DWM1001C module anchors and tag are programmed using Python in Visual Studio Code. The code is divided into three sections—fetch, dump and plot. The fetch section fetches the position coordinates by estimating the time period of the message sent to the anchor and the response received. The dump section communicates the position (x, y) of the tag carried by the user to the remote system. The plot section plots the graph with the coordinates communicated by the dump section. The graphs are plotted as internal and external graphs using Matplotlib and Grafana as specified in Sect. 4.2.

5.3 Step 3—Initialization Phase

The anchors are fixed nodes and initiate the communication to the mobile nodes-tags. DWM1001C devices communicate at 3.3 GHz. Anchors initiate the communication with the tag by sending a message. The mobile node-tag receives the message faster from the nearest anchor than the message communicated by the distant anchor. TWR communication is achieved as the tag telecasts a response to every anchor in the range. The fetch section estimates the position (x, y) with the response duration and frequency of communication.

5.4 Step 4—Communication Phase

The coordinates sent by the fetch section are communicated to the dump section in the remote system via a local network or Cloud. The system receives x and y values depicting the position in Universal Asynchronous Receiver Transmitter (UART) mode. DWM1001 module initiates UART Generic mode by default. The shell mode can be switched on by pressing the enter command twice within a second.

5.5 Step 5—Decoding Phase

The system obtains the position coordinates (x, y) values and calculates the position of the tag with the known static positions of the anchors. The position coordinates are sent to the plot section with the anchor positions.

5.6 Step 6—Positioning Phase

The plot sections plots the graph internally and externally. Matplotlib tool represents the graph marking the anchor points and displaying the tag movement. Matplotlib visualizes the location of the user and positions the movements line in Fig. 7. The tag exiting the range of the anchors is shown in Fig. 8. The movement line visualized using Grafana is represented with Fig. 9.

6 Results and Conclusions

High-precision indoor tracking using Ultra-Wide Band devices and open standards is a real-time system deployed in a bounded environment. The real-time location of a human is obtained as Cartesian coordinates with DWM1001C devices. A minimum of four anchors is suggested to be used to position a single tag carried by the human to fetch the coordinates. The methodologies stated for real-time indoor tracking involves minimal infrastructure and achieves high-precision location coordinates. The implementation of DWM1001C Module with RPi Zero as tag and DWM1001C as anchors and DWM1001C as tag and anchors achieve precise coordinates of the user is shown in Figs. 7 and 8.

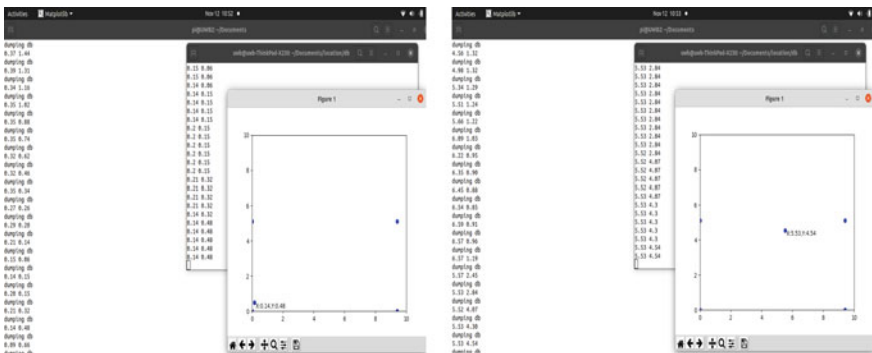


Fig. 7 Using matplotlib to plot the position with DWM1001C module

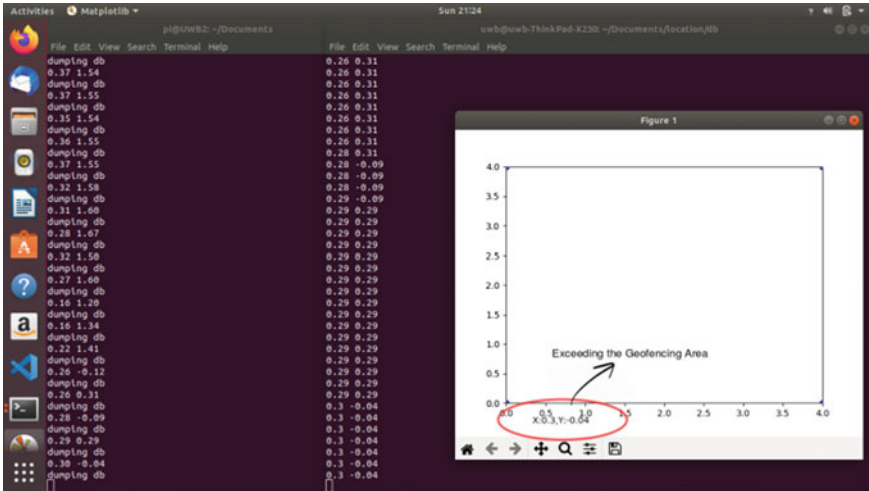


Fig. 8 Signal strength transmitted between the UWB node and anchors



Fig. 9 Visualization graph plotted with the Grafana

DWM1001C Ultra-Wide Band devices are accessed based on accuracy and data rate. The results prove that the accuracy is up to 30cm and the data rate is up to 70Mbps in the indoor environment. UWB devices are programmed with DWM1001C module as 100m distance mapped to 75 Kbps distance. The presence of obstacles and different interfering signals can affect the performance in the bounded environment. The schemes like Chirp Spread Spectrum (CSS) used by UWB are energy efficient, less complex and robust to minimize the adverse effects of multi-

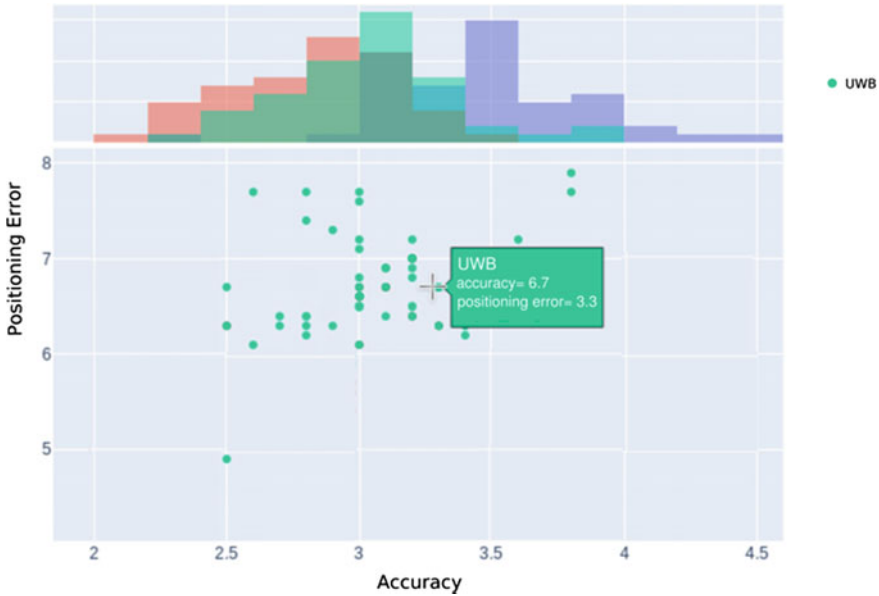


Fig. 10 Accuracy and positioning error of UWB devices

path and noise. The static obstacles such as walls and ceilings have modeled the effect of human presence. Human bodies absorb, reflect, and diffract radio signals, which could affect the value of radio frequency. The offline mapping is performed with no or few people and positioning is performed with many people and is observed that the system could lose reliability. Accuracy and positioning error of UWB is shown in Fig. 10. Distance and displacement between multiple UWB devices is represented in Fig. 11. The signal strength transmitted between UWB devices is depicted in Fig. 12.

The results of past studies have shown that, on average, the presence of human bodies increased the error rate by 11% regardless of the algorithm used [4]. UWB radio provides high-speed data rate communication over the personal area network space. UWB devices transmit extremely short pulses and use techniques that cause a spreading of the radio waves through wide frequency band with very low power, spectral density [5]. The high bandwidth offers high data throughput for communication. The low frequency of UWB pulses enables the signal to effectively pass through obstacles such as walls and objects. In the presence of multipath, given the system bandwidth limitation of 125 kHz, signal paths are often indistinguishable. Only the average channel delay can be estimated. In some cases, the direct signal path is not present, introducing a delay offset into the frame timestamps, as only reflection paths are seen.

UWB devices with DWM1001C module positions using TWR, ToF and TDoA operating at 500MHz frequency. The accuracy of the position is obtained at 10–

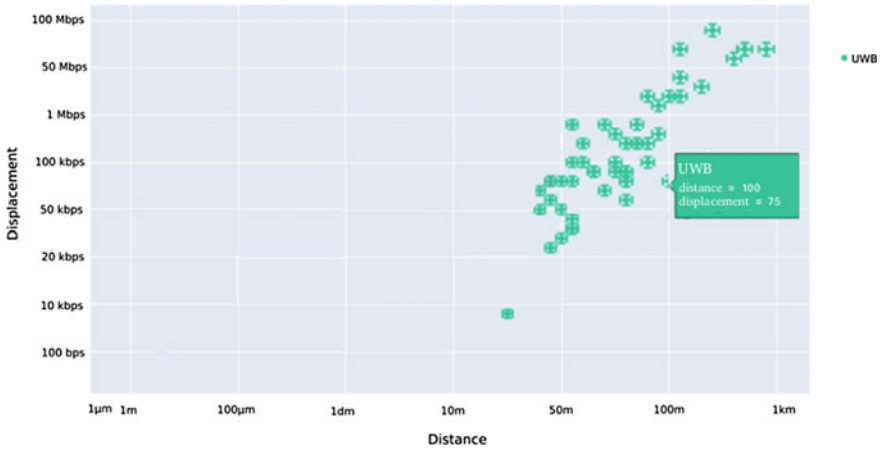


Fig. 11 Distance and displacement of UWB devices

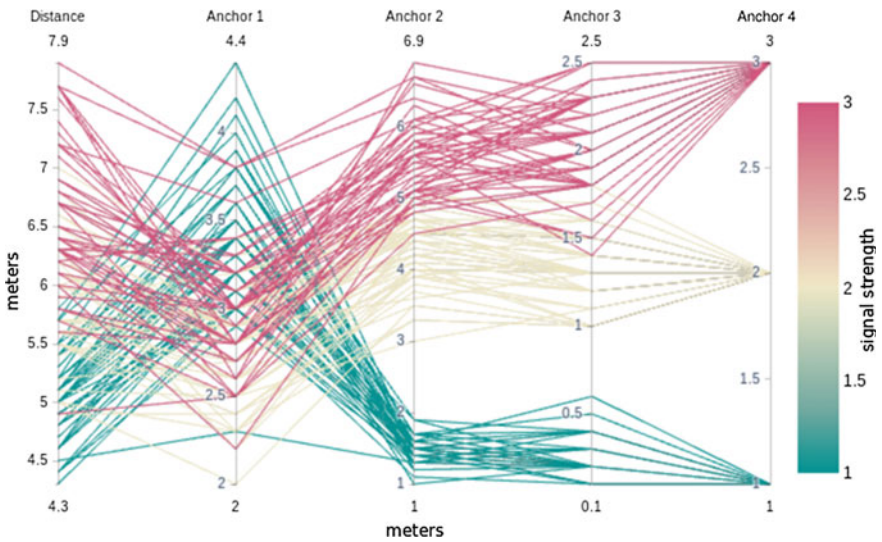


Fig. 12 Signal strength transmitted between the UWB devices

30cm in the indoor environment, and the range of 20m is achieved with LoS or NLoS. OFDM spread spectrum is transmitted with BPSK, QPSK modulation. The channel bandwidth is about 3.5–10GHz.

The inclusion of multiple anchors can be used to improve the performance of the system for enhanced efficiency and more accuracy of data. The direction and position of the human can be facilitated by determining the direction at multiple locations for positioning or moving closer towards the anchor. The experimental results show that high-precision indoor tracking using Ultra-Wide Band devices and open standards is

efficient and robust for indoor environments with multiple tags and specified count of anchors. Implementation of trilateration algorithm for positioning in an indoor environment achieves better efficiency.

References

1. S. Flores, J. Geiß, M. Vossiek, An ultrasonic sensor network for high-quality range-bearing-based indoor positioning, in *2016 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, Apr 2016, pp. 572–576
2. U. Grossmann, M. Schauch, S. Hakobyan, RSSI based WLAN indoor positioning with personal digital assistants, in *2007 4th IEEE Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications*, Sept 2007, pp. 653–656
3. G. Gu, G. Peng, The survey of GSM wireless communication system, in *2010 International Conference on Computer and Information Application*, Dec 2010, pp. 121–124
4. X. He, S. Badieli, D. Aloii, J. Li, Wifi ilocate: Wifi based indoor localization for smartphone, in *2014 Wireless Telecommunications Symposium*, Apr 2014, pp. 1–7
5. R.T. Hocht, Multiple access capacity in multipath channels of delay-hopped transmitted-reference UWB, in *IEEE Conference on Ultra Wideband Systems and Technologies*, vol. 2003, Nov 2003, pp. 315–319
6. A. Jain, P. Tupe-Waghmare, Radiation measurements at repeated intervals for various locations of SIU campus and calculation of compliance distance from cell tower, in *2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICADOT)*, Sept 2016, pp. 804–808
7. I.K. Laga Dwi Pandika, B. Irawan, C. Setianingsih, Application of optimization heavy traffic path with Floyd-Warshall algorithm, in *2018 International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC)*, Dec 2018, pp. 57–62
8. K. Nakano, I. Jinbu, Y. Sate, K. Kato, Three-dimensional position measurement system for indoor flying robot that uses ultrasonic harmonic waves, in *2018 57th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*, Sep 2018, pp. 1261–1266
9. K. O’Keefe, Y. Jiang, M. Petovello, An investigation of tightly-coupled UWB/low-cost GPS for vehicle-to-infrastructure relative positioning, in *2014 IEEE Radar Conference*, May 2014, pp. 1295–1300
10. F. Sato, Y. Motomura, C. Premachandra, K. Kato, Absolute positioning control of indoor flying robot using ultrasonic waves and verification system, in *2016 16th International Conference on Control, Automation and Systems (ICCAS)*, Oct 2016, pp. 1600–1605
11. H.G. Schantz, C. Weil, A.H. Uden, Characterization of error in a near-field electromagnetic ranging (NFER) real-time location system (RTLS), in *2011 IEEE Radio and Wireless Symposium*, Jan 2011, pp. 379–382
12. A.S. Tasbas, E. Erdal, S. Özdemir, Real-time object and personnel tracking in indoor location, in *2019 4th International Conference on Computer Science and Engineering (UBMK) (2019)*, pp. 585–590. <https://doi.org/10.1109/UBMK.2019.8907062>
13. G. Wang, W. Kong, Angle-dependent pulse distortion in UWB radiation and its impact on UWB impulse communications. *Electron. Lett.* **41**(25), 1360–1362 (2005)
14. Y. Wang, H. Zhang, S. Su, Z. Tian, A location privacy-aware method for knn query in location based services, in *2018 IEEE Third International Conference on Data Science in Cyberspace (DSC)*, June 2018, pp. 537–541
15. C. Xiaoli, T. Mao, D. Xiaoping, W. Hongfen, Moving target detection of TWR based on FPGA, in *2011 International Conference on Electrical and Control Engineering*, Sept 2011, pp. 3536–3539

16. D. Xu, W. Zhang, B. Jiang, P. Shi, S. Wang, Directed-graph-observer-based model-free cooperative sliding mode control for distributed energy storage systems in dc microgrid. *IEEE Trans. Ind. Inform.* 1–1 (2019)
17. H. Xu, C. Yuan, P. Li, Y. Wang, Design and implementation of action recognition system based on RFID sensor, in *2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, July 2017, pp. 3021–3025
18. W. Yao, L. Ma, Research and application of indoor positioning method based on fixed infrared beacon, in *2018 37th Chinese Control Conference (CCC)*, July 2018, pp. 5375–5379
19. L. Zhang, R. Takei, J. Lu, N. Makimoto, T. Kobayashi, T. Itoh, Development of wide-band low-frequency MEMS vibration energy harvester for utility infrastructure core monitoring system, in *2017 Symposium on Design, Test, Integration and Packaging of MEMS/MOEMS (DTIP)*, May 2017, pp. 1–4
20. Y. Zhao, Z. Li, B. Hao, P. Wan, L. Wang, How to select the best sensors for TDOA and TDOA/AOA localization? *China Commun.* **16**(2), 134–145 (2019)

Design and Analysis of Single-Phase Inverter for Avionics System



K. Lavenya and M. Umavathi

Abstract In this paper, a single-phase inverter for avionics application is designed. Usually, the load in the aircraft works on high frequency and with a specific power supply. In the proposed work, a single-phase inverter of 115 V, 400 Hz is designed with an input DC power of 28 V. For the proposed system, DC-DC SEPIC converter is used to step up the input voltage to achieve high gain and an H-bridge inverter followed by a filter to get sinusoidal output. To improve power quality, high efficiency and low total harmonic distortion value closed-loop implementation are validated through detailed simulation in MATLAB Simulink and experimentally verified using a prototype hardware model to get high frequency.

Keywords SEPIC converter · H-bridge inverter · LC filter · Feedback loop · MOSFET · Gate driver · Total harmonic distortion (THD)

1 Introduction

The communication, navigation, and other control systems in the aircraft mainly depend on the electric power supply. The electric power for the aircraft is supplied by the generator coupled with an engine or through a battery source for the backup, as the load in the aircraft must have a constant electric power supply without any interrupt [1]. The generator coupled with an engine is known as the primary power source, and the battery source is known as a secondary source. The technology in the field of power electronics is recently increasing due to its low power loss and easy to control. The secondary source is designed with an inverter and DC-DC converter units [2]. Usually, the loads in the aircraft work on high frequency because of its advantages like less machine weight, smaller, or lighter power supply as the size of the machine is reduced. Though the resistive losses are increased with high frequency, while designing the major concern is about the size, and weight therefore losses are neglected [3]. For an uninterrupted power supply to the load, a power

K. Lavenya (✉) · M. Umavathi
Department of Electrical and Electronics Engineering, B.M.S. College of Engineering, Bengaluru, India
e-mail: lavenyak.epe19@bmsce.ac.in

supply is designed for 115 V, 400 Hz with 28 V. To achieve more efficiency and low total harmonic distortion (THD), a transformer-less inverter topology is designed to get desired output frequency and voltage. Instead of a transformer, a passive filter is used to reduce the THD and to get desired output wave shape [4]. In the first stage, to boost the input voltage. A switched-mode power supply boosts the input DC voltage by turning on and off a controlled switch. SEPIC converter is designed. It has advantages like non-inverting output, diode work as a reverse blocking mode. A comparative study is made with different DC-DC converters, and it concluded that using SEPIC converter isolation for input and output is easy, and maximum efficiency is obtained [5, 6]. In the second stage, an inverter is used to convert direct current to alternating current. Sinusoidal pulse width modulation (SPWM) technique is used as it has the advantages of using a small filter at the output, reducing the harmonic distortions [7, 8]. Finally, to obtain sinusoidal waveform output filter is used to convert square wave to sinusoidal wave [9]. In [10], a study is made between synchronous pulse-width modulation and output filter-based inverter. It was concluded that for a load that is sensitive to sinusoidal waveform output filter should be used to reduce the distortions and to achieve pure sinusoidal waveform [11]. In [12], three-phase aircraft is designed from AC main. Here, space vector modulation technique is used to control the inverter switches. Current harmonics of 3% are achieved. All the loads connected to the system are nonlinear which produce harmonics that distort the sinusoidal waveform. According to IEEE standard [13], the fundamental current harmonics should be limited to 5% and the voltage to 8%. In [14], the individual frequency voltage harmonics is 2.3% and current is 4%. In our proposed system, the current and voltage harmonics are made less than 1%.

2 Proposed Method

Figure 1 shows the overall circuit diagram of the project. A 28 Vdc is converted to 115 Vac of 400 Hz. There are two stages of conversion required to achieve the output voltage. The first stage is to boost the input voltage by using a DC-DC converter,

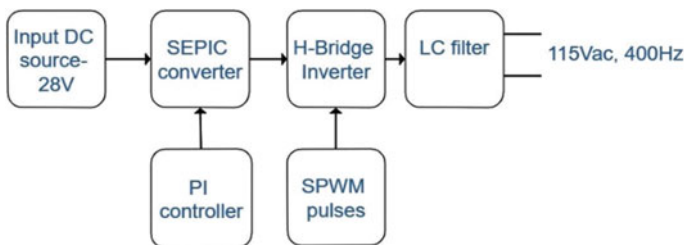
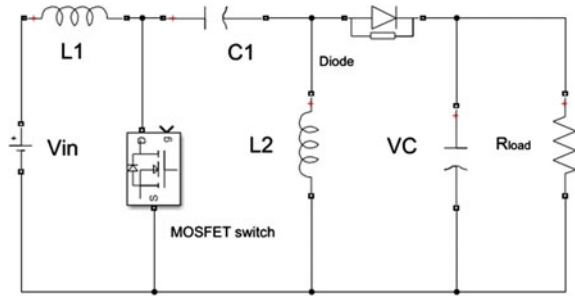


Fig. 1 Block diagram of the proposed method

Fig. 2 SEPIC converter



and the second stage is to use an inverter to convert the DC voltage to the required AC voltage.

In the first stage, 28 Vdc is converted into 200 Vdc by using a SEPIC converter. A PI controller is used to control the duty cycle of the active switch in the converter. In the second stage, 200 Vdc is converted into AC voltage by using an inverter. The output of the inverter is a square wave. To convert it into a sinusoidal waveform, an LC filter is used at the output. PWM technique is used to generate the pulses.

3 Circuit Design and Functions

3.1 SEPIC Converter

The primary step of boosting the input voltage is done by using a SEPIC converter. 28 Vdc input voltage is stepped up to 200 Vdc. SEPIC converter is a buck-boost converter which gives non-inverting output voltage. As the inductor core is connected to the same core, the electromagnetic induction noise is reduced in the SEPIC converter when compared to the boost converter and therefore the input filter can be reduced [15].

Figure 2 shows the circuit of the SEPIC converter. The SEPIC converter is designed for 200 Vdc, 250 W from 28 V input. Here, closed-loop implementation is done using a PI controller [16]. If there is any variation in the output voltage value, the PI controller adjusts the duty cycle of the MOSFET switch and the output voltage is controlled and made constant.

3.2 Design Parameters of SEPIC Converter

The duty cycle D of the system is [17]

$$D = \frac{V_{out} + V_D}{V_{in} + V_D + V_{out}} \tag{1}$$

where V_{out} is the output voltage, V_D is diode forward voltage, and V_{in} is the input voltage.

The load resistance R can be expressed as

$$R = \frac{V_{out}^3}{P_{out}} \tag{2}$$

where P_{out} is the output power.

The output current and the inductor current can be calculated as the system are expressed as

$$I_{out} = \frac{V_{out}}{R} \tag{3}$$

$$I_{L1} = \frac{P_s}{V_s} \tag{4}$$

$$I_{L2} = \frac{P_{out}}{V_o} \tag{5}$$

where R is the load resistance.

I_{out} is the output current.

I_{L1} is the $L1$ current.

I_{L2} is the $L2$ current.

Therefore, inductor $L1$, $L2$ and capacitor $C1$, $C2$ can be calculated [18]. The calculated values are mentioned in Table 1.

Table 1 SEPIC converter specifications

Input voltage (V_{in})	28 V
Output voltage (V_{out})	200 V
Output power (P_{out})	250 W
Inductance $L1$ and $L2$	27 μ H and 195 μ H
Capacitance $C1$ and $C2$	435 μ F and 435 μ F
K_i	10
K_p	50
Load resistance	160 Ω
Switching frequency	25 kHz

$$L_1 = \frac{V_s * D}{f_s * \Delta i_{L1}} \tag{6}$$

$$L_2 = \frac{V_s * D}{f_s * \Delta i_{L2}} \tag{7}$$

$$C_1 = \frac{V_s * D}{R * f_s * \Delta V_{C1}} \tag{8}$$

$$C_2 = \frac{V_s * D}{R * f_s * \Delta V_{C2}} \tag{9}$$

3.3 H-Bridge Inverter

Figure 3 is the H-bridge inverter. In this, both the modes are explained. In mode one, switch $S1$ and $S2$ are on. Therefore, the current flow through V_{dc+} , $S1$, load, $S2$, and back to V_{dc-} . The LC filter is used to smoothen the output waveform that is the square waveform is converted to a sinusoidal waveform [9].

In the mode two, switches $S3$ and $S4$ are turned on. Therefore, the current flows through V_{dc+} , $S3$, load, $S4$, and back to V_{dc-} . In mode one, a positive cycle of the output waveform is obtained. In mode two, negative cycles of the voltage are obtained. The unipolar SPWM technique is used to generate the pulses for the MOSFET switches. The sinusoidal waveform is compared with the triangular waveform to generate the pulses. From a comparative analysis, it is deduced that the unipolar

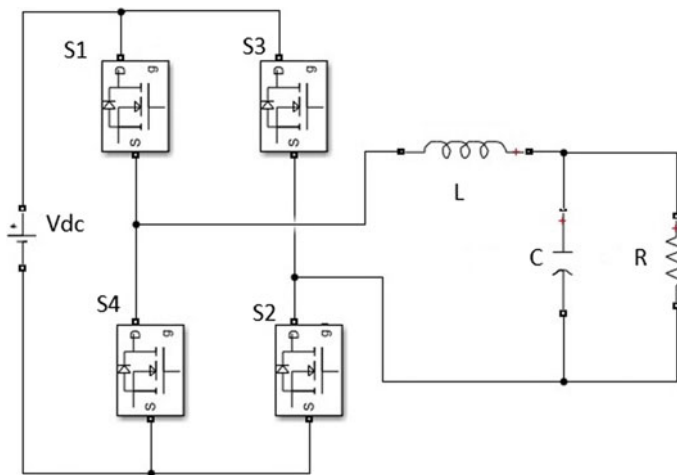


Fig. 3 H-bridge inverter

pulse width modulation technique improves the power quality and decreases the THD by properly selecting the frequency modulation index [19].

3.4 Design of LC Filter

The inverter output is connected with a full load, the maximum current (I_{max}) flowing through it, and the minimum voltage (V_{min}) is noted. The filter component L_f and C_f are calculated using the following expressions [1].

$$R_d = \frac{V_{min}}{I_{max}} = 56 \Omega \tag{10}$$

$$L_f = \frac{R_d}{2 * \pi * f_0} = 21 \text{ mH} \tag{11}$$

$$C_f = \frac{1}{2 * \pi * R_d * f_o} = 7.5 \mu\text{F} \tag{12}$$

4 Simulation Results and Discussion

The circuit has been simulated in the MATLAB platform, and the results were analyzed to check the quality of the design.

Figure 4 shows the overall simulation of the aircraft system. There are two stages of conversion. In the first stage, the input voltage of 28 V is boosted to 200 V using

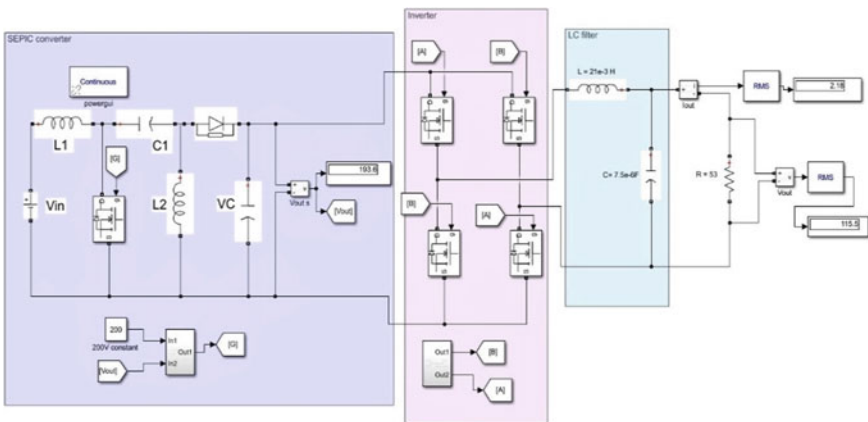


Fig. 4 Proposed method simulation

a SEPIC converter and the output of the SEPIC converter is given to the inverter to convert DC voltage to AC and followed by an LC filter. In the first stage, SEPIC converter boost 28–200 V. Here, 196 V is achieved. 196 V is given to the inverter to convert DC voltage to AC. At the output before filter, 189 V of square waveform is obtained. To convert square to sinusoidal waveform LC filter is designed. From the simulation, it is observed that the output voltage is 115.5 Vac (RMS) and the current is 2.18 A (RMS).

Figure 5 is the output voltage of the SEPIC converter which is near to 200 V. Figures 6 and 7 are the inverter output current and voltage waveform. It is observed from Fig. 7 that the peak voltage is 163 Vm, and the time required to complete one cycle is 2.48 ms, therefore, the frequency is 401 Hz.

Fig. 5 SEPIC converter output voltage

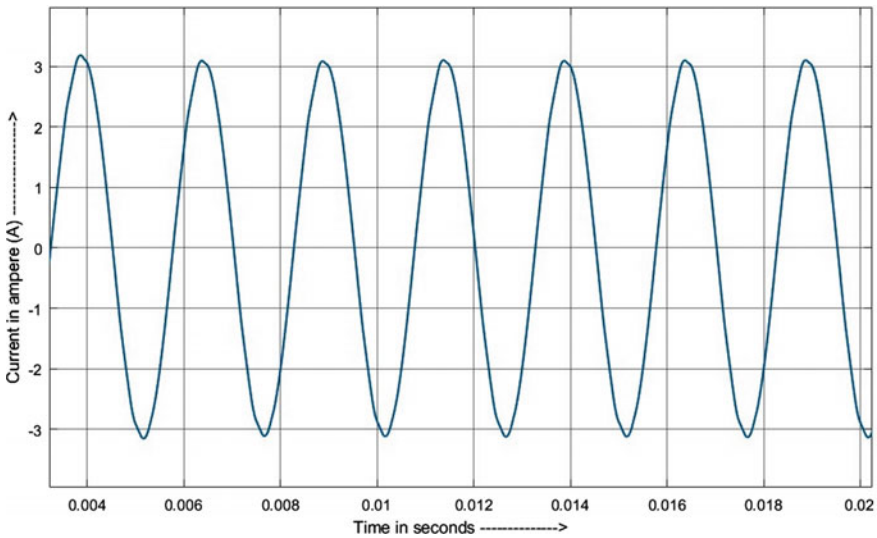
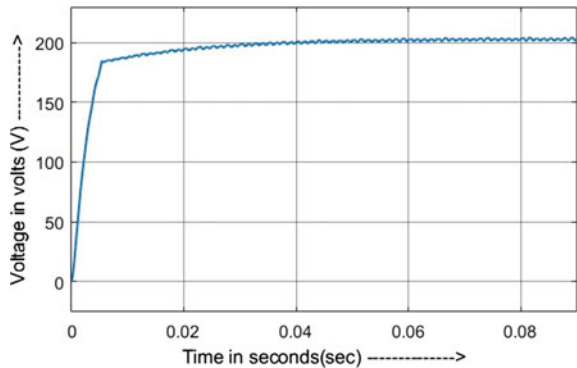


Fig. 6 Inverter output current

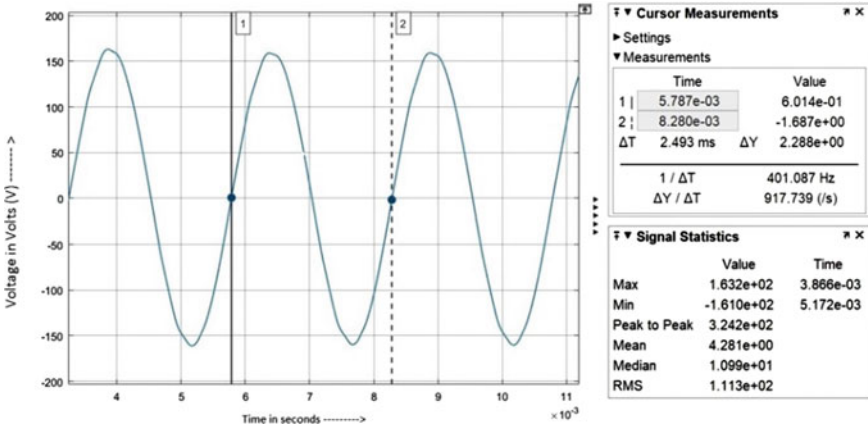


Fig. 7 Inverter output voltage

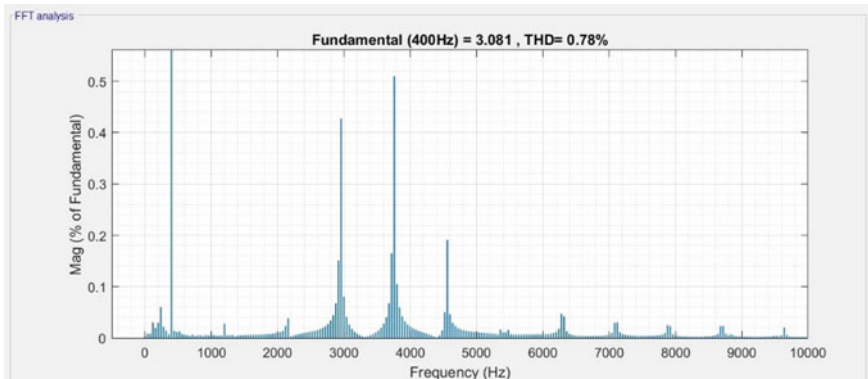


Fig. 8 THD analysis of inverter output current

Figures 8 and 9 show the total harmonic distortions of the current waveform and voltage waveform. The THD obtained is 0.78% for the analysis.

The comparative analysis of overall system is done as mentioned in Table 2.

5 Prototype Experimental Model

A prototype model of the aircraft model is designed to obtain 400 Hz frequency at the output of the inverter. In this model, SEPIC converter is designed for 35 V output from 12 V in input and an H-bridge inverter is constructed with MOSFET switches to get AC output with 400 Hz.

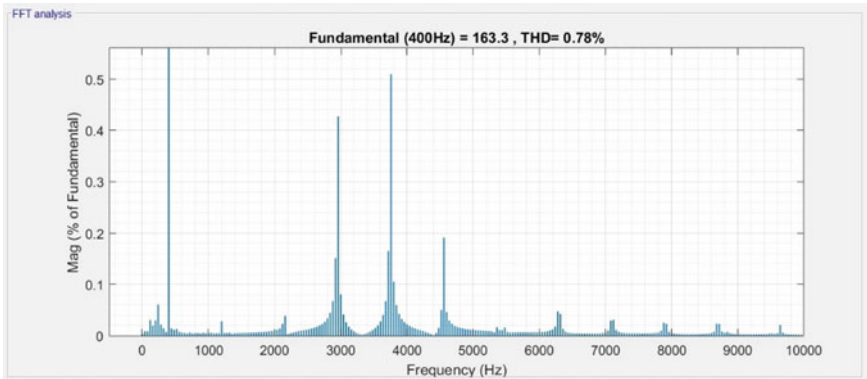


Fig. 9 Inverter output voltage THD analysis

Table 2 Comparison analysis

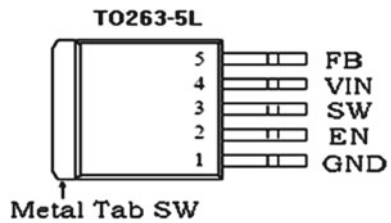
Existing system	Proposed system
Boost converter is used	SEPIC converter is used isolation that is provided for input and output
SG3524N IC is used in hardware to trigger the pulse in boost converter	XL6009 IC is used in SEPIC converter which provide closed loop control
THD = 1.17%	THD = 0.78%

The main components are the inductor, capacitor, and switching regulator. XL6009 IC is used as a switching regulator.

Figure 10 shows the XL6009 IC pin diagram. XL6009 IC is used for closed-loop implementation. The input voltage ranges from 5 to 30 V which makes it possible to use a smaller inductor. The IC support output is current up to 3 A.

Figure 11 shows the SEPIC converter model for the prototype model. The position of $L1$ and $L2$ decides the modes of operation of the converter. The capacitor $C1$ is used to remove the transients from the input supply. Capacitor $C2$ is used for isolating two inductors. Here, Schottky diode is used as it has a fast recovery time and less noise.

Fig. 10 PIN configuration of XL6009 [20]



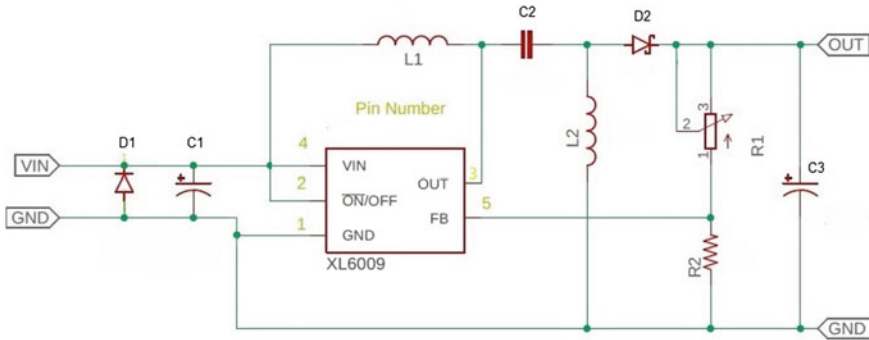


Fig. 11 Hardware schematic of SEPIC converter [20]

In boost mode, when the input voltage is less than the output voltage $L2$ is charged and the current flows through the load when the switch is turned off. The output voltage is checked with the voltage divider circuit, and the duty cycle is skipped to sync the output voltage to the required value.

5.1 Gate Driver

The pulses for H-bridge MOSFET are generated from Arduino Uno. The generated pulses are given to the gate driver which provides isolation between input and output. TL250 is used as a gate driver. The signal from the Arduino is given as input to the driver IC. TLP250 is a high-speed linear optocoupler.

Figure 12 shows the driver circuit, where $C1$ and $C2$ are decoupling capacitors and resistors $R2$ and $R3$ are used to amplify the signal to drive the switch. The input resistance $R1$ depends on the input voltage to offer a certain driver current. Care must

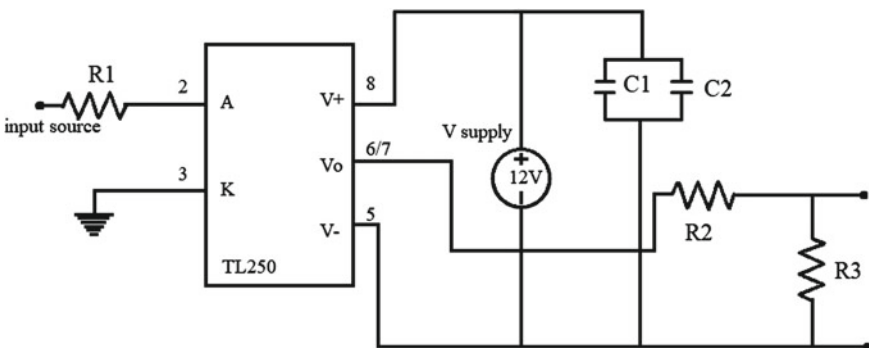


Fig. 12 Gate driver circuit

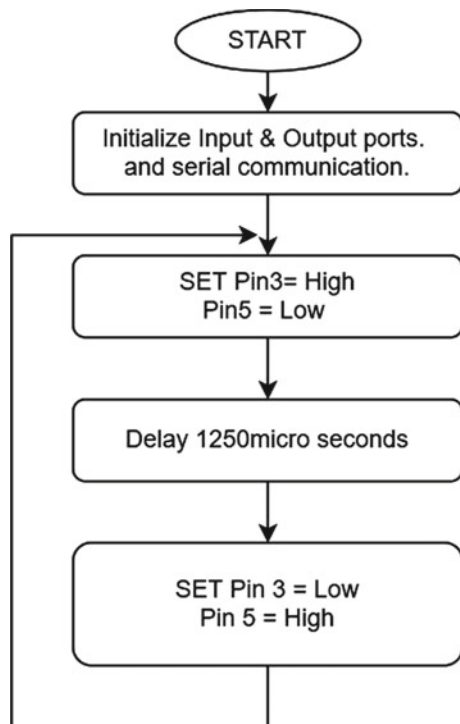
be taken in making a different ground connection for high-voltage and low-voltage sides. As it cannot be connected to the same core because it will lose isolation [21].

The program flowchart for driving the MOSFET in inverter circuit is shown in Fig. 13. First, the input ports and serial communication port are initialized. Pin 3 is set high, and Pin 5 is set low. After a delay of 1250 μ s delay, pin 3 is set low and pin 5 is set high.

Figure 14 shows the complete hardware model. In the first stage, 12 V input is boosted using a SEPIC converter to 35 V. The output of the SEPIC converter is given as the input to the H-bridge inverter. A step-down transformer is used for the voltage source for the gate driver. The pulses are generated from Arduino Uno. To trigger MOSFET switches, four gate driver circuits are used. The output of the inverter is observed in the oscilloscope.

Figure 15 shows the output voltage waveform of the inverter on the oscilloscope. The amplitude value is the peak-to-peak voltage that is 65.60 V, and the frequency is 392.70 Hz. The rise time and fall time are 20 μ s, and the width of the waveform is 1.26 ms. Usually, the output frequency of the inverter is 50/60 Hz, for a particular application like avionics the circuit is designed to get 400 Hz.

Fig. 13 Flowchart of Arduino program



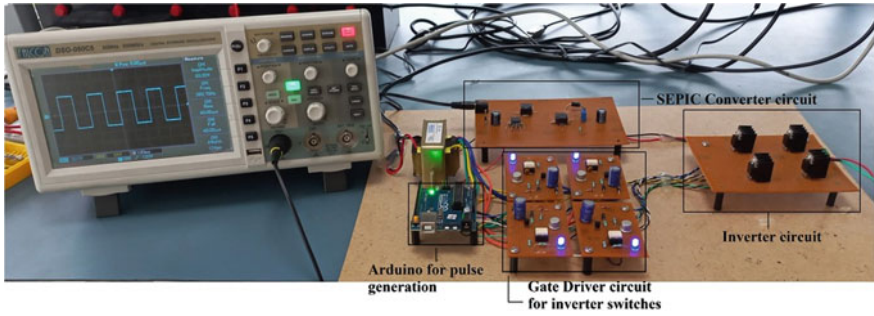


Fig. 14 Complete hardware setup

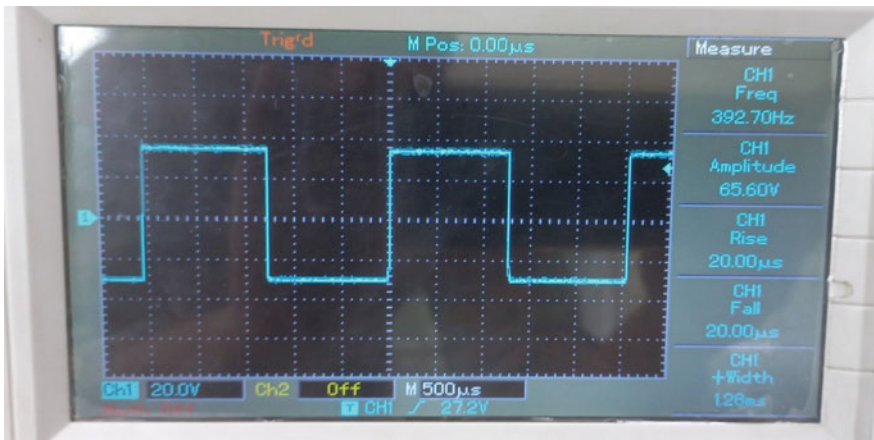


Fig. 15 Inverter output voltage

6 Conclusions

In this work, the high-frequency system for aircraft application was designed. A 115 V of 400 Hz was designed and simulated using a SEPIC converter and H-bridge inverter. This work was experimentally analyzed using a prototype hardware model of 32 V, 2.5 A. The overall efficiency of the proposed system was 88%, and the total harmonic distortion of 0.78 was analyzed for inverter output current using FFT analyzes which is less than IEEE standard. The proposed system was simulated in MATLAB (R2018a).

Acknowledgements Many people merit my true appreciation for helping me to finish this project. First and chief, I want to express my true appreciation to my Guide Dr. Umavathi M, Assistant Prof. Dept. of EEE, B.M.S. College of Engineering, Bull Temple Road, and Bengaluru for her significant direction and assistance.

References

1. M.N. Parripati, V.S. Kirthika Devi, A single phase DC-AC inverter for aircraft application, in *International Conference on Inventive Systems and Control (ICISC)* (2020)
2. O.D. Dsouza, P. Manjunath, S. Arjun, Design of a single phase inverter for aircraft applications. *Int. J. Innov. Technol. Explor. Eng. (IJITEE)* (2019)
3. E. Sener, G. Ertasgin, Design of a 400 Hz current-source 1-ph inverter topology for avionic systems, in *5th International Conference on Advanced Technology & Sciences (ICAT'17)*, Istanbul, Turkey May 09–12
4. Md. K. Islam, Md. M. Rahman, Md. F. Rabbi, Transformer less, lower THD and highly efficient inverter system, in *2015 International Conference on Advances in Electrical Engineering (ICAEE)*, Dhaka, Bangladesh, 17–19 Dec 2015
5. C. Verma, B. Anjanee Kumar, Comparison of dc-dc converters with SEPIC converter for wind driven Induction generators. *Int. J. Eng. Trends Technol. (IJETT)* **39**, 180–183 (2016)
6. V. Sharma, U. Sultana, U.K. Gupta, Designing of modified SEPIC converter for LED lamp driver. *IJLTEMAS* (2014)
7. N.F. Abdul Hamid, M.A. Abd Jal, N.S. Syahirah Mohamed, Design and simulation of single phase inverter using SPWM unipolar technique, in *ICE4CT 2019 Journal of Physics: Conference Series* (2019)
8. J. Soomro, T.D. Memon, M.A. Shah, Design and analysis of single phase voltage source inverter using unipolar and bipolar pulse width modulation techniques, in *International Conference on Advances in Electrical, Electronic and System Engineering*, Malaysia (2016)
9. W. Zhang, X. Zhang, W. Liu, W. Li, Modeling and simulation of aviation static inverter based on Dymola and Modelica, in *International Conference on Aircraft Utility Systems (AUS)* (2016)
10. M. Bilal Cheema, S. Ali Hasnain, M. Maaz Ahsan, et al., Comparative analysis of SPWM and square wave output filtration based pure sine wave inverters, in *IEEE 15th International Conference on Environment and Electrical Engineering (EEEIC)*, Rome, Italy, 10–13 June 2015
11. A. Ahmad, S. Maqbool, Square wave inverters—a performance comparison with pure sine wave inverters. *Int. J. Res. Electron. Comput. Eng. (IJRECE)* (2019)
12. G.L. Basile, S. Buso, F. Fasolo, P. Tenti, P. Tomasin, A 400 Hz, 100 kVA, digitally controlled UPS for airport installations, in *Conference Record of the 2000 IEEE Industry Applications Conference. Thirty-Fifth IAS Annual Meeting and World Conference on Industrial Applications of Electrical Energy (Cat. No.00CH37129)* (2000)
13. IEEE Recommended Practice for Conducting Harmonic Studies and Analysis of Industrial and Commercial Power Systems, IEEE-SA Standards Board, 27 Sept 2018
14. C. Rivetta, F. Arteché, F. Szoncsó, A common 400Hz AC power supply distribution system for CMS front-end electronics. in *8th Workshop on Electronics for LHC Experiments* (2002)
15. M. Umavathi, K. Udhayakumar, D. Thanya, Investigation on solar-wind hybrid renewable energy system using single SEPIC converter, in *National Conference on Advancement in Electrical Sciences at SNS College of Technology*, Coimbatore, 13 Apr 2019.
16. H.H. Tesfamikael, A. Fray, I. Mengsteab, A. Semere, Z. Amanuel, Construction of mathematical model of DC servo motor mechanism with PID controller for electric wheel chair arrangement. *J. Electron. Inform.* (2021)
17. I.A. AlMohaisin, A.A. Mahfouz, V.T. Akhila, A review on SEPIC converter topologies. *Int. J. Res. Eng. Sci. Manage.* (2019)
18. J. Leema Rose, B. Sankaragomathi, Design, modeling, analysis and simulation of a SEPIC converter. *Middle-East J. Sci. Res.* **24** (2016)
19. G.M. Tina, G. Celsa, A Matlab/Simulink model of a grid connected single-phase inverter, in *50th International Universities Power Engineering Conference (UPEC)* (2015)
20. XLSEMI [Online]. Available: <https://datasheetspdf.com/pdf/775384/XLSEMI/XL6009/1>
21. J. Xu, K. Han, The single-phase inverter design for photovoltaic system, in *International Symposium on Computer, Consumer and Control* (2016)

IoT-Based Novel Framework for Solid Waste Management in Smart Cities



Mohd Anjum, M. Sarosh Umar, and Sana Shahab

Abstract With the increasing settlements, advancement in lifestyle and services, smart cities are experiencing many challenges in solid waste management (SWM) services. The smart city development framework comprises Internet of things (IoT) and information and communication technology (ICT) as key technologies to build an efficient solutions for various services. This research has proposed an IoT-based innovative novel framework to deal with the various challenges involved in SWM services. The framework performs real-time monitoring of waste level in the bin and provides the optimize routes for efficient collection of waste. The overall system constitutes three major components, namely smart bin, smart truck and work server. The smart bin design incorporates various sensors such as ultrasonic, temperature, humidity, load cell, proximity and accelerometer along with radio frequency identification (RFID) tag and wireless communication module. The smart truck is equipped with RFID reader, Global Positioning System (GPS) and wireless communication modules. The work server comprises web server, database management system along route optimization and decision-making programs. The implementation of framework at massive level will reduce the manpower effort and operating cost, optimize the resources needed, improve the environmental quality and contribute to decrease in traffic congestion and noise pollution. Additionally, an experiment was performed using developed prototype which comprised the simplified version of proposed framework. It incorporated the ultrasonic sensor with communication devices. The testing was performed to verify the working of the prototype. Additionally, it was checked that the system was able to detect the different levels of waste in the bin.

M. Anjum (✉) · M. Sarosh Umar
Department of Computer Engineering, Aligarh Muslim University, Aligarh, India
e-mail: mohdanjum@zhcet.ac.in

M. Sarosh Umar
e-mail: saroshumar@zhcet.ac.in

S. Shahab
College of Business Administration, Princess Nourah Bint Abdulrahman University, Riyadh,
Saudi Arabia
e-mail: sshahab@pnu.edu.sa

Keywords Solid waste (SW) · IoT · ICT · Smart city · Smart bin · Smart truck · Sensor · RFID

1 Introduction

Increasing volume of the SW has emerged as a critical challenge especially in smart cities of developing countries due to fast settlement, exceptional socio-economic development, and unprecedented advancement in lifestyle. The World Bank report about global SW exhibits that worldwide generation of SW was approximately 2.01 billion tons per year in 2016. At least 33% of above is not managed in environmentally safe manner. It is estimated that global population will generate nearly 2.59 and 3.40 billion tons per annum by the year 2030 and 2050, respectively [1]. In last two decades, the waste generation has grown tremendously in urban regions of developing countries. Nowadays, most of these areas are being incorporated with IoT-enabled services to develop smart cities. The IoT refers to a system of interconnect physical devices or objects, called things that are integrated with various sensors, custom software and ICT to transfer the data with other objects using Internet [2]. IoT enables to manage these objects from remote location through Internet; it provides the protocols to implement the wirelessly controlled systems into the real world [3].

A smart city refers to an urban region that utilizes the IoT technologies to exchange information which is employed to manage resources and administer services effectively [4]. One of such crucial services is SW collection and transportation to recycling and landfill disposal facility [2]. The literature exhibits that many factors play a critical role in making SWM in smart cities more challenging and directly affect the SWM services. These key factors comprise massive migration of people from rural areas to smart cities, growing rate of consumption of goods due to high economic standards and advanced living style, lack of data to simulate the forecasting models for waste generation prediction and improper use of IoT technologies to support the SWM system [3]. Now, it is demanded to develop an IoT-based efficient SWM system for smart cities to effectively manage the increasing amount of waste in environmentally safer manner.

The objective of this study is to illustrate the various IoT and ICT technologies suitable for developing the SWM system and proposes the novel SWM architecture based on these technologies to efficiently manage all the activities involved in the SWM process. The physical implementation of proposed framework at massive level will significantly decrease the operating cost of waste collection process through exchange of information in real time among bins, collection vehicles and central server. Therefore, smart city administration involved in SWM activities are diverging from the conventional approach to highly advanced and technological alternatives to fabricate more automated and service-oriented system. The proposed conceptual framework exploits the tremendous power of IoT protocols that instate the traceability among heterogeneous physical devices utilizing existing ICT infrastructure.

The research illustrates the blueprint of an integrated physical SWM system which combines the diverse range of technologies and primarily exploits the tremendous power of sensors and wireless access of network to enhance the smart city SWM services. The system design comprises of three major components, namely smart bin, smart truck and work server. The smart bin comprises the IoT-based prototype fabricated by various sensors to measure the level of waste in bin, weight, humidity, temperature and lid state. Additionally, it incorporates the RFID for unique identification of bins and GPS to track the location of bin. It has the capacity to transfer gathered information to central server through a wireless network. The gathered data are further employed as an input to statistical forecasting and artificial neural network models to predict the bin fill level and amount of waste generation in future. Moreover, real-time bin status can also be directly used to determine the visiting nodes for collecting vehicle. Smart truck has also the capability to transfer the data to central sever through wireless network and equipped with RFID reader to read the bin identity and GPS for tracking the current position of the truck. Work server comprises the web server to host and manage the web application, database management system and algorithms to identify optimize route and schedule for smart trucks fleet. Additionally, it also incorporates the various decision-making and user services programs. The major contributions of this paper are given as follows.

- The proposed novel framework will produce a smart SWM system that will provide the real time and efficient waste collection and monitoring processes for smart city.
- It will significantly control the pollution caused by the waste and enhance the environmental quality which will lead to decrease in resident's health problem by periodically creating emergency alters for collection. The enhancement in environmental quality will support in maintaining the smart environment within the smart city.
- It will substantially decrease the operating cost, optimize the resources needed and provide the methodology to utilize the resources in efficient way.
- It will remarkably reduce both human resource for running the SWM system and the manpower effort due to effective utilization of smart bin.
- It will also contribute to decrease the city traffic congestion and noise pollution due to the scheduled transportation.
- A prototype was built to demonstrate the real-time monitoring of the bin.

2 Literature Survey

Nowadays, IoT technology is more pervasive and usable due to ubiquitous Internet connectivity and significant enhancement in data transfer speed. Therefore, a lot of researchers have exploited these advantages to build novel systems and applications for supporting smart city development [5]. A smart city is described as: "A Smart City is a city well performing in a forward-looking way in the following

fundamental components (i.e., Smart Economy, Smart Mobility, Smart Environment, Smart People, Smart Living, and Smart Governance), built on the ‘smart’ combination of endowments and activities of self-decisive, independent and aware citizens” [6]. According to above definition, one of the building pillars of smart city is smart environment which is directly related to environmental pollution. The principal measure to limit the environmental pollution in a smart is the IoT-enabled waste monitoring, collection and disposal system.

A real-time decision support system has been implemented to dynamically optimize the route of the waste collection truck in a smart city. This system has significantly reduced the energy consumption and optimized the operational efficiency [2]. The system is integrated with a data sharing model to exchange data in real time among truck drivers, and incorporates surveillance camera to capture the images of areas that have waste management related problems [2]. Above implemented system does not involve the bin status monitoring, so truck drivers can visit empty or semi-empty bins, which may cause the energy demand and reduce the system throughput. A multiagent-based waste management architecture has been simulated to determine the performance of the system in various scenarios. The system has three agents, namely bin incorporated with ultrasonic sensor and global system for mobile communication (GSM) module, waste collection trucks and citizens. The bin status is transferred to the central database and status-full, bin locations are used as nodes to determine the optimize path [7]. Misra et al. and Khan et al. have built a smart bin which comprises the ultrasonic sensor to automatically determine the garbage level inside the bin and gas sensor to sense the hazardous gases. The sensor data is transferred to the central web server along with cloud and utilized by the concern authority to take immediate action related to physical condition of the bin [8, 9].

Additionally, Sankeerth et al. have also developed similar type of smart bin without gas sensor [10]. A multitiered architecture for smart city development has been proposed and applied to the SWM services. This architecture has remarkably involved IoT and ICT technologies to manage various services in smart city [11, 12]. Additionally, it has integrated SWM system with other services such surveillance, transportation, infrastructure and provided the design protocols of each layer [11, 12]. A novel approach, artificial intelligent of things, has been introduced for real-time monitoring of the trash bin. This approach implements the fuzzy expert system to determine the appropriate locations for trash bin installation, and real-time bin status data is utilized as an input in optimization algorithm to obtain optimal path for collecting vehicles [13]. Bin status data has also been used to train the machine learning algorithm which predicts the probabilities of different waste levels in bin [5, 14]. Additionally, graph theory and machine learning algorithm have been applied on same data to compute the shortest collection paths [5]. Ali et al. and one other have built hardware of smart bin using sensors used GSM module to transfer data from bin to web server and GPS module to track the location of waste collecting vehicles [3, 15]. The location data is utilized to compute the total distance and sensors data is employed to estimate the forecasting model parameters [3, 16]. Baldo et al. has proposed the multilayer infrastructure for smart city waste management using long range wide area network (LoRaWAN). RFID is incorporated to uniquely identify the

bin along with various sensors. This system exploited the data transferring capabilities of LoRaWAN and integrated with city surveillance which is used to detect fire and drop of garbage in proximity of bin [17]. Table 1 displays the comparison of various IoT, and ICT components employed in system development to accomplish the various tasks.

3 Proposed Innovative Novel SWM Framework

3.1 *Smart Bin*

Smart bin has two tier architectures, microcontroller at the core and sensors, namely ultrasonic, proximity, load cell, accelerometer, temperature and humidity, wireless communication module, namely GSM module, GPS, RFID tag, powered with solar energy at the periphery as shown in Fig. 1. A programmable microcontroller process detective measures of the bin to periodically communicate with central server enabling remote retrieval of data. An accelerometer is an electromechanical dynamic sensor capable of measuring force in one-, two- or three-dimensional space. Generally, it remains in idle mode and only responds with displacement of lid. Core functioning of the proximity sensor is to detect nearby objects and intimate the lid status to the microcontroller. Remaining sensors start transmitting status signals to microcontroller as soon as the lid is opened.

An ultrasonic sensor is a distance detector that record time lag between sending and bouncing back of an acoustic wave. Its basic function is to detect the level of waste in the bin and perceive the fullness of the bin. Another aspect to measure the bin overflow is achieved with a load cell sensor. It efficiently measures the weight by transforming the output into a voltage signal. The environmental sensor is necessary to provide ambient condition inside the bin for suitable functioning of other sensory activities. Along with the above features, bin encompasses GPS for describing its position and GSM module for interacting with an external network. Self-describing property of bin is implemented through RFID and microcontroller is programmed such that bin responds intelligently on any activity of peripherals.

3.2 *Smart Truck*

Nowadays, waste collecting trucks must be smarter with the incorporation of modern electronics and communication technologies for collecting and communicating essential data related to a daily schedule, driver personal information and bins visited for more productivity and transparency of fleet operations. Additionally, the everyday list of picking nodes is assigned to each truck related to its specific route. The truck with onboard microcontroller interfacing with GPS module, RFID reader and GSM

Table 1 Demonstration of key IoT and ICT components involved in smart waste management

References	Objective	Prototype	IoT and ICT components
[17]	Waste management infrastructure based on multilayer LoRaWAN	Physical	Microcontroller, Ultrasonic, Temperature, Tilt sensor, LoRaWAN module, RFID and Video surveillance unit Edge computing, Image processing and Web application
[16]	Smart bin monitoring and waste collection	Physical	Microcontroller, Ultrasonic sensor and GSM module Web application, Bin monitoring and collection algorithm
[3]	Smart bin building, waste collection, monitoring and prediction	Physical	Microcontroller, GSM module, Ultrasonic, Flame, Temperature, Humidity and Weight sensor MySQL, Web application, Distance calculation model and Waste forecasting algorithm
[5]	Smart bin building, monitoring and collection	Physical	Microcontroller, Ultrasonic sensor and LoRa module Machine learning algorithm for optimal route
[13]	Artificial intelligent of things based real-time bin monitoring	Physical	Microcontroller, Ultrasonic sensor, GPS and GSM module Web application, Cloud, Fuzzy logic and Route optimization algorithm
[11]	Waste collection and management infrastructure	Physical	Microcontroller, Wi-Fi module, RFID tag and reader Web application and Cloud database,
[10]	Smart bin construction, and monitoring	Physical	Microcontroller, Ultrasonic sensor and Wi-Fi module Web application
[8]	Cloud based waste monitoring	Physical	Microcontroller, Ultrasonic, Gas sensor and Wi-Fi module Web application, Route optimization and cost analysis algorithm
[7]	Multiagent based smart waste collection and monitoring	Simulated	Microcontroller, GSM module and Ultrasonic sensor Web interface, Database and Route optimization algorithm
[2]	Intelligent transportation system for waste collection	Simulated	Location: OpenStreetMap, Routing: GraphHopper and Web application

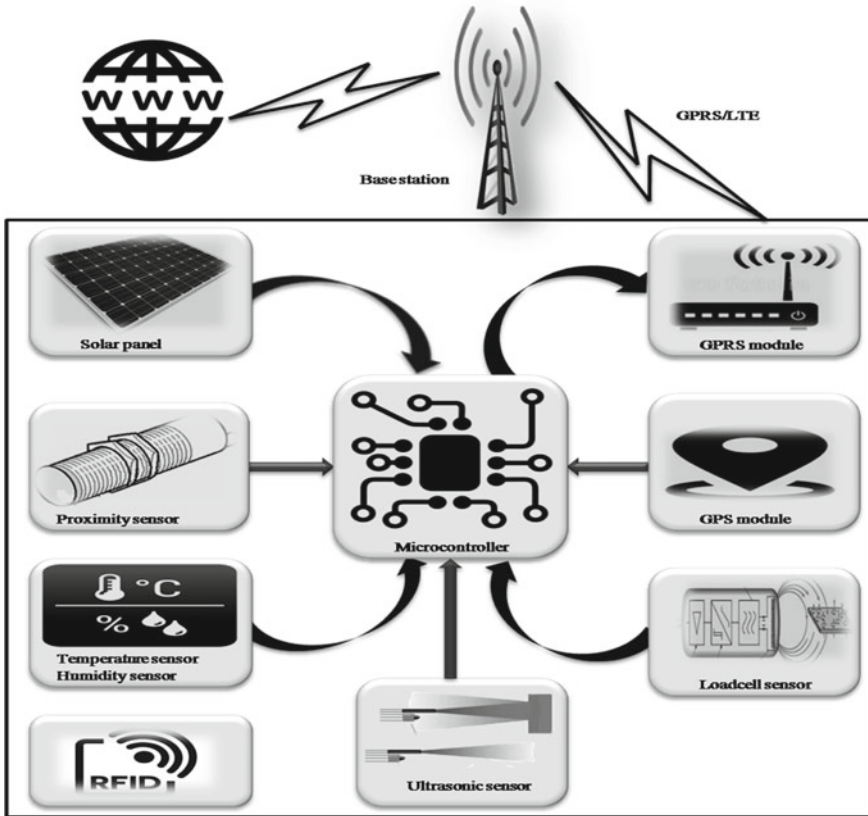


Fig. 1 Solid waste bin internal architecture

module to impart precise performance declaration and real-time communication is shown in Fig. 2. RFID reader reads the bin unique ID before unloading the waste and communicates bin unloading information along with bin ID to work server. GPS module traces the path followed by collecting vehicle and transmit over the network through GPRS module.

3.3 Work Server

Work server hosts the waste management web application, route optimization program and database management system. It contains continuously running multi-threaded programs that establish communication to handle the requests from the vehicles and bins. Real-time data is updated related to the bin's status, current position and volume of waste collected by trucks. Database server program manages



Fig. 2 Communication link between on board devices of smart truck and bin

the database, which is used as input for the optimization program to generate daily schedules and routes for each truck. Web server program responds the requests from clients to access data related to the waste management system. It also maintains the dynamic web interface visualizing current bin status, route map and citizen communication with system and complaint facility. Big data analytics tools are applied on dynamic sensing data to make decision related to waste management and administrative services. The solid waste optimization and collection problem consider static input and dynamic input to evaluate and handle these inputs server framework as presented in Fig. 3.

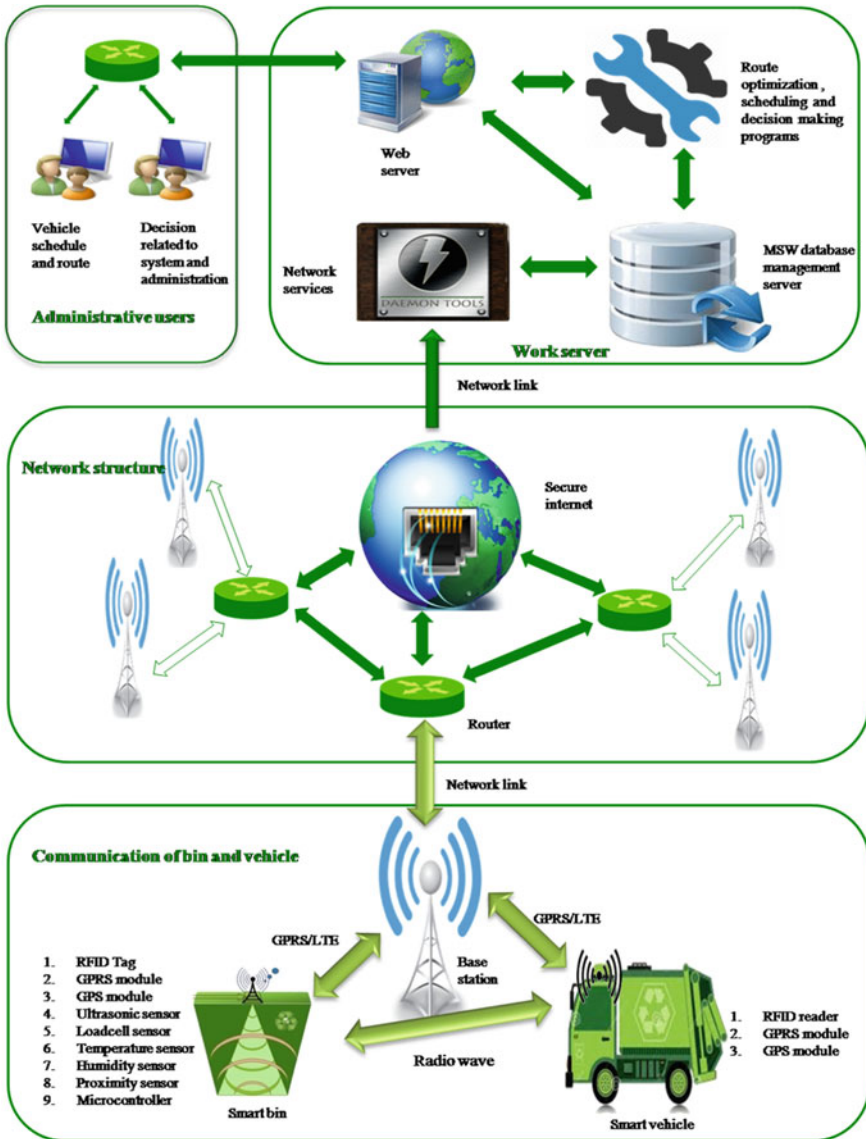


Fig. 3 Architectural overview of solid waste management system

3.4 Working Principle

With the displacement of the lid for loading and unloading, the accelerometer detects the acceleration to estimate the operating parameters and sends signal to the microcontroller to awaken the other sensory devices of bin if operating situations are

achieved, else it remains in idle mode to reduce energy consumption. After awakening, proximity sensor records time lag of lid motion and intimates about the overloading status and microcontroller directs other sensors to percept their status parameters, namely ultrasonic and load cell sensor to highlight the level and weight of garbage. In event of bin unloading, microcontroller develops a frame of sensor data encapsulated with bin information and transmits it to work server. In process of unloading, bin affixed with RFID tag receives signal from vehicle mounted with the reader when within vicinity area of bin and acquire bin detail and location. This information, along with bin status, is updated on central server through GSM module connected with microcontroller and confirms the unloading operation.

A multithreaded program continuously runs on work server, which responds to connection request and updates dynamic data of bin and vehicle in the database. Collected data is managed and manipulated through MySQL to retain consistency in the transaction. This system assists authority or decision makers to reallocate the bin and vehicle routes and oversee management problems related to system performance and waste generation quantity of specified area. Dynamic information obtained via sensory devices with GSM module enhances transparency in system and provides dynamic scheduling of each vehicle, enumerate the quantity of assets, efficient monitoring and verify task accomplishment.

4 Experimental System and Results

The study has proposed an IoT-based ideal SWM framework for smart cities which comprises various sensors, RFID and GPS along with other devices. But the experimental system design comprises the simplified version of proposed framework which incorporates the ultrasonic sensor with communication devices. The following sections illustrates the architecture of experimental system, its working and results.

4.1 *Experimental System Design*

Figure 4 displays the schematic block diagram of experimental prototype for waste monitoring along with placement of devices in bin. The prototype incorporates microcontroller in its core, and ultrasonic sensor, light emitting diode (LED) indicator and GSM module at peripheral.

The ultrasonic sensor measures the level of waste inside the bin and sends this data to microcontroller. The microcontroller generates the warning message if the waste level is more or equal to the threshold level. The warning message is sent via short message service (SMS) to the municipal authorities using the GSM module. Additionally, LED is used to indicate the different status of bin to the residents.

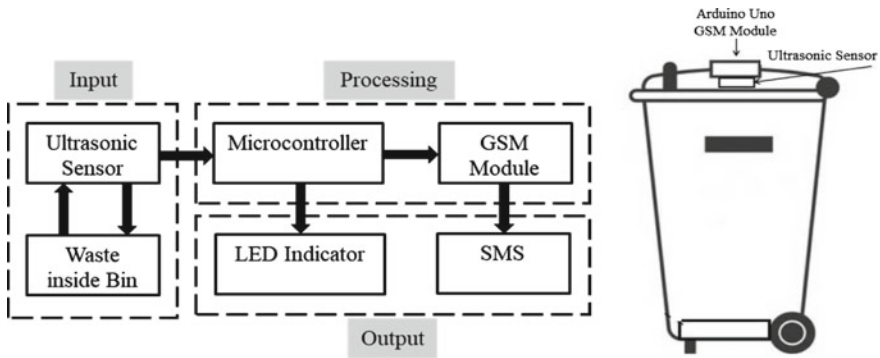


Fig. 4 Block diagram of experimental system and placement of devices in bin

4.2 System Work Flow

The ultrasonic sensor measures the level of waste in bin and send to the Arduino Uno for processing. After processing the received information, microcontroller evaluates the condition whether the waste level has surpassed the threshold level or not. In this system, two threshold levels are set, namely TL1 and TL2. TL1 is set approximately at 75% of the bin height from bottom while TL2 is at 90%. If the waste level crosses the threshold TL1, then microcontroller sends the first warning message to the municipal authority and the green LED will turned on to indicates the status to the residents. Similarly, if the threshold TL2 crosses, then second warning is sent and red LED is turned on.

4.3 Testing

The experimental prototype and hardware connection setup are displayed in Fig. 5. The ultrasonic sensor HC-SR04 is connected to the I/O pin of the Arduino Uno while the SIM900 GSM/GPRS module is serially connected to the microcontroller board. In this serial connection, D7 (RX port) and D8 (TX port) pins of the GSM module are connected to the TX and RX port of the microcontroller. Additionally, both LEDs are connected to the microcontroller.

The testing was performed to determine the outcomes for different waste levels in the bin. First test is conducted to the condition when the bin was empty or waste level was below the threshold TL1. Then, it was observed that neither LED was turned on nor the SMS was obtained. Then, the bin was filled with more waste until its level surpassed the threshold TL1. Now, it was obtained that the green LED was turned on and first warning SMS was sent to the municipal authority as shown in Fig. 6. Then, more waste was added until the bin was completely filled or exceeding the threshold TL2. Now, as the outcome, the green LED was turned off while the red LED was

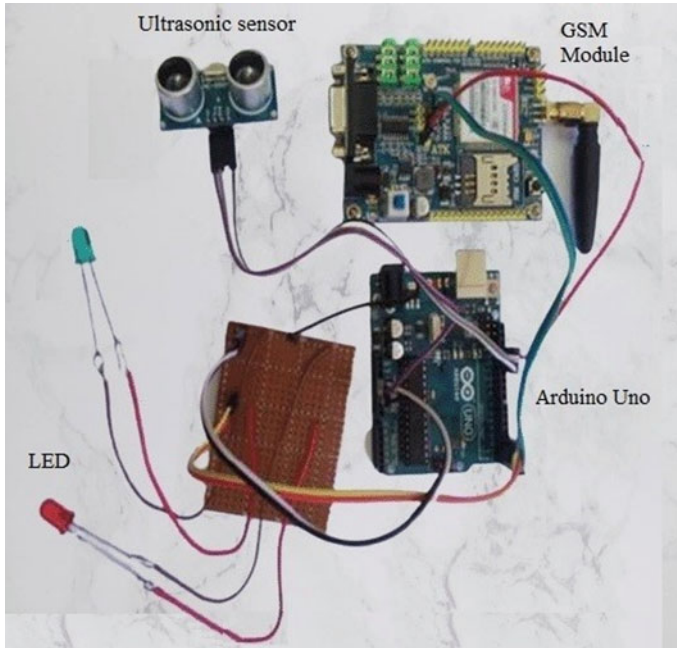


Fig. 5 Connection of components in prototype

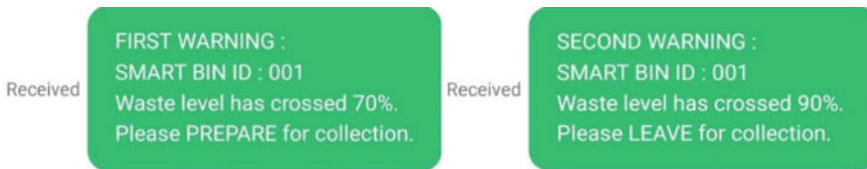


Fig. 6 Received messages after surpassing TL1 and TL2

turned on and second warning message was sent to the municipal authority as shown in Fig. 6.

5 Conclusion

The study has presented a novel framework for smart cities to collect and monitor the SW in efficient and smart way. The system design comprises the IoT sensing prototype to measure the waste bin status such as waste level in bin, waste load, environmental conditions and bin lid states. The sensory information is sent to work server along with bin location which is utilized for real-time monitoring and extracting the

optimize path for collection. RFID is utilized to uniquely identify the bin, and the smart truck sends the information of collected bins to central database. The work server manages the sensory, integrates optimization program to make the day-to-day waste collection efficient and decision-making programs to support the municipal administration.

Additionally, the research also highlights the various IoT and ICT components that have been potentially used to develop the efficient SWM system for smart cities. The proposed framework reduces the waste collection time and save energy through skipping the visit of empty and semi-empty bins. It optimizes the system operating cost, resources needed and manpower effort. It significantly controls the pollution to protect the environmental quality and resident's health. It also contributes to decrease the city traffic congestion and noise pollution due to the scheduled transportation. Moreover, the physical implementation of simplified version of proposed design concludes that complete system can be implemented successfully at massive level in real world to improve the SWM services.

References

1. S. Kaza, L.C. Yao, P. Bhada-Tata, F. Van Woerden, *What a Waste 2.0: A Global Snapshot of Solid Waste Management to 2050* (World Bank Publications, The World Bank Group, Washington, DC, 2018). <https://doi.org/10.1596/978-1-4648-1329-0>
2. A. Medvedev, P. Fedchenkov, A. Zaslavsky, T. Anagnostopoulos, S. Khoruzhnikov, Waste management as an IoT-enabled service in smart cities, in *Internet of Things, Smart Spaces, and Next Generation Networks and Systems* (Springer, Cham, 2015), pp. 104–115. <https://doi.org/10.1007/978-3-319-23126-6>
3. T. Ali, M. Irfan, A.S. Alwadie, A. Glowacz, IoT-based smart waste bin monitoring and municipal solid waste management system for smart cities. *Arab. J. Sci. Eng.* **45**, 10185–10198 (2020). <https://doi.org/10.1007/s13369-020-04637-w>
4. N. Abdullah, O.A. Alwesabi, R. Abdullah, IoT-based smart waste management system in a smart city. Springer International Publishing (2019). https://doi.org/10.1007/978-3-319-99007-1_35
5. T. Anh Khoa, C.H. Phuc, P.D. Lam, L.M.B. Nhu, N.M. Trong, N.T.H. Phuong, N.V. Dung, N. Tan-Y, H.N. Nguyen, D.N.M. Duc, Waste management system using IoT-based machine learning in University. *Wirel. Commun. Mob. Comput.* 1–13 (2020). <https://doi.org/10.1155/2020/6138637>
6. R. Giffinger, N. Pichler-Milanović, *Smart Cities: Ranking of European Medium-Sized Cities*. <http://www.smartcities.eu>
7. E.D. Likotiko, D. Nyambo, J. Mwangoka, Multi-agent based Iot smart waste monitoring and collection architecture. *Int. J. Comput. Sci. Eng. Inf. Technol.* **7**, 1–14 (2017)
8. D. Misra, G. Das, T. Chakraborty, D. Das, An IoT-based waste management system monitored by cloud. *J. Mater. Cycles Waste Manag.* **20**, 1574–1582 (2018). <https://doi.org/10.1007/s10163-018-0720-y>
9. A.A. Khan, A.A. Sajib, F. Shetu, S. Bari, M.S.R. Zishan, K. Shikder, Smart waste management system for Bangladesh, in *ICREST 2021—2nd International Conference on Robotics, Electrical and Signal Processing Techniques* (Institute of Electrical and Electronics Engineers Inc., 2021), pp. 659–663. <https://doi.org/10.1109/ICREST51555.2021.9331159>

10. V.P. Sankeerth, V.S. Markandeya, E.S. Ranga, V. Bhavana, Smart waste management system using IoT, in *International Conference on Inventive Computation Technologies* (Springer, 2019), pp. 661–668. https://doi.org/10.1007/978-3-030-33846-6_71
11. P. Marques, D. Manfro, E. Deitos, J. Cegoni, R. Castilhos, J. Rochol, E. Pignaton, R. Kunst, An IoT-based smart cities infrastructure architecture applied to a waste management scenario. *Ad Hoc Netw.* **87**, 200–208 (2019). <https://doi.org/10.1016/j.adhoc.2018.12.009>
12. M.U. Sohag, A.K. Podder, Smart garbage management system for a sustainable urban life: an IoT based application. *Internet of Things.* **11**, 100255 (2020). <https://doi.org/10.1016/j.iot.2020.100255>
13. A. Bano, I. Ud Din, A.A. Al-Huqail, AIoT-based smart bin for real-time monitoring and management of solid waste. *Sci. Program.* (2020). <https://doi.org/10.1155/2020/6613263>
14. S. Varudandi, R. Mehta, J. Mahetalia, H. Parmar, K. Samdani, A smart waste management and segregation system that uses internet of things, machine learning and android application, in *2021 6th International Conference for Convergence in Technology, I2CT 2021* (Institute of Electrical and Electronics Engineers Inc., 2021). <https://doi.org/10.1109/I2CT51068.2021.9418125>
15. K. Pardini, J.J.P.C. Rodrigues, O. Diallo, A.K. Das, V.H.C. de Albuquerque, S.A. Kozlov, A smart waste management solution geared towards citizens. *Sensors (Switzerland)*. **20** (2020). <https://doi.org/10.3390/s20082380>
16. R. Roshan, O.P. Rishi, Effective and efficient smart waste management system for the smart cities using Internet of Things (IoT): an Indian perspective. *Rising Threat. Expert Appl. Solut.* 473–479 (2021). https://doi.org/10.1007/978-981-15-6014-9_54
17. D. Baldo, A. Mecocci, S. Parrino, G. Peruzzi, A. Pozzebon, A multi-layer lorawan infrastructure for smart waste management. *Sensors.* **21** (2021). <https://doi.org/10.3390/s21082600>

A Novel Algorithm to Withstand Attacks on Blockchain Using Transaction History and Block Publishing Time



Anjaneyulu Endurthi, Aparna Pyarapu, Gayathri Jagiri,
and SaiSuma Vennam

Abstract Blockchain is an emerging technology with many kinds of cryptocurrencies like bitcoin, Ethereum etc. where one can earn rewards by creating new blocks through the process of mining. But how about only few getting all these rewards and trying to get control over the network? Does it pose a threat to blockchain integrity? Of course, yes which is leading to attacks such as 51% attack, selfish mining attack and double spending. Proof of work (PoW) is a protocol used in blockchain to reduce these problems, but it is not sufficiently secure. So, this paper proposes a technique, using history of miners and the history of their transactions that helps us to reduce the chances of these attacks. We have chosen history of transactions in order to find the genuineness of a miner, thus helping us to reduce double spending. Analysis shows that the risk of these attacks can be decreased using this technique.

Keywords Blockchain · Sybil attack · Selfish mining attack · Double spending · Proof of work · Cryptocurrency

1 Introduction

Today's generation is overflowing with technologies. Still, the craze over new technology is rising day by day. Not only there is growth in technology, but also there are a lot of privacy and security vulnerabilities. Presently, one of the new technologies is Blockchain [1]. Blockchain technology records information in the transparent digital ledger such that the information is immutable and cannot be compromised. Blocks in the blockchain are records of transactions joined by cryptographic hashes. Every block in the blockchain has a recent block hash value, timestamp and transactions data in it (Fig. 1).

Blockchain has a distributed ledger [2], i.e. data is used in a decentralized [3] way and can be managed by multiple participants (miners). A transaction is valid in blockchain only if more than 51% of participants in distributed network of blockchain approves it. Transactions are added to the block after mining and validation.

A. Endurthi (✉) · A. Pyarapu · G. Jagiri · S. Vennam

Department of Computer Science and Engineering, Rajiv Gandhi University of Knowledge Technologies—IIIT, Basar, Telangana, India

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
S. Smys et al. (eds.), *Inventive Computation and Information Technologies*, Lecture Notes
in Networks and Systems 336, https://doi.org/10.1007/978-981-16-6723-7_51

701

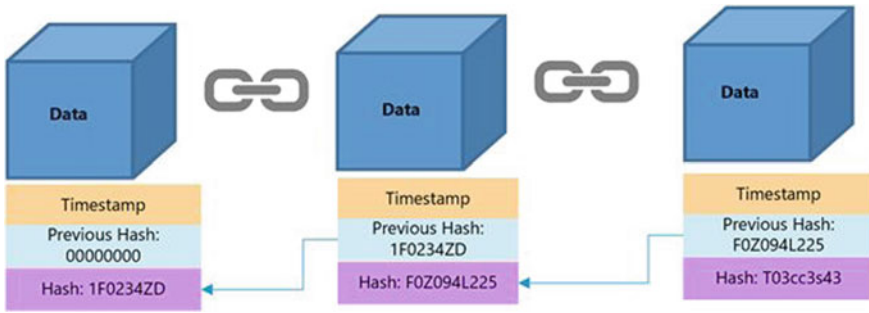


Fig. 1 Blockchain representation

1.1 Attacks on Blockchain

Even though blockchain is immutable, there are many attacks present as of now [4]. Blockchain network is secure and scalable, but if someone collectively in the network controls over 51% of the network, then there are chances of attacks on blockchain. The following sub-sections gives an idea of such attacks.

Double Spending It occurs when the attacker (Bob) would make some transaction between two other people (Lisa and Alice) and claim that he has sent the actual currency to both the people. But actually, he has sent digital currency to one person (Lisa) and a copy to other (Alice) or vice versa (Fig. 2).

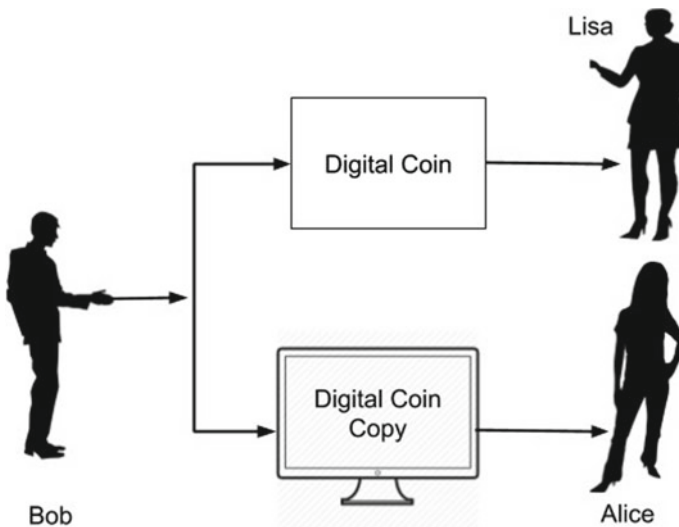


Fig. 2 Diagram displaying double spending



Fig. 3 Diagram displaying how 51% of the network attacking blockchain

51% Attack It is an attack on a blockchain by a miner or a group of miners who control more than 51% of the network’s mining power [5]. Thus, misusing the power by modifying the transactions of new blocks or including double spending transactions in the upcoming blocks. In the following figure, the red-coloured nodes are collectively working and has more than 51% hashing power (Fig. 3).

Sybil attack It is a kind of security threat where one person (attacker) tries to take over the network by creating multiple accounts, nodes and pretend as different people. Thus, trying to outvote the honest miners and rejecting the blocks those are created by honest miners (Fig. 4).

Selfish mining attack The selfish mining attack [6] or block withholding attack occurs when a miner decides to keep a valid block, that they have successfully mined, instead of broadcasting it on to the network, he/she mines the next blocks making his branch as a long chain and claim rewards by outrunning the existing valid blockchain [7] (Fig. 5).

In this paper, literature is discussed in the next section, i.e. Sect. 2 and proposed algorithm along with its advantages is discussed in Sect. 3. Section 4 consists of conclusions and future directions. References are provided at the end of the paper.

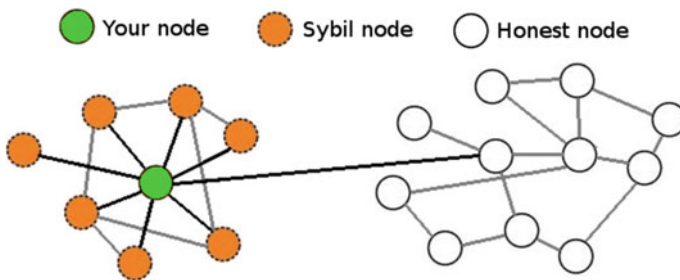


Fig. 4 Diagram showing Sybil attack

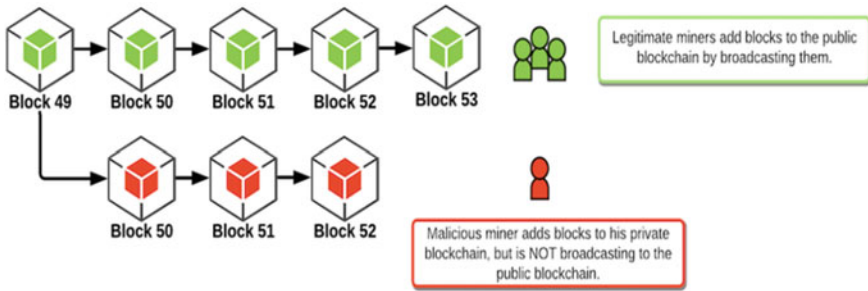


Fig. 5 Diagram of selfish mining attack

2 Literature Review

As there are many attacks that can occur on blockchain, there are many solutions proposed in the literature. Most of the solutions are specific to a particular kind of attack. To reduce block withholding or selfish mining attack, Siamak Solat, Maria Potop-Butucaru proposed ZeroBlock mechanism. In this, they added a timestamp. If miner creates a block, then he/she has to publish within the specified duration given by the network. If miners are unable to publish, then a dummy zero block is created which will be added to the blockchain [8]. This solves the selfish mining attack, but a greater number of unnecessary zero blocks will be added to the blockchain.

In order to overcome the drawback of ZeroBlock mechanism, a paper entitled Selfish Mining [9] was proposed. In this algorithm, a block has to be generated and propagated within maximum acceptance time (MAT). If no block is created within this MAT interval, then new target hash will be generated. All the blocks which were created based on old hash will not be accepted after the generation of new target hash.

To differentiate honest and dishonest branches, there is an algorithm called history weighted difficulty [10]. In this, they have calculated the frequency of each miner. If there is a split in the branches, history weighted difficulty of the branches is calculated based on frequency of miners and difficulty of the branch. Branch with higher history weighted difficulty value is selected.

Penalizing equivocation by loss of Bitcoins is proposed in [11]. In this paper, they proposed a technique to mitigate the behaviour of a double spending attacker. They are penalizing the one who is trying to double spend. This can be done in two ways; one way is to deposit some funds, and the other way is to extract the secret key of the one who has done double spending and use it to force the loss of attacker's funds.

The concept of transaction creation time was given as a solution for one of the attacks in [12]. In this paper, they have proposed fork protocol to differentiate honest and dishonest miners. They have changed the block structure by adding one field to the block that is transaction created time. Using this, we can calculate the average transaction time. The miners with less average transaction time were called as honest miners.

The paper titled Majority is not enough Bitcoin mining is vulnerable [13]. In this, they have proved that the existing bitcoin protocol is not incentive-compatible. It means that the selfish miners are getting more rewards than their computational power. So, they have modified the bitcoin protocol to encourage the honest miners. By this protocol, network will consist of 2/3 honest miners.

A backward compatible defence against selfish mining in bitcoin is discussed in [14]. In this, they have introduced fork-resolving policy weighted FRP and an upper bound 'T' on block propagation time which is known as In-time. A block A1 is considered to be the uncle of another block A2 if A1 is a competing in time block of A2's parent block. Weight of the chain is calculated based on in-time blocks and in-time uncle hashes. So, when there are competing chain of blocks, weighted FRP is calculated; if one chain is less than other chain by 'k' blocks, the chain with highest weight is chosen by the miner.

Another paper named Disincentivize large bitcoin mining pool, which they have proposed two-phase proof of work [15] to control the formation of large mining pools. Process involves two steps: (1) Double hash of the header. (2) Hash over header which is signed with private key of the miner, i.e. SHA256 (SIG header, private key).

Another paper entitled Weird trick to stop selfish mining [16] proposed a technique to stop selfish mining. In this, for every 's' seconds, it publishes a new random value 'R'. 'R' is unpredictable before the publication, and they both form a tuple (R, T) and verified by anyone. Where, 'R' is the randomness and 'T' is the timestamp. If any block whose 'R' is before the timestamp, 'T' is considered as honest block and if it is after the timestamp or if beacon is compromised by an attacker, then it is considered as dishonest block.

Each of these solutions in the literature gives solutions to some specific attacks could not be able to withstand other attacks. This motivated us to propose an algorithm, which can withstand against more than one kind of attack. The proposed algorithm is discussed in the next section.

3 Proposed Algorithm

This section describes about minimizing attacks on blockchain using history of transactions and selfish mining strategy. The main motto of this algorithm is to detect a selfish miner and to differentiate honest and dishonest branches based on miner's frequency in the history and history of their transactions.

Selfish miner can be identified by comparing the time he/she is trying to publish the block and his/her block creation time. This paper concentrates on setting a time limit to publish the newly mined block. If the miner has published his block within this time limit, his/her block will be accepted, else it will be rejected.

Double spending can be reduced using this algorithm as we are disposing a constraint on the transactions of miner's and trying to calculate how malicious a miner is based upon his/her valid and invalid transactions. So, if the miners tend to make invalid transactions, they are treated as malicious and thus their mining branch

would be discarded. As we are discarding the malicious branch, attacker cannot have control over the network and thus withstanding against 51% attack.

3.1 Algorithm

The following algorithm is divided into three steps. The second step is executed only when the first step is cleared and step 3 is executed only when it clears step 2.

Step 1—Check if the miner is selfish

Time limit for newly mined block to publish = “T”.

```

If(there is branch split)
    Calculate HWD and HTV for all the nodes present in the
    branches(Where, HWT – History Weighted difficulty and HVT
    - History Transaction Value)

    HW = 0
    HTV=0
    d = 0
    w = Length (W)  (Where W - array of historic blocks window)
    l = Length (B)  (Where B - array of branch blocks)
    Let R[1 ...l] be new arrays
    Let Tr[1...l]=0 -> be transaction history of an individual miner
    in the branch
    for i = 1 to l do=>Calculate miner appearance frequency in
    historic blocks window and calculate transaction history of
    miner
        R[i]=0
        Tr[i]=Number of invalid transactions of a miner/Total
        number of transactions of a miner
        for j = 1 to w do
            if( Miner(W[j]) == Miner(B[i]) then
                R[i]+ = 1
        R[i]/ = w
    for k = 1 to Length(B) do=>Sum Historic Weight
        HW = HW + R[k]
    for k = 1 to Length(B) do =>Sum branch difficulty
        d = d + Diff(B[[k])
    for k = 1 to Length(B) do =>sum transaction history
        HTV=HTV+Tr[k]
    HWD = HW * d
  
```

Step 2—If newly mined block is accepted

If(there is branch split)

Calculate HWD and HTV for all the nodes present in the branches(Where, HWT – History Weighted difficulty and HVT - History Transaction Value)

HW = 0

HTV=0

d = 0

w = Length (W) (Where W - array of historic blocks window)

l = Length (B) (Where B - array of branch blocks)

Let R[1 ...l] be new arrays

Let Tr[1...l]=0 -> be transaction history of an individual miner in the branch

for i = 1 to l **do**=>Calculate miner appearance frequency in historic blocks window and calculate transaction history of miner

R[i]=0

Tr[i]=Number of invalid transactions of a miner/Total number of transactions of a miner

for j = 1 to w **do**

if(Miner(W[j]) == Miner(B[i]) then

R[i]+ = 1

R[i]/ = w

for k = 1 to Length(B) **do**=>Sum Historic Weight

HW = HW + R[k]

for k = 1 to Length(B) **do** =>Sum branch difficulty

d = d + Diff(B[[k])

for k = 1 to Length(B) **do** =>sum transaction history

HTV=HTV+Tr[k]

HWD = HW * d

Step 3—Compare the HWD and HTV values of the branches

If(HTV is high and HWD is high)
Do not accept this branch
else if(HTV is high and HWD is low)
Do not accept this branch
else if(HTV is low and HWD is high)
Continue mining from this branch
else if(HTV is low and HWD is high)
Continue mining from this branch

3.2 Flow Chart Representation:

Following is the flow chart representation of the algorithm (Fig. 6).

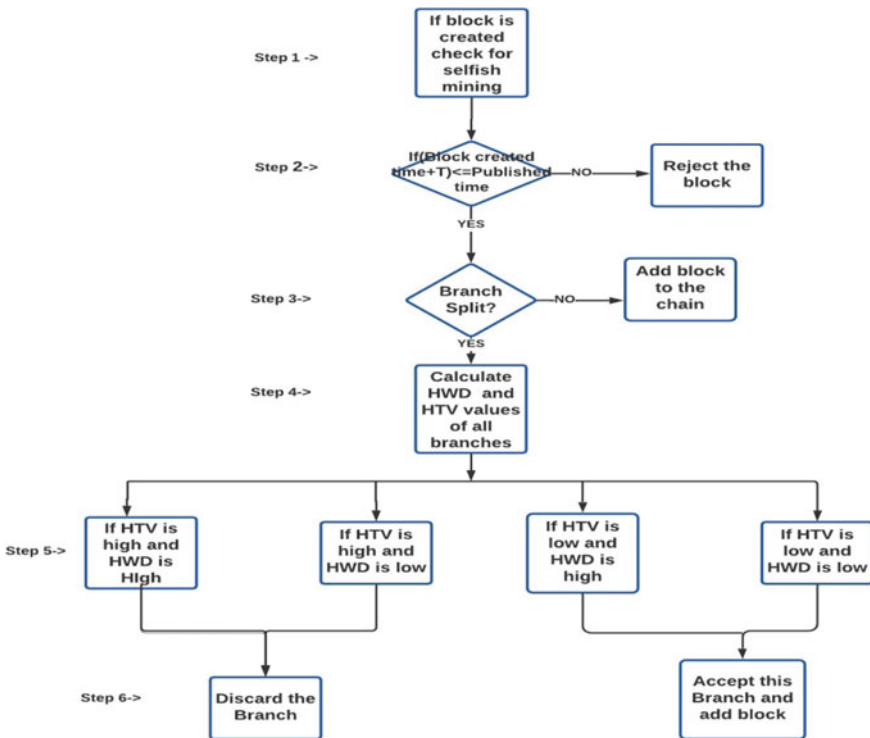


Fig. 6 Flow chart representation of algorithm

If any block is mined recently, it should be checked whether it is selfishly mined or not. And then honest and malicious branches should be differentiated, if there's more than one branch. History weighted difficulty (HWD), transaction history value (HTV) should be calculated for all the branches. Based on those values, branch should be either accepted or discarded, and newly mined block will be added to honest blockchain branch.

3.3 Advantages of Proposed Algorithm

All the existing algorithms of selfish mining have efficacy to some extent and possesses drawbacks. But in the proposed algorithm, we can reduce selfish mining to greater extent as we are making miner to publish block soon after its creation and also restricting the miners based on their transaction history. Moreover, we can successfully differentiate honest and dishonest branches which can withstand against attacks like 51% attack, selfish mining attack and also double spending attack.

4 Conclusion and Future Directions

This paper proposes an approach to reduce the chance of 51% attack and selfish mining attack on proof of work-based blockchain protocols. This approach makes use of frequency of miners and history of their transactions and determines if a branch switch is needed or not. It also detects selfish miner based upon their block publishing time. Thus, making it withstand against dangerous attacks on blockchain. In future, the same solution could be extended to withstand other attacks.

References

1. S. Nakamoto, *Bitcoin: A Peer-to-Peer Electronic Cash System* (2008)
2. R. Wattenhofer, *Distributed Ledger Technology—The Science of Blockchain* (Forest Publishing, 2017)
3. M. Atzori, *Blockchain Technology and Decentralized Governance: Is the State Still Necessary* (2015)
4. M. Saad, J. Spaulding, L. Njilla, C. Kamhoua, S. Shetty, D. Nyang, et al., Exploring the Attack Surface of Blockchain: A Comprehensive Survey. *IEEE Commun. Surveys Tuts.* **22**(3), 1977–2008 (2020)
5. D. Mories, *51% Attack Are Growing a Threat to Smaller Blockchain; Komodo May Be the Solution* (2019)
6. S. Zhang, K. Zhang, *Bettina Kemme Analysing the Benefit of Selfish Mining with Multiple Players* (2020)
7. V. Chicarino, E.F. Jesus, C. Albuquerque, A. Rocha, A heuristic for the detection of selfish miner and stalker attacks in blockchains networks, in *IEEE Blockchain, Robotics and AI for Networking Security Conference, 2019, Rio de Janeiro*. BRAINS, IEEE, pp 1–6 (2019)

8. S. Solat, M. Potop-Butucaru, *Zeroblock: Preventing Selfish Mining in Bitcoin*. arXiv preprint [arXiv:1605.02435](https://arxiv.org/abs/1605.02435) (2016)
9. A. Endurthi, B. Dastagiri, S. Janipasha, D. Santhosh, A. Khare, *Transformed Puzzle for Preventing Selfish Mining: A Non-Viable Way to Defend Zero Block Algorithm* (2020)
10. X. Yang, Y. Chen, X. Chen, *Effective Scheme Against 51% Attack on Proof-of-Work Blockchain with History Weighted Information* (2019)
11. T. Ruffing et al., Liar, liar, coins on fire!: Penalizing equivocation by loss of Bitcoins, in *ACM Conference on Computer and Communications Security* (2015)
12. J. Lee, Y. Kim, *Preventing Bitcoin Selfish Mining Using Transaction Creation Time* (2018)
13. I. Eyal, E.G. Sirer, Majority is not enough: Bitcoin mining is vulnerable, in *International conference on financial cryptography and data security* (Springer, Berlin, Heidelberg, 2014)
14. R. Zhang, B. Preneel, Publish or perish: a backward-compatible defense against selfish mining in bitcoin, in *Cryptographers' Track at the RSA Conference* (Springer, Cham, 2017)
15. I. Eyal, E.G. Sirer, *How to Disincentivize Large Bitcoin Mining Pools* (2014)
16. E. Heilman, One Weird Trick to Stop Selfish Miners: Fresh Bitcoins, a Solution for the Honest Miner, in *International Conference on Financial Cryptography and Data Security* (Springer, Berlin, Heidelberg, 2014)

Mental Health Prediction Using Data Mining



I. G. Hemanandhini and C. Padmavathy

Abstract Mental illness is a condition that affects the behaviour, attitude and mannerisms of a person. They are highly common in these days of isolation due to the on-going pandemic. Almost 450 million people worldwide suffer from some kind of mental illness. Mental health problems do not only affect adults, but also it has significant impact on kids and teenagers. It is totally normal and understandable to experience fear during the time of COVID-19 pandemic. Loneliness, isolation, unhealthy alcohol and substance usage, self-harm or suicidal behaviour are all projected to escalate as new policies and impacts are implemented, especially quarantine and its effects on many people's usual habits, schedules or livelihoods. Furthermore, psychiatric disorders have become one of the most severe and widespread public health issues. Early diagnosis of mental health issues is critical for further understanding mental health disorders and providing better medical care. Unlike the diagnosis of most health diseases, which is dependent on laboratory testing and measures, psychiatric disorders are usually classified based on a person's self-report of detailed questionnaires intended to identify specific patterns. The project would use a person's tweets, a few customized questions and answers, and a few personal data to measure a person's mental well-being ranking. This initiative would be immensely helpful to anyone who uses social media sites on a regular basis in order to live a stress-free life and diagnose mental health problems before they get too serious.

Keywords Mental illness · Psychiatric disorders · Mental well-being · COVID-19

I. G. Hemanandhini (✉) · C. Padmavathy
Department of Computer Science and Engineering, Sri Ramakrishna Engineering College,
Coimbatore, India
e-mail: hemanandhini@srec.ac.in

C. Padmavathy
e-mail: padma.dhanush@srec.ac.in

1 Introduction

COVID-19 has resulted in quarantining or isolating as a new trend that has a very passive effect on individuals. Individuals may develop thoughts such as depression, anxiety and suicidal ideation as a result of this. Workplaces and educational institutions are doing their best to address the situation, but it is insufficient. The challenge is to develop a model that can accurately predict an individual's mental health state and assist in monitoring and curing it at an earlier stage.

The primary goal of this paper is to anticipate a person's mental well-being using social media platforms, specifically Twitter. Based on the individual's tweets, this paper will predict a mental health score. It will be extremely beneficial to those who use social media platforms on a daily basis and will assist them in monitoring their mental health in order to live a stress-free life.

2 Related Work

This section briefs about the related works carried out for predicting a person's mental well-being via social platforms.

S. E. Jordan et al. conducted a survey dictating the use of Twitter data for predicting public health. Here, various methods were used for mining the Twitter data for public health monitoring. Research papers where Twitter data is classified as users or tweets are considered for the survey for monitoring the health of persons in a better way. Also, papers published from 2010 to 2017 were taken for conducting survey. The approaches used to categorize the Twitter content in many ways are distinguished. While it is difficult to compare research, since there are so many various ways for using Twitter and interpreting data, this state-of-the-art review highlights the huge potential of using Twitter for public health surveillance.

Heiervang et al. conducted a structured psychiatric interview for the parents for predicting the child mental state. Parents were interviewed face to face in 2003, and they finished the interview online in 2006. Interviews were preceded by printed questionnaires covering child and family variables in both surveys. Web-based surveys can be completed more quickly and at a lesser cost than traditional methods including personal interviews. Point estimates of psychopathology appear to be particularly vulnerable to selective participation although patterns of connections appear to be more durable.

3 Proposed System

Coronavirus has crossed the globe, isolating or disengaging numerous people, bringing about antagonistic psychological well-being impacts for some like uneasiness, melancholy, self-destruction and self-hurt. Working environments/educational establishments that encourage mental prosperity and help individuals with mental incapacities are bound to limit non-attendance, improve profitability and receive the expert and individual rewards that accompany it. The test is to make a model that will foresee the emotional wellness of people and accordingly help psychological well-being suppliers to convey forward with the treatment in this period of scarcity.

These days, most of the mental health problems are identified and treated at later stage. We propose a unique technique to mental health detection using user tweets as an early discovery system to actively identify probable mental health situations. A machine learning framework has been developed for finding a person's mental well-being. We analyse a person's tweets, along with a few of their personal details and predict whether or not the person should see a therapist based on a series of quizzes. The proposed approach employs naïve Bayes and linear regression techniques to find a person's mental health by means of their tweets. To improve efficiency, we perform a quiz analysis with a decision tree algorithm and predict the scores.

4 Algorithm Description

4.1 Naïve Bayes Algorithm

It is a method based on Bayes' theorem and the assumption that indicators are autonomous. A naïve Bayes classifier, in simple terms, assumes that the presence of one variable in a class has no influence on the presence of another. For example, if a natural product is red, oval and around 3 creeps across, it is considered an apple. Regardless of whether these characteristics rely on one another or on the existence of other characteristics, they all contribute to the probability that this natural commodity is an apple, which is why it is regarded as 'Credulous'. The Bayes model is simple to construct and is particularly useful for extremely large informative indexes. Along with its simplicity, naïve Bayes is considered to outperform even the most sophisticated order techniques. From $P(c)$, $P(x)$ and $P(x|c)$, the Bayes hypothesis provides a method for determining back probability $P(c|x)$.

4.2 Linear Regression Algorithm

Linear regression is an AI calculation that is based on learned data. It performs a relapse simulation. A regression model is used to model an objective expectation

esteem that is contingent on free variables. It is primarily used for evaluating and exploring the relationship between variables. Different relapse models differ in terms of the type of relationship; they consider between reliant and autonomous factors as well as the number of free factors they employ.

Linear regression enacts the task of predicting the value of a dependent variable (y) in light of a given autonomous variable (x). In this way, this relapse protocol discovers a direct link between x (input) and y (output). Linear regression is the name given to it as a result.

4.3 Decision Tree

The decision tree algorithm is part of the supervised learning algorithms family. The decision tree algorithm, unlike other supervised learning algorithms, can also be used to solve regression and classification problems.

The global variables we have determines the different sorts of decision trees we have. There are two forms of it:

Categorical Variable Decision Tree: A categorical variable decision tree is a decision tree with a categorical target variable.

Continuous Variable Decision Tree: A continuous variable decision tree is one that has a continuous focus variable (Fig. 1).

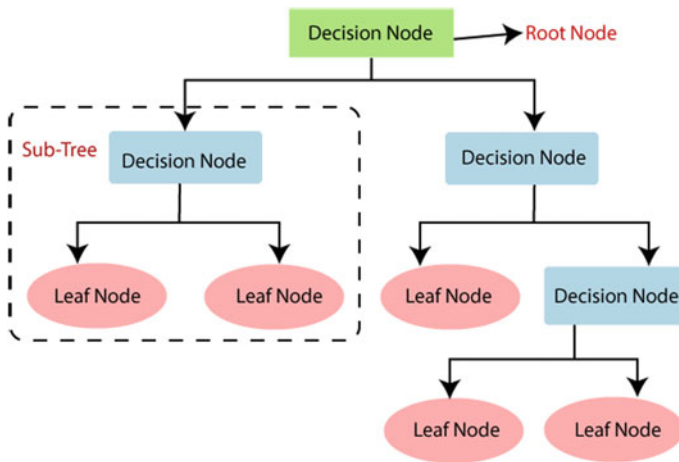


Fig. 1 Decision tree

5 Implementation

Today, psychological wellness is anticipated at a later stage. To effectively distinguish expected psychological well-being by mining information logs of online media clients as an early discovery framework, we present a novel way for recognizing emotional well-being. We foster an AI structure to distinguish emotional wellness. The proposed approach can be communicated to provide early notification of anticipated patients. We analyse the client's tweets and apply naïve Bayes and linear regression calculations to get the clients OCEAN examination, i.e. openness, conscientiousness, extraversion, agreeableness and neuroticism. We utilize a test investigation to become familiar with the client and a choice tree calculation to figure their emotional wellness score. Alongside these two qualities, we get some close to home data, and the emotional wellness score is anticipated, alongside a message showing whether the individual should see a therapist.

5.1 Module Description

1. **OCEAN Analysis:** In this module, we take the tweets of the users as the datasets and process the datasets to get the OCEAN (openness, conscientiousness, extraversion, agreeableness, neuroticism) analysis of the user. The tweets are hence cleaned and algorithms such as naïve Bayes and linear regression are performed to calculate the emotion of the tweets.
2. **Quiz Analysis:** The proposed quiz analysis consists of 20 different customized questions that will help us to give a clearer analysis of the mental state. We apply a decision tree algorithm to the answers and predict the score.
3. **Prediction of Mental Health:** Here, we predict the final score using both the scores obtained from ocean analysis and quiz analysis. We suggest depending on the score whether a person needs to consult the therapist or not.

5.2 System Architecture

See Fig. 2.

6 Result and Discussions

In this paper, we used different algorithms to try to determine the mental health of people who use social media sites, mostly Twitter. We successfully predicted the user's mental health score and recommended whether or not the individual should see a therapist. Using naïve Bayes, linear regression and decision tree on their tweets, we

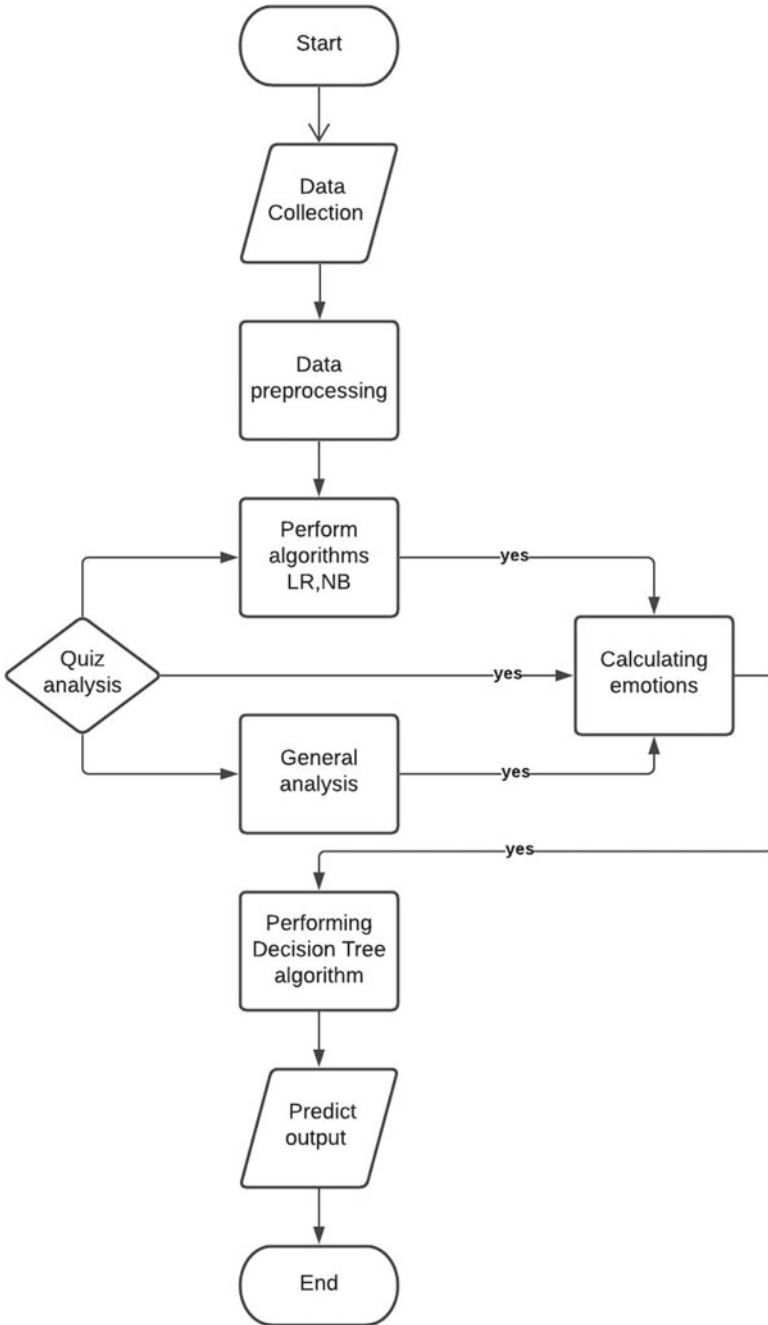


Fig. 2 System architecture

performed OCEAN analysis and got an output with greater accuracy. Another such output was predicted using a survey with several questions based on the candidate’s behaviour. Personal details like work scenarios and family history were taken into account as well. These outputs were then added and taken average of. The following result is more accurate than the previous works based on various parameters (Figs. 3, 4, 5, 6, 7, and 8).

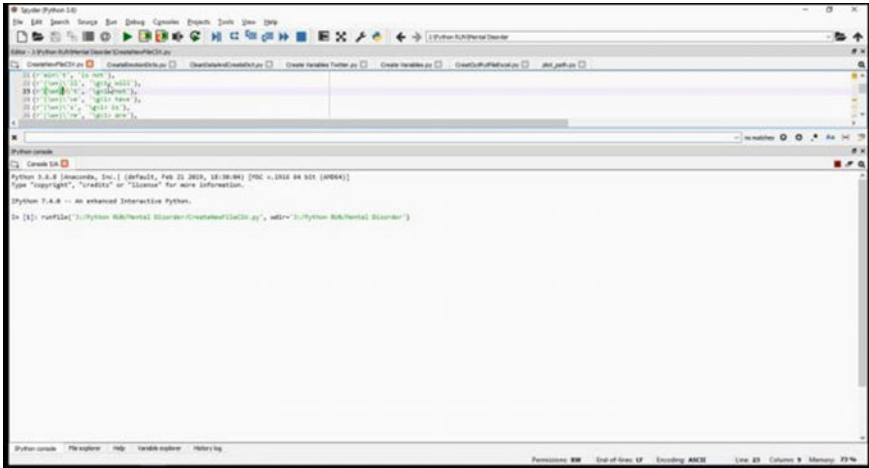


Fig. 3 Data pre-processing

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	ID	Post	Months	Hours	anger	anticipation	disgust	fear	joy	openness	conscienti	extraversio	agreeabler	neuroticisr
2	b7b7764cf	likes soum	3	15	0	0	0	0	0	0	0	0	0	0
3	b7b7764cf	sleepy not	3	08	0	0	0	0	0	0	0	0	0	0
4	b7b7764cf	sore wants	3	13	2	0	0	0	1	0	2	0	2	0
5	b7b7764cf	likes day s	3	04	0	0	0	0	0	0	0	0	0	0
6	b7b7764cf	home 3	3	02	0	0	0	0	0	0	0	0	0	0
7	b7b7764cf	wwwthejok	3	15	0	0	0	0	0	0	0	0	0	0
8	b7b7764cf	saw nun zc	3	05	0	0	0	0	0	1	0	0	0	0
9	b7b7764cf	kentucky 4	3	06	0	1	0	1	1	0	1	0	0	0
10	b7b7764cf	fmish digti	3	14	1	0	1	0	2	1	0	1	0	1
11	b7b7764cf	celebrating	3	23	0	1	0	0	2	0	2	1	0	0
12	b7b7764cf	crush gree	3	19	0	0	0	0	1	0	1	0	0	0
13	b7b7764cf	magic brai	3	04	0	0	0	0	0	0	0	0	0	0
14	b7b7764cf	saw transf	3	04	0	1	0	0	1	1	1	1	1	1
15	b7b7764cf	wants mee	3	03	0	0	0	0	0	0	0	0	0	0
16	b7b7764cf	desires thr	3	21	1	3	0	1	3	0	3	0	2	1
17	b7b7764cf	going bed	3	01	0	0	0	0	0	0	0	0	0	0
18	b7b7764cf	reading ad	3	22	0	0	0	0	0	0	1	0	0	0
19	b7b7764cf	thinks inta	3	02	0	0	0	0	0	1	0	0	0	0
20	b7b7764cf	tired let go	3	05	0	0	0	0	0	0	1	0	0	0
21	b7b7764cf	discovering	3	06	0	0	0	0	0	0	0	0	0	0
22	b7b7764cf	watching c	3	04	0	1	0	1	0	0	0	0	0	0
23	b7b7764cf	getting urg	3	00	0	0	0	0	0	0	0	0	0	0
24	b7b7764cf	woulda tho	3	03	0	1	0	1	0	0	0	0	0	0
25	b7b7764cf	wishes dev	4	02	0	1	0	1	0	0	1	0	0	0
26	b7b7764cf	tell draw pl	3	09	0	0	0	0	0	0	0	0	0	0
27	b7b7764cf	found bunn	3	19	0	0	0	2	1	2	1	2	0	1
28	b7b7764cf	3	4	23	0	0	0	0	0	0	0	0	0	0
29	b7b7764cf	insane	4	21	1	0	0	1	0	1	0	0	0	0
30	b7b7764cf	wants slee	4	01	0	0	0	0	0	0	0	0	0	0
31	b7b7764cf	really hate	4	03	2	1	2	2	1	2	1	2	1	1
32	b7b7764cf	watch mat	3	04	0	2	0	1	-1	0	0	0	0	0
33	b7b7764cf	loved 9 alk	4	02	0	0	0	0	1	0	2	0	0	0
34	b7b7764cf	not sit futu	4	01	-1	0	-1	0	0	0	0	-1	0	0
35	b7b7764cf	spend less	3	06	0	1	0	0	0	0	2	0	0	1
36	b7b7764cf	super-brope	3	11	0	1	0	1	0	0	0	0	0	0

Fig. 4 Calculating emotions

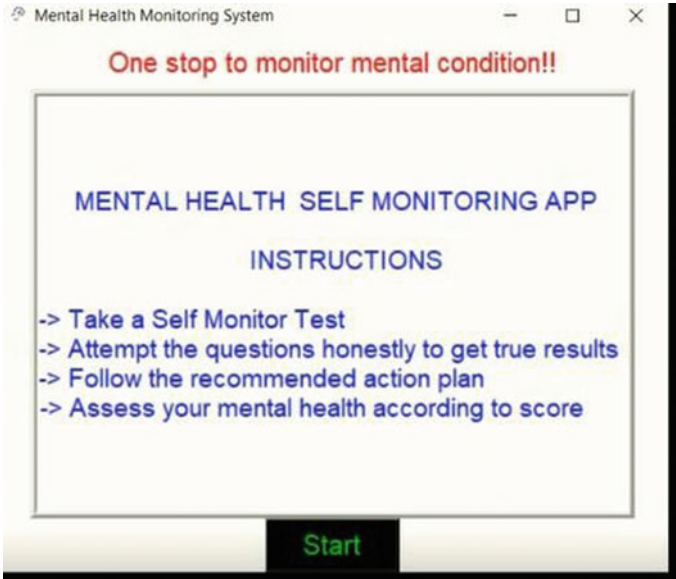


Fig. 5 Quiz analysis

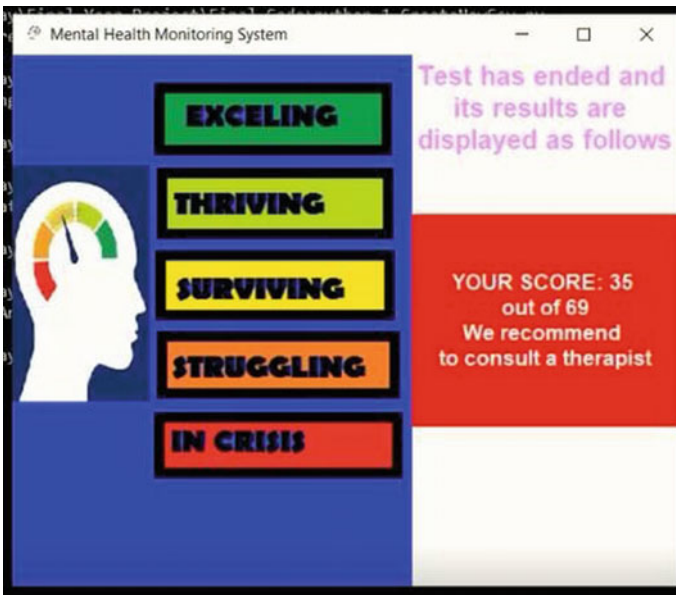


Fig. 6 Quiz analysis score

```
C:\Windows\system32\cmd.exe
aInput : 1
Self Employed --- Yes - 0 No - 1
nInput : 0
rFamily History --- Yes - 0 No - 1
nInput : 1
Remote Work --- Yes - 0 No - 1
tInput : 0
SWork Interface --- Yes - 0 No - 1
atInput : 0
BASED ON OCEAN ANALYSIS
Person Doesn't Need Medical Treatment
C
BASED ON GENERAL INFORMATION
DePerson Needs Medical Treatment
le
FINAL RESULT
ctFinal Score : 50.00%
ur50.0
nYou May need to visit a Therapist
n
c
-----
o
Press any key to exit_
```

Fig. 7 Personal details

```
C:\Windows\system32\cmd.exe
E:\Sujay\Final Year Project\Final Code>python 1_CreateNewCsv.py
Data PreProcessing
E:\Sujay\Final Year Project\Final Code>python 2_CreateEmoDict.py
Creating Emotional Dictionary
E:\Sujay\Final Year Project\Final Code>python CleanDataAndCreateDict.py
E:\Sujay\Final Year Project\Final Code>python 4_CreateVarTwitter.py
Calculating Emotion
E:\Sujay\Final Year Project\Final Code>python 5_CreateVar.py
E:\Sujay\Final Year Project\Final Code>python 6_CreateOPFile.py
Ocean Analysis done for 9 tweets
E:\Sujay\Final Year Project\Final Code>python 7_app.py
E:\Sujay\Final Year Project\Final Code>python 8_Final.py
Gender --- Male - 1 Female - 0
Input : 1
Self Employed --- Yes - 0 No - 1
Input : 0
Family History --- Yes - 0 No - 1
Input : 1
```

Fig. 8 Mental health prediction

References

1. K. Young, M. Pistner, J. O'Mara, J. Buchanan, *Cyberpsychol. Behav.* **2**(5), 475–479 (1999). <https://doi.org/10.1089/cpb.1999.2.475> (PMID: 19178220)
2. Y. Mehta, N. Majumder, A. Gelbukh, E. Cambria, Recent trends in deep learning based personality detection. *Artif. Intell. Rev.* **53**, 2313–2339 (2020). <https://doi.org/10.1007/s10462-019-09770-z>
3. D. Xue, Z. Hong, S. Guo, L. Gao, L. Wu, J. Zheng, N. Zhao, Personality recognition on social media with label distribution learning. *IEEE Access.* <https://doi.org/10.1109/ACCESS.2017.2719018>
4. J. Block, Issues of DSM-V: internet addiction. *Am. J. Psychiatr.* **165**(3), 306–307 (2008). <https://doi.org/10.1176/appi.ajp.2007.07101556> (PMID: 18316427)
5. K.S. Young, Internet addiction: the emergence of a new clinical disorder. *Cyber Psychol. Behav.* **1**, 237–244 (1998). <https://doi.org/10.1089/cpb.1998.1.237>
6. I.-H. Lin, C.-H. Ko, Y.-P. Chang, T.-L. Liu, P.-W. Wang, H.-C. Lin, M.-F. Huang, Y.-C. Yeh, W.-J. Chou, C.-F. Yen, The association between suicidality and Internet addiction and activities in Taiwanese adolescents. *Compr. Psychiat.* (2014)
7. Y. Baek, Y. Bae, H. Jang, Social and parasocial relationships on social network sites and their differential relationships with users' psychological well-being. *Cyberpsychol. Behav. Soc. Netw.* (2013)
8. D. La Barbera, F. La Paglia, R. Valsavoia, Social network and addiction. *Cyberpsychol. Behav.* (2009)
9. K. Chak, L. Leung, Shyness and locus of control as predictors of internet addiction and internet use. *Cyberpsychol. Behav.* (2004)
10. K. Caballero, R. Akella, Dynamically modeling patients health state from electronic medical records a time series approach. *KDD* (2016)
11. L. Zhao, J. Ye, F. Chen, C.-T. Lu, N. Ramakrishnan, Hierarchical Incomplete multi-source feature learning for Spatiotemporal Event Forecasting. *KDD* (2016)
12. E. Baumer, P. Adams, V. Khovanskaya, T. Liao, M. Smith, V. Sosik, K. Williams, Limiting, leaving, and (re)lapsing: an exploration of Facebook non-use practices and experiences. *CHI* (2013)
13. S.E. Jordan, S.E. Hovet, I.C.-H. Fung, H. Liang, K.-W. Fu, Z.T.H. Tse, *Using Twitter for Public Health Surveillance from Monitoring and Prediction to Public Response*. *Big Data and Digital Health*
14. E. Heiervang, R. Goodman, Advantages and limitations of web-based surveys: evidence from a child mental health survey. *Soc. Psychiat. Epidemiol.* **46**, 69–76 (2011). <https://doi.org/10.1007/s00127-009-0171-9>

Decision Rules Generation Using Decision Tree Classifier and Their Optimization for Anemia Classification



Rajan Vohra, Anil Kumar Dudyala, Jankisharan Pahareeya,
and Abir Hussain

Abstract Anemia disease is one of the prevalent diseases observed across women and children in most of the developing countries. This is caused due to the iron deficiency in human body. Detecting this disease at the early stage can help the medical fraternity to prescribe proper medicines and come up with alternate solutions that can prolong the patient's initial stage before it enters into critical stage. Due to the non-availability of historical data of the Anemia patients, there is very sparse literature that addresses the problem of detection of this disease. In this paper, a real-time Anemia dataset pertaining to Indian context is considered and due to the imbalance nature of the dataset, SMOTE is employed for balancing. With the help of decision tree rule-based learning method, rules for detecting the Anemia are derived using two datasets original and SMOTE. The efficacy of the rules is evaluated, and reduced rules are selected based on their individual classifying accuracy. In a quest to give a simple human understandable and optimal rule which can be used by any medical fraternity for detecting the presence of Anemia at different stages, we tried to propose the reduced rules-based method which may come handy. The efficacy of the rules is promising and helps in identifying the presence of Anemia at early stage.

Keywords Decision tree (DT) · Reduced rules · Anemia detection · Data balancing

R. Vohra (✉) · A. Hussain

Department of Computer Science, Liverpool John Moores University, Liverpool, UK

e-mail: R.vohra@ljmu.ac.uk

A. Hussain

e-mail: A.Hussain@ljmu.ac.uk

A. K. Dudyala

Department of Computer Science, National Institute of Technology Patna (NIT Patna), Patna,
India

e-mail: ak@nitp.ac.in

J. Pahareeya

Department of Information Technology, Rustamji Institute of Technology, BSF Academy,
Tekanpur, Gwalior, India

e-mail: jankisharan@rjit.org

1 Introduction

Human body is composed of several proteins and amino acids. The sustenance of this body is carried out with the help of minerals and vitamins. Deficiency of any of these essential minerals or vitamins will cause either malfunction of the human body or will lead to disease. One such deficiency of iron in human body will lead to a disease called Anemia. Anemia can be termed as deficiency of hemoglobin caused due to shortage of iron. Anemia is found to be prevalent among the developing countries and most popularly among women and children compared to men of these countries. Globally it is observed as one of the critical health problems.

Identifying the Anemia at the early stage so that it can be prolonged from further deteriorating into advanced stages is one of the most challenging issues. This is due to the non-availability of the data in real-time scenario. It is observed in the literature very few works that have explored this issue of detecting the anemia [1–5]. Hence, we have considered the Indian dataset which had been collected from Eureka diagnostic center, Lucknow, India [6] for the experimental purpose and with the help of Decision Tree model a set of rules has been generated that helps in detecting the anemia at the early stages.

The remaining sections are arranged as follows, Sect. 2 describes related work. Proposed method is discussed in Sect. 3. Section 4 deals with the data description and algorithm used. Results and discussion are elaborated in Sect. 5, while Sect. 6 concludes the work.

2 Related Work

Anemia that is caused by the deficiency of Iron is one of the most critical health problems globally and is a serious public health issue [7]. According to the World Health Organization (WHO), Anemia prevalence of over 40% in a community makes it a public health issue of critical importance [8]. While Anemia prevalence in children can be caused due to genetic reasons or due to deficiencies in nutrition like deficiencies in iron or folate or vitamins A/B12 and copper, iron deficiency is the most important determinant of anemia [9]. Socio demographic characteristics of mothers, households such as region, wealth index, water sources, working status and anemia status along with child features like age, nutritional status, child size at birth are the most critical features influencing anemia in the age group of 6–59 months in children [1]. According to WHO, Anemia prevalence occurs in most of the countries in Africa and South Asia and some countries in East Asia and the Pacific. While the highest prevalence of anemia is found in Africa, the largest numbers of children affected by anemia are found in Asia [2]. Many machine learning models are increasingly used in the analysis and prediction of diseases in the healthcare [10]. Most of studies indicated that machine learning techniques such as support vector machines (SVM), Random Forest and artificial neural networks (ANN) have been applied for

the classification of different diseases such as Diabetes [11–13], Appendicitis [14], and multiple sclerosis [15]. Machine learning techniques to classify anemia in children are still evolving. Along with traditional clinical practices, machine learning techniques can be utilized to predict the risk of anemia prevalence in children. Some key research in this direction has been undertaken as demonstrated in [3, 16], which have constructed prediction models for anemia status in children. The prevalence of anemia among adults was studied by taking complete blood count (CBC) at a referral hospital in southern Ethiopia. Prevalence and severity were related with age and gender and were analyzed [17]. Social factors such as income, wealth, education can affect health markers in people such as blood pressure, body mass index (BMI), and waist size, etc. [18]. Sow et al. used support vector machines (SVM) and demographic health survey data from Senegal to classify malaria and anemia status accurately [4, 19]. Using feature selection, the number of features of both anemia and malaria datasets were reduced. Using variable importance in projection (VIP) scores, the relative importance of social determinants for both anemia and malaria prevalence were computed. Finally, machine learning algorithms were utilized for the classification of both anemia and malaria—Artificial neural networks (ANN), K nearest neighbors (KNN), Random Forests, Naïve Bayes and support vector machines (SVM) were used [20]. Lisboa has demonstrated the utility and potential of Artificial Neural Networks (ANN) in health care interventions [5]. Using CBC samples, a study to classify anemia using Random Forests, C4.5 (Decision tree), and Naïve Bayes (NB) was undertaken. Comparison of the classifying algorithms using mean absolute error (MAE) and classifier accuracy were computed and tabulated in [21]. Some of the research also applied the Naïve Bayes Classifier and entropy classifier for the purpose of classification [6].

Almugren et al. in 2018 conducted a study using the anemia dataset and investigated how Artificial neural networks (ANN), Naïve Bayes (NB), C4.5 and Jrip data mining algorithms can be used to classify instances in the given dataset as being anemic or normal—that is a binary classification problem. In this study, the performance of these algorithms was benchmarked for a comparative analysis, and it was found that ANN and Jrip algorithms were the best performing algorithms in this regard [22]. In a study, Jatoi et al. used data mining methods on complete blood count (CBC) dataset of 400 patients for detecting the presence of anemia. It was found that Naïve Bayes (NB) algorithm had 98% accuracy in predicting the presence of the disease correctly [23]. In the study conducted in 2019, Meena et al. have used Decision tree algorithms to perform classification on an input dataset representing children for the diagnosis of anemia in the given dataset. They also identified the significant features driving the prevalence of anemia in reference to the feeding practices adopted for infant feeding [24]. Ching Chin Chern et al. have used Decision Tree Classifier models to acquire decision rules for classifying eligibility of Taiwanese citizens to be suitable recipients of tele health services. Involvement of a physician, social worker and health care managers was done to ensure a thorough process and J48 algorithm and logistic regression techniques were used to generate the decision trees representing the decision rules generated [25]. A study done by Lakshmi K.S et al. has used Association rule mining on medical records to

extract decision rules of the type symptom disease. In this computation well-known Association rule mining algorithms like A Priori and FP Growth have been used to derive the decision rules [26]. Song Yan et al. have studied how decision trees can be used to generate decision rules for various medical conditions. In this paper, they have constructed a Decision tree model representing decision rules for the classification and diagnosis of Major Depressive disorder (MDD) [27]. In a study done by Yildiz et al. on a health care dataset obtained from a hospital in Turkey, the authors have used four classification algorithms—artificial neural networks (ANN), support vector machines (SVM), Naïve Bayes (NB) and Ensemble Decision trees to perform classification for various types of anemia and the performance of the algorithms is benchmarked in which Bagged Decision Trees was the best performing algorithm [28]. Heru Mardiansyah et al. have studied the problem of imbalanced datasets and how this can be resolved by using SMOTE techniques to balance the original dataset. In their study, they have selected four datasets from the UCI machine learning repository—German credit cards, Winconsi, Glass and Ecoli to show the application of SMOTE techniques on the given datasets and the resulting datasets arising from this computation [29].

Kilicarslan et al. have constructed two hybrid models using genetic algorithms and Deep learning algorithms of stacked autoencoder (SAE) and convolutional neural networks (CNN) for the prediction and classification of iron deficiency anemia and benchmarked the performance of these two hybrid models in the classification computation for iron deficiency anemia [30].

Although several clinical different machine learning algorithms have been proposed that incorporate several data mining techniques for Anemia prediction, none of them had come up with a set of rules, which come handy in identifying the Anemia existing at different stage. Our proposed methodology is attempting to cover this gap by proposing optimal number of rules.

3 Proposed Method

This work provides an understandable set of rules which can be used for detecting Anemia at early stages using optimized rules extracted from Decision Tree Model using Anemia dataset. As there is unequal ratio of samples of each class, we adopted Synthetic Minority Oversampling Technique (SMOTE) for balancing the minority class. The balanced (SMOTE) dataset is also used with the Decision Tree Classifier for extracting rules. Thus, obtained rules are further optimized and evaluated for their efficacy and strength in classifying the Anemia dataset.

The complete description of the data is given in Sect. 4 and the complete features of the dataset are shown in Table 1.

The block diagram of the proposed model can be seen below in Fig. 1. The original dataset is partitioned into training and testing parts using stratified sampling with the ratio of seventy for training and thirty for testing. The SMOTE is applied only on training dataset to obtain the balanced (SMOTE) dataset. Thus, Decision tree

Table 1 Anemia dataset description

S. No.	Attributes names	Type of attribute	Abbreviation
1	Age	Numerical	Age
2	Gender	Character	Gender
3	Hemoglobin	Numerical	HGB
4	Mean cell volume	Numerical	MCV
5	Mean cell hemoglobin	Numerical	MCH
6	Mean cell hemoglobin concentration	Numerical	MCHC
7	Red cell distribution width	Numerical	RDW
8	Red blood cell count	Numerical	RBC
9	White blood cell count	Numerical	WBC
10	Platelet count	Numerical	PLT
11	Packed cell volume	Numerical	PCV

is trained on these two different training datasets (i.e., Actual and Balanced dataset) separately. The complete description of the proposed model can be seen in the next sub-section.

3.1 Description of the Proposed Model

The Decision Tree Classifier is trained using Actual Training dataset and Balanced (SMOTE) dataset separately. The model obtained after training, is tested with the testing data and the results obtained are noted as testing results. The rules from the Decision Tree model are also extracted separately for each model. It has been observed that 232 rules were generated by the Decision Tree Classifier when it has been trained using Actual Training dataset. On the other hand, it has given only 26 rules when Decision Tree Classifier is trained using SMOTE dataset.

Since 26 rules obtained from SMOTE dataset were giving significantly good results compared to results using Actual dataset we have considered these rules to take further for evaluating their efficacy and strengths and also to reduce the rules further.

These rules were coded individually and evaluated using the Anemia dataset for their relevancy and efficacy in terms of detecting the Anemia and they are sorted in descending order of their accuracies for each class. Next, two top yielding rules for each class are selected as reduced rules. These rules are again coded and are evaluated using the Testing dataset and the final results are obtained in the form of performance metrics defined.

The description of the proposed model can be defined using the following algorithm.

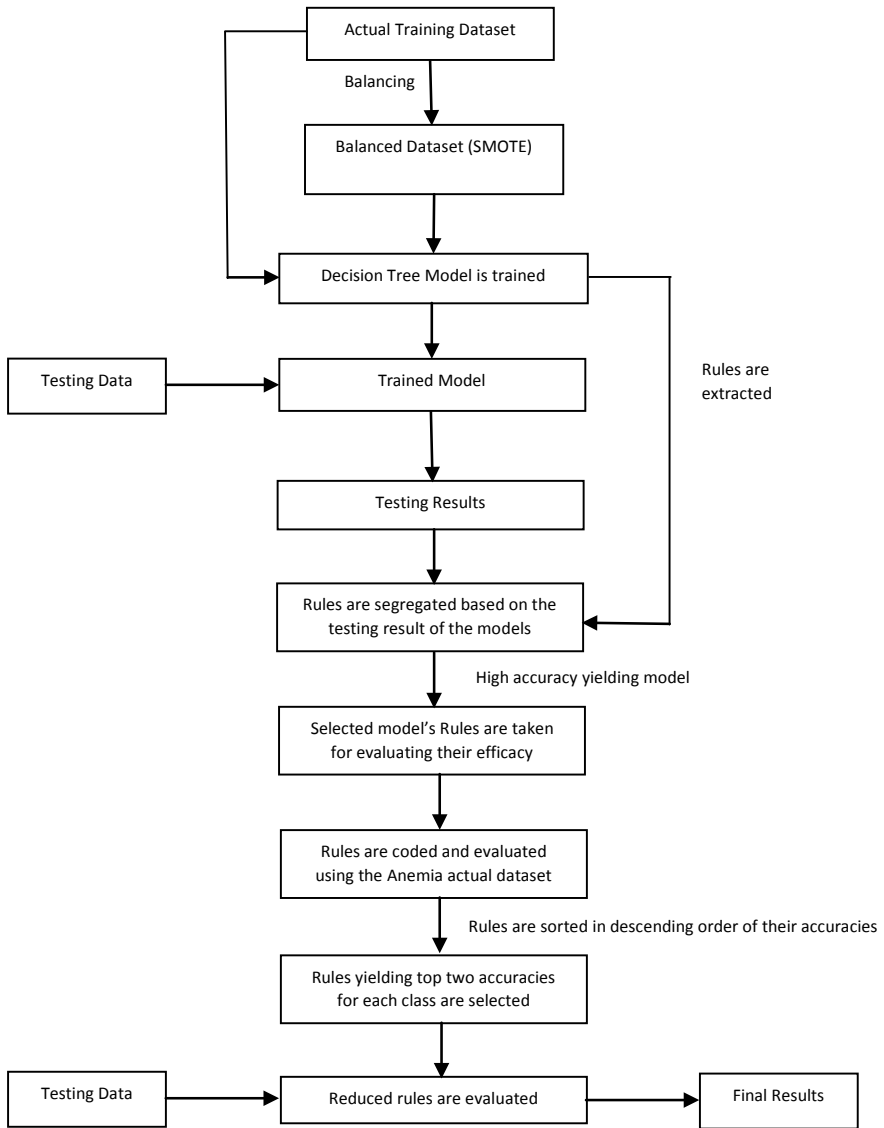


Fig. 1 Block diagram of the proposed model

<p>Input: Train data, Test data</p> <p>Output: decision rules</p> <p>Pre-processing steps:</p> <ol style="list-style-type: none"> 1. Data pre-processing is done (cleaning, removing of redundancy, etc.,) 2. Data is partitioned into training and testing sets with 70:30 ratio 3. Training data is balanced using SMOTE technique. <p>Algorithm steps:</p> <ol style="list-style-type: none"> 1) Start 2) For each training dataset 3) Apply Decision Tree classifier 4) Perform hyper parameter tuning, check accuracy 5) If accuracy is high enough (saturated) 6) Extract rules from the model break. 7) Else repeat step 3 8) End for 9) Test the trained model 10) Obtain the performance metrics (Recall, Precision and Accuracy) 11) Select the model with the best results 12) Code the extracted rules of the selected model 13) Evaluate the efficacy of the rules 14) Select the top two performing rules(reduced rules) for each class 15) Code the reduced rules 16) Evaluate the rules using Actual data 17) Obtain the performance metrics (Recall, Precision and Accuracy) 18) Stop

4 Dataset Description

The data used for the experiment was collected for the period of September 2020 to December 2020 in the form of CBC test reports from the Eureka diagnostic center, Lucknow, India [31]. Data was collected with ethical clearance from the diagnostic center and patient consent was obtained.

The Anemia can be classified into three different types based on their severity level. They are

- Mild,
- Moderate,
- Severe.

The Data consists of eleven attributes as illustrated in Table 1 with the size of the dataset being 364 records. The class variable is named as Hemoglobin (HGB) which has three classes, namely **Mild**, **Moderate** and **Severe**. These three classes have been defined using the range of values, where the Mild range lies between 11.0 and 12.0, Moderate range lies between 8.0 and 11.0 and Severe values lies less than 8.0. The distribution of the three classes in the dataset is 70.32% of Mild, 25.28% of Moderate and 4.40% of Severe.

1	Age	Sex	RBC	PCV	MCV	MCH	MCHC	RDW	TLC	PLT/mm ³	HGB
2	28	0	5.66	34	60.1	17	28.2	20	11.1	128.3	1
3	41	0	4.78	44.5	93.1	28.9	31	13	7.02	419	0
4	40	1	4.65	41.6	89.5	28.8	32.2	13	8.09	325	0
5	76	0	4.24	36.7	86.6	26.7	30.8	14.9	13.41	264	0
6	20	1	4.14	36.9	89.1	27.8	31.2	13.2	4.75	196	0
7	24	0	4.29	40.1	93.5	29.6	31.7	14.5	13.96	233	0
8	28	1	4.98	42.3	84.9	24.9	29.3	16.2	9.33	213	0
9	14	0	4.97	43.8	88.1	28	31.7	15.2	3.92	229	0
10	16	0	4.16	38.7	93	28.8	31	17.9	5.77	211	0
11	62	0	5.25	45.6	86.9	25.3	29.2	15.6	10.68	151	0
12	42	0	2.17	28.3	93.5	28.1	30	24.6	3.46	92	2
13	28	0	4.81	44.4	92.3	27.9	30.2	14.3	6.22	150	0
14	59	0	3.41	32.9	96.5	29.9	31	16.8	6.62	132	1
15	28	1	2.26	26.9	119	41.2	34.6	15.6	5.27	222	1

Fig. 2 Snapshot of the original dataset

As the ratio of the three classes were not balanced, we adopted a SMOTE balancing technique for ensuring proper balancing among all the classes of the dataset, so that the machine learning technique used for training would not get biased.

The snapshot of the dataset can be seen below in Fig. 2.

4.1 Algorithm Used (Decision Tree Classifier)

Decision Tree Classifier is a rule-based classifier which works on the basis of entropy. It uses different criterion functions like Gini Index and Information Gain for splitting the given data into one of the classes [32]. It can be clearly represented using a hierarchical tree-based diagram, where the classes are represented at the leaf level and the splitting features are represented at the interior nodes. This algorithm is mostly suitable for the decision-making problems where it mostly classifies the given problem into different classes more accurately. A snapshot of the Decision tree can be seen in Fig. 3.

The implementation of the Decision Tree Classifier was done in python using the sklearn library. Gini Index was taken as the criterion function.

5 Results and Discussion

All the experiments were conducted using hold-out method. The training data was taken a 70%, and testing data was taken as 30%. The splitting was done using stratified random sampling. Results of three models, two Decision Trees with Actual and SMOTED dataset and one proposed reduced rule-based method are evaluated and are presented in Table 2.

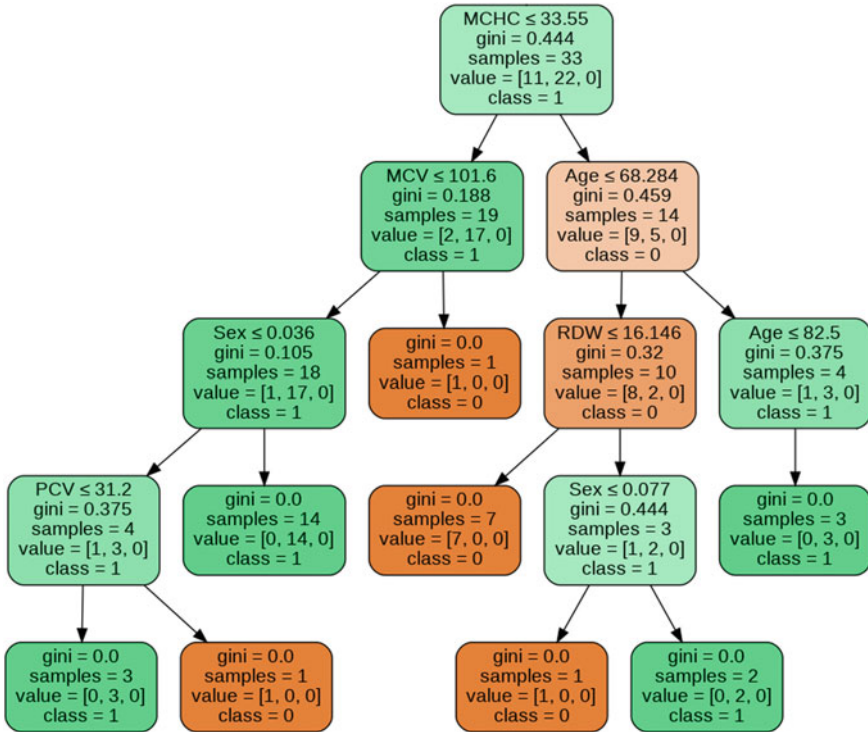


Fig. 3 Example of a decision tree

Table 2 Results of the decision tree using different datasets and reduced rules-based method

Metric	Actual dataset (decision tree)	SMOTED dataset (decision tree)	Coded reduced rules-based method
Accuracy	84.72	93.1	92.03
<i>Recall</i>			
Mild class	90.2	94.0	96.48
Moderate class	72.2	88.0	78.26
Severe class	66.7	95.0	100
<i>Precision</i>			
Mild class	97.9	95.0	95.54
Moderate class	72.2	88.0	78.26
Severe class	28.6	95.0	100

Table 3 Efficacy of the reduced rules

S. No	Rule number	Class	Efficacy
1	Rule 3	Severe	100
2	Rule 4	Severe	87.5
3	Rule 10	Moderate	68.47
4	Rule 15	Moderate	47.82
5	Rule 17	Mild	74.60
6	Rule 19	Mild	92.96

Accuracy, Recall and Precision have been used as performance metrics for measuring the efficacy of the results. These three can be defined as follows

- Accuracy can be defined as the ratio of total number of patients correctly classified irrespective of class to the total number of patients.

$$\text{Accuracy} = \frac{\text{Total number of correctly predicted samples(TP + FN)}}{\text{Total number of samples(TP + TN + FN + TF)}}$$

- Recall can be defined as the ratio of the number of patients correctly classified as Anemia class to the total number of Anemia class patients present in the dataset

$$\text{Recall} = \frac{\text{number of correctly predicted samples of a class(TP)}}{\text{Total number of samples in that class(TP + FN)}}$$

- Precision can be defined as the ratio of the number of patients correctly classified as Anemia class to the total number of patients classified as Anemia class.

$$\text{Precision} = \frac{\text{number of correctly predicted samples of a class(TP)}}{\text{Total number of samples classified as class(TP + FP)}}$$

The rules extracted using Actual dataset were 232 which is actually a huge number, and it would be difficult for the end user to interpret or use these many rules. Moreover, the results using these rules were not so significant. This is due to the imbalance nature of the dataset. Hence, SMOTE technique has been employed to balance the dataset. Using the balanced dataset with the Decision Tree Classifier, we obtained 26 rules which were more concise than earlier method and were also giving improving results. The rules using the balanced dataset are shown in Appendix A. Though the rules were concise, their efficacy was not so promising, and they were containing some not so important rules. Due to this, it would be difficult for the end user to use these set of rules handy.

So, in order to evaluate the efficacy of the rules, the 26 rules were coded and their efficacy in detecting the Anemia were computed by applying these rules on the Actual dataset. Then, the rules were sorted in descending order of their efficacy of their respective classes. Top two rules from each class are extracted as

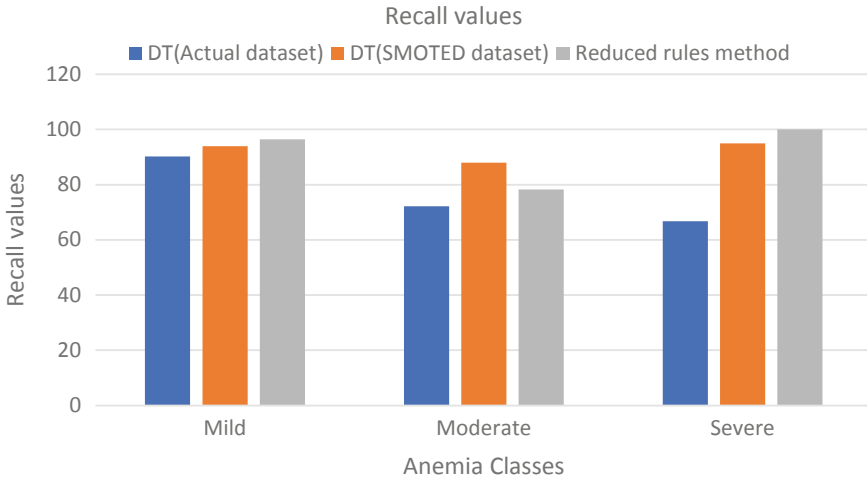


Fig. 4 Recall values for the three classes using decision tree and reduced rules-based method

reduced rules whose efficacies are shown below in Table 3. The reduced rules are presented in Appendix B.

Efficacy is computed using the following formula,

$$\text{Efficacy} = \frac{\text{number of correctly classified patients of a given class by a specific rule}}{\text{Total number of patients in that class}}$$

Next, these reduced rules were also coded and their performance metrics like Recall, Precision and Accuracy were also computed on the Actual dataset which are shown in Figs. 4, 5 and 6.

It can be observed from Table 2 that the Decision Tree Classifier using the Actual dataset has given the Accuracy, Recall and Precision values which are very low. This is due to the imbalance nature of the dataset. Whereas, in the case of Decision Tree Classifier using the SMOTED dataset, it can be observed that all the three metrics have improved compared to the results of the Actual dataset. While in the case of proposed Reduced rules-based method, it is evident that due to the elimination of not so important rules the recall value for the Anemia dataset has improved significantly in the case of Mild class and Severe class. Though there is slight reduction of 1% in the accuracy of Reduced rules-based method compared to Decision Tree Classifier using SMOTE dataset, it might be noted that the rules were reduced upto 77% which is a promising and significant contribution of this work. Having minimum rules which can be used for detection of Anemia would be an important tool for the end user to use.

Notations used in the following figures are,

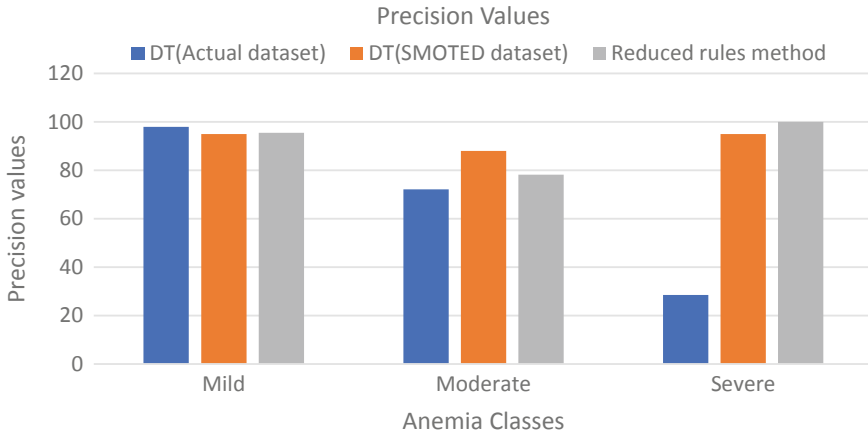


Fig. 5 Precision values for the three classes using decision tree and reduced rules-based method

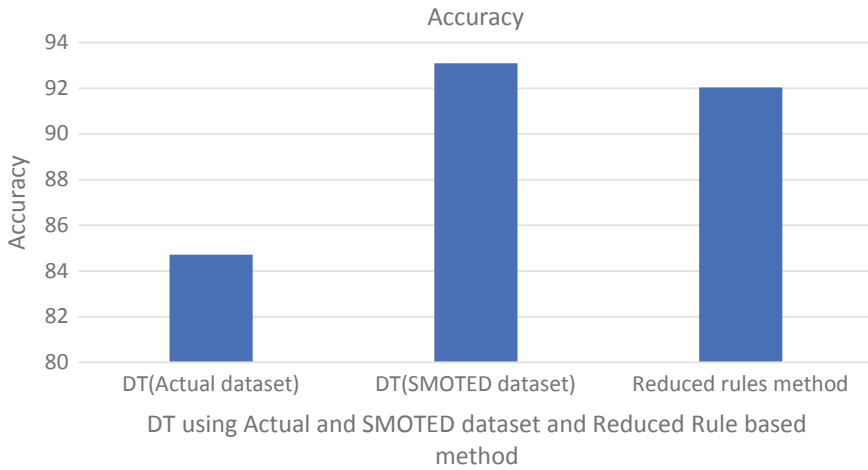


Fig. 6 Accuracy values for the three classes using decision tree and reduced rules-based method

- DT (Actual dataset): Decision Tree using the Actual dataset
- DT (Actual dataset): Decision Tree using the SMOTE dataset
- Reduced rules method: Reduced rule-based method on Actual dataset

Figure 4 shows the recall values of the three classes using the three different methods, two Decision Tree with Actual and SMOTED dataset and one with proposed Reduced rules-based method. It can be seen that the Reduced rules-based method has given good recall values of 96.48 and 100% compared to other methods in the case of Mild and Severe class. Whereas in the case of Moderate class, the Decision Tree Classifier with reduced (SMOTED) dataset has given good recall value of 88%.

Figure 5 shows the results of the precision values of the three different methods. It can be observed that in the case of Decision Tree Classifier using the Actual dataset, the precision values have been decreasing across different classes. While in the case of the Decision Tree Classifier using the SMOTED dataset and coded Reduced rules method the precision values have been slightly differing across all the classes. The precision value in case of Mild is topped by the Actual dataset, whereas in the case of Severe class the proposed Reduced rules-based method has given high result. While in case of Moderate class the Decision Tree Classifier using SMOTED dataset has topped.

As far as Accuracy is concerned, Fig. 6 shows that the accuracy has been improved in case of the Decision Tree Classifier using SMOTE dataset compared to Actual dataset and reduced rule dataset. The Reduced rules-based method has ranked second compared to the Decision Tree Classifier using SMOTE dataset. This may be due to the loss of information due to the reduction in the rules.

This significant improvement in Accuracy, Recall and Precision of the Decision Tree Classifier using SMOTE data and Reduced rules method can be attributed to the unbalanced nature of the dataset.

Since recall is an important parameter which helps in identifying the target class, it is significant to get a high recall which can classify any given test sample more confidently. Hence, as the Reduced rules-based method has given highest Recall for the Mild and Severe class, it helps in detecting the presence of Anemia at the early stages there by helping the medical fraternity. The use of reduced decision rules would come handy for the medical practitioners in detecting the Mild class Anemia and thereby giving suitable medication for delaying from further deterioration. This is the novel contribution of this work.

6 Conclusion and Direction for Future Work

Anemia detection is one of the challenging issues in current scenario. To address this issue, we have taken Anemia dataset from India. Due to the unbalanced nature of the dataset, we have used SMOTE technique to balance the classes. Decision Tree Classifier and Reduced rule-based method has been used to detect the presence of Anemia from a given dataset. The Reduced rule-based method was able to achieve significant results, especially in the case of Mild class compared to the Decision Tree Classifier using Actual dataset and SMOTE dataset. Due to the smaller number of rules given by reduced rule-based method, it can also be used as handy tool for detection of Anemia. As a future work deep learning algorithm can be used to anemia classifying and optimization-based methods may be used for rules reduction.

Appendix A

Rules Obtained Using Decision Tree on Balanced (SMOTE) Dataset

S. No.	Antecedent	Consequent
1.	if(PCV <= 29.22) and if(PCV <= 26.13) and if(TLC <= 4.548)	Moderate
2.	if(PCV <= 29.22) and if(PCV <= 26.13) and if(TLC <= 4.548) else if(MCHC <= 38.43) and if(PCV <= 24.88) and if(Age <= 24.53) and if(PLT/mm3 <= 214.05)	Moderate
3.	if(PCV <= 29.22) and if(PCV <= 26.13) and if(TLC <= 4.548) else if(MCHC <= 38.43) and if(PCV <= 24.88) and if(Age <= 24.53) and if(PLT/mm3 <= 214.05) else	Severe
4.	if(PCV <= 29.22) and if(PCV <= 26.13) and if(TLC <= 4.548) else if(MCHC <= 38.43) and if(PCV <= 24.88) and if(Age <= 24.53) else	Severe
5.	if(PCV <= 29.22) and if(PCV <= 26.13) and if(TLC <= 4.548) else if(MCHC <= 38.43) and if(PCV <= 24.88) else if(MCHC <= 31.89) and if(RDW <= 15.54)	Moderate
6.	if(PCV <= 29.22) and if(PCV <= 26.13) and if(TLC <= 4.548) else if(MCHC <= 38.43) and if(PCV <= 24.88) else if(MCHC <= 31.89) and if(RDW <= 15.54) else	Severe
7.	if(PCV <= 29.22) and if(PCV <= 26.13) and if(TLC <= 4.548) else if(MCHC <= 38.43) and if(PCV <= 24.88) else if(MCHC <= 31.89) else	Moderate
8.	if(PCV <= 29.22) and if(PCV <= 26.13) and if(TLC <= 4.548) else if(MCHC <= 38.43) else if(RBC <= 2.14)	Severe
9.	if(PCV <= 29.22) and if(PCV <= 26.13) and if(TLC <= 4.548) else if(MCHC <= 38.43) else if(RBC <= 2.14) else	Moderate
10.	if(PCV <= 29.22) and if(PCV <= 26.13) else if(RDW <= 17.42) and if(MCHC <= 38.66)	Moderate
11.	if(PCV <= 29.22) and if(PCV <= 26.13) else if(RDW <= 17.42) and if(MCHC <= 38.66) else if(Age <= 64.0)	Moderate
12.	if(PCV <= 29.22) and if(PCV <= 26.13) else if(RDW <= 17.42) and if(MCHC <= 38.66) else if(Age <= 64.0) else	Mild
13.	if(PCV <= 29.22) and if(PCV <= 26.13) else if(RDW <= 17.42) else if(Age <= 24.23)	Moderate
14.	if(PCV <= 29.22) and if(PCV <= 26.13) else if(RDW <= 17.42) else if(Age <= 24.23) else	Severe
15.	if(PCV <= 29.22) else if(PCV <= 36.48) and if(MCHC <= 32.64) and if(RBC <= 4.71) and if(PCV <= 35.27)	Moderate
16.	if(PCV <= 29.22) else if(PCV <= 36.48) and if(MCHC <= 32.64) and if(RBC <= 4.71) and if(PCV <= 35.27) else if(MCHC <= 30.58)	Moderate
17.	if(PCV <= 29.22) else if(PCV <= 36.48) and if(MCHC <= 32.64) and if(RBC <= 4.71) and if(PCV <= 35.27) else if(MCHC <= 30.58) else	Mild

(continued)

(continued)

S. No.	Antecedent	Consequent
18.	if(PCV <= 29.22) else if(PCV <= 36.48) and if(MCHC <= 32.64) and if(RBC <= 4.71) else if(MCH <= 20.92)	Moderate
19.	if(PCV <= 29.22) else if(PCV <= 36.48) and if(MCHC <= 32.64) and if(RBC <= 4.71) else if(MCH <= 20.92) else	Mild
20.	if(PCV <= 29.22) else if(PCV <= 36.48) and if(MCHC <= 32.64) else if(RBC <= 3.46)	Moderate
21.	if(PCV <= 29.22) else if(PCV <= 36.48) and if(MCHC <= 32.64) else if(RBC <= 3.46) else if(MCH <= 27.66) and if(MCV <= 81.63)	Mild
22.	if(PCV <= 29.22) else if(PCV <= 36.48) and if(MCHC <= 32.64) else if(RBC <= 3.46) else if(MCH <= 27.66) and if(MCV <= 81.63) else	Moderate
23.	if(PCV <= 29.22) else if(PCV <= 36.48) and if(MCHC <= 32.64) else if(RBC <= 3.46) else if(MCH <= 27.66) else	Mild
24.	if(PCV <= 29.22) else if(PCV <= 36.48) else if(MCHC <= 29.08) and if(PCV <= 37.64)	Moderate
25.	if(PCV <= 29.22) else if(PCV <= 36.48) else if(MCHC <= 29.08) and if(PCV <= 37.64) else	Mild
26.	if(PCV <= 29.22) else if(PCV <= 36.48) else if(MCHC <= 29.08) else	Mild

Appendix B

Reduced Rules

S. No.	Antecedent	Consequent
1.	if(PCV <= 29.22) and if(PCV <= 26.13) and if(TLC <= 4.548) else if(MCHC <= 38.43) and if(PCV <= 24.88) and if(Age <= 24.53) and if(PLT/mm3 <= 214.05) else	Severe
2.	if(PCV <= 29.22) and if(PCV <= 26.13) and if(TLC <= 4.548) else if(MCHC <= 38.43) and if(PCV <= 24.88) and if(Age <= 24.53) else	Severe
3.	if(PCV <= 29.22) and if(PCV <= 26.13) else if(RDW <= 17.42) and if(MCHC <= 38.66)	Moderate
4.	if(PCV <= 29.22) else if(PCV <= 36.48) and if(MCHC <= 32.64) and if(RBC <= 4.71) and if(PCV <= 35.27)	Moderate
5.	if(PCV <= 29.22) else if(PCV <= 36.48) and if(MCHC <= 32.64) and if(RBC <= 4.71) and if(PCV <= 35.27) else if(MCHC <= 30.58) else	Mild
6.	if(PCV <= 29.22) else if(PCV <= 36.48) and if(MCHC <= 32.64) and if(RBC <= 4.71) else if(MCH <= 20.92) else	Mild

References

1. J.R. Khan, N. Awan, F. Misu, Determinants of anemia among 6–59 months aged children in Bangladesh: evidence from nationally representative data. *BMC Pediatr.* **16**(1), 3 (2016)
2. J.E. Ewusie, C. Ahiadeke, J. Beyene, J.S. Hamid, Prevalence of anemia among under 5 children in the Ghanaian population: estimates from the Ghana demographic and health survey. *BMC Public Health* **14**(1), 626 (2014)
3. M. Abdullah, S. Al-Asmari, Anemia types prediction based on data mining classification algorithms, in *Communication, Management and Information Technology*, ed. by Sampaio de Alencar (2017)
4. B. Sow, S. Hiroki, M. Hamid, A. Hafiz Farooq, Using biological variables and social determinants to predict malaria and anemia among children in Senegal. *IEICE Swim* **117**, 13–20 (2017)
5. J.G.I. Paulo, A review of evidence of health benefit from artificial neural networks in medical intervention. *Neural Netw.* **15**(1), 11–39 (2002)
6. S. Smys, W. Haoxiang, Naïve Bayes and entropy based analysis and classification of humans and chat bots. *J. ISMAC* **3**(01), 40–49 (2021)
7. World Health Organization, *The World Health Report 2002: Reducing Risks, Promoting Healthy Life* (World Health Organization, 2002)
8. B.J. Brabin, M. Hakimi, D. Pelletier, Iron deficiency anemia: reexamining the nature and magnitude of the public health problem. *J. Nutr.* **131**, 6045–6155 (2001)
9. E. Mclean, M. Cogswell, I. Egli, B. Wojdyla, B. De Benoist, Worldwide prevalence of anemia, WHO vitamin and mineral nutrition information system, 1993–2005. *Public Health Nutr.* **12**(4), 444–454 (2009)
10. G. Battineni, G.G. Sagaro, N. Chinatalapudi, F. Amenta, Applications of machine learning predictive models in the chronic disease diagnosis. *J. Pers. Med.* **10**(2), 21 (2020)
11. X.H. Meng, Y.X. Huang, D.P. Rao, Q. Zhang, Q. Liu, Comparison of three data mining models for predicting diabetes or prediabetes by risk factors. *Kaohsiung J. Med. Sci.* **29**(2), 93–99 (2013)
12. S.B. Choi, W.J. Kim, T.K. Yoo, J.S. Park, J.W. Chung, Y.H. Lee, E.S. Kang, D.W. Kim, Screening for prediabetes using machine learning models. *Comput. Math. Methods Med.* **2014**, 618976 (2014)
13. W. Yu, T. Liu, R. Valdez, M. Gwinn, M.J. Khoury, Applications of support vector machine modeling for prediction of common diseases: the case of diabetes and pre diabetes. *BMC Med. Inform. Decis. Mak.* **10**(1), 16 (2010)
14. C.H. Hsieh, R.H. Lu, N.H. Lee, W.T. Chiu, M.H. Hsu, Y.C. Li, Novel solutions for an old disease: diagnosis of acute appendicitis with random forest, support vector machines and artificial neural networks. *Surgery* **149**(1), 87–93 (2011)
15. Y. Zhao, B.C. Healy, D. Rotstein, C.R. Guttmann, R. Bakshi, H.L. Weiner, C.E. Brodley, T. Chitnis, Exploration of machine learning techniques in predicting multiple sclerosis disease course. *PloS one* **12**(4), e0174866
16. S.A. Sanap, M. Nagori, V. Kshirsagar, Classification of anemia using data mining techniques, in *International Conference on Swarm, Evolutionary and Memetic Computing 2011* (Springer, Berlin, Heidelberg, 2011), pp. 113–121
17. M.B. Mengesha, Dadi, *Prevalence of anemia among adults at Hawassa University referral hospital, Southern Ethiopia.* *BMC Hematol.* **19**, 1 (2019)
18. S. Benjamin, T. Shripad, R. David, Machine learning approaches to the social determinants of health in the health and retirement study. *SSM Popul. Health* **4**, 95–99 (2018)
19. A. Widodo, B.-S. Yang, Support vector machine in machine condition monitoring and fault diagnosis. *Mech. Syst. Signal Process.* **21**(6), 2560–2574 (2007)
20. B. Sow, H. Mukhtar, H.F. Ahmad, H. Suguri, Assessing the relative importance of social determinants of health in malaria and anemia classification based on machine learning techniques. *Inform. Health Soc. Care* **45**(3), 229–241 (2020)

21. M. Jaiswal, A. Srivastava, T.J. Siddiqui, Machine learning algorithms for anemia disease prediction, in *Recent Trends in Communication, Computing, and Electronics* (Springer, Singapore, 2019), pp. 463–469
22. N. Almgren, N. Alrumayyan, R. Alnashwan, A. Alfutamani, I. Al-Turaiki, O. Almgren, The effect of Vitamin B12 deficiency on blood count using data mining, in *5th International Symposium on Data Mining Applications* (Springer, Cham, 2018), pp. 234–245
23. S. Jatoi, M.A. Panhwar, M.S. Memon, J.A. Baloch, S. Saddar, Mining complete blood count reports for disease discovery. *Int. J. Comput. Sci. Netw. Secur.* **18**(1), 121–127 (2018)
24. K. Meena, D.K. Tayal, V. Gupta, A. Fatima, Using classification techniques for statistical analysis of Anemia. *Artif. Intell. Med.* **94**, 138–152 (2019)
25. C.C. Chern, Y.J. Chen, B. Hsiao, Decision tree-based classifier in providing telehealth service. *BMC Med. Inform. Decis. Mak.* **19**(1), 1–15 (2019)
26. K.S. Lakshmi, G. Vadivu, Extracting association rules from medical health records using multi-criteria decision analysis. *Procedia Comput. Sci.* **115**, 290–295 (2017)
27. Y. Song, Y. Lu, Decision tree methods: applications for classification and prediction. *Shanghai Arch. Psychiatr.* **27**(2), 130–135 (2015)
28. T.K. Yıldız, N. Yurtay, B. Öneç, Classifying anemia types using artificial learning methods. *Eng. Sci. Technol. Int. J.* **24**(1), 50–70 (2021)
29. H. Mardiansyah, R.W. Sembiring, S. Efendi, Handling problems of credit data for imbalanced classes using SMOTEXGBoost. *J. Phys. Confer. Ser.* **1830**(1), 012011 (2021)
30. S. Kilicarslan, M. Celik, Ş. Sahin, Hybrid models based on genetic algorithm and deep learning algorithms for nutritional Anemia disease classification. *Biomed. Signal Proces. Control* **63**, 102231 (2021)
31. R. Vohra, J. Pahareeya, A. Hussain, Complete blood count anemia diagnosis. *Mendeley Data* **V1** (2021). <https://doi.org/10.17632/dy9mfjchm7.1>
32. S.R. Safavian, D. Landgrebe, A survey of decision tree classifier methodology. *IEEE Trans. Syst. Man Cybern.* **21**(3), 660–674 (1991)

Design of Low Power Sequential Circuits Using GDI Cells



Sujatha Hiremath and Deepali Koppad

Abstract This paper describes the development of a 180 nm standard cell library designed for building power efficient digital circuits. Creating an Integrated Circuit (IC) can be very time consuming if high flexibility or low power is demanded. This paper will try to solve this problem by creating own standard cell libraries, which in turn gives less power. Having these libraries makes it possible to map Verilog code to these libraries, using them instead of predefined cell libraries. The modified full swing Gate Diffusion Input (GDI) design style is used for designing the schematics of basic cells in the standard cell library. The benefits of using this low power technique at circuit level helps to reduce the static power dissipation as compare to CMOS logic for the digital circuits. The static power reduction for sequential designs reduces up to 20% as compared to CMOS logic for the different ISCAS sequential circuits.

Keywords Standard cell library · Gate diffusion input · Low power design · Cadence

1 Introduction

The large Integrated Circuit (IC) consists of major blocks such as logic, memory, controller and standard cell library, etc. The overall power consumption of the chip is depends upon the design of all the blocks within it. Among these blocks, the standard cell library contributes more to the overall power consumption of the chip. Hence, developing a low power standard library for power efficient designs is very much important. Designing our own standard cell library for the low power applications results better than the existing library. In this paper using low power standard cells, some of the ISCAS sequential circuits are synthesized and power is compared with the CMOS standard cell library. The standard cells are developed by using the Gate

S. Hiremath (✉)

Department of Electronics and Communication Engineering, RVCE, Bengaluru, India
e-mail: sujathah@rvce.edu.in

D. Koppad

Department of Electronics and Communication Engineering, RIT, Bengaluru, India

Diffusion Input (GDI) technique which is one of the low power technique. All the basic gates such as AND, OR, NOT, XOR gates are designed using this technique. The limitation of the GDI technique is the reduced swing at the output. The basic GDI gates are modified to achieve full swing at the output using additional transistor as a restoring logic. After the schematic, layout of the same schematics are created and verified by performing the Design Rule Check (DRC), Layout Vs Schematic (LVS). The back annotation is performed to extract the parasitic from the layout, i.e., av_extracted view. Using the Liberate tool characterization of the cells are carried out with typical-typical process corner, voltage of 1.8 V and at room temperature. Verilog code for sequential designs are synthesis using low power standard cell library.

2 Design Methodology

2.1 Design of Proposed Gates

The proposed library has been designed using a cadence 180 nm technology, which consists of basic gates, adder, and D-Flip flop cells. The standard cells are designed using CMOS logic. For low power cell design, Gate Diffusion Input technique alternate to CMOS logic is used. The Gate Diffusion Input technique is one of the low power technique used to design basic gates and other digital circuits. The basic GDI cells are having the limitations of reduced swing. Hence, these cells are modified to achieve the full swing at the output [1]. The existing GDI basic gates as shown in Fig. 1a–c are having the limitation of producing the full swing voltage at the output for few combination of the input. Those combination of inputs are modified to achieve the full voltage swing. Using basic GDI technique, the number of transistors required to implement AND, OR, NOT is only 2 as compared to 6 transistor using CMOS logic. The XOR gate is designed using only 4 transistor.

The simulated output of GDI AND gate and OR gate is shown in Fig. 2a, b, respectively.

Figure 2a shows the output of GDI AND gate with weak logic 0 and 1 for few combination of inputs. This is because of the NMOS/PMOS transistor behavior. The

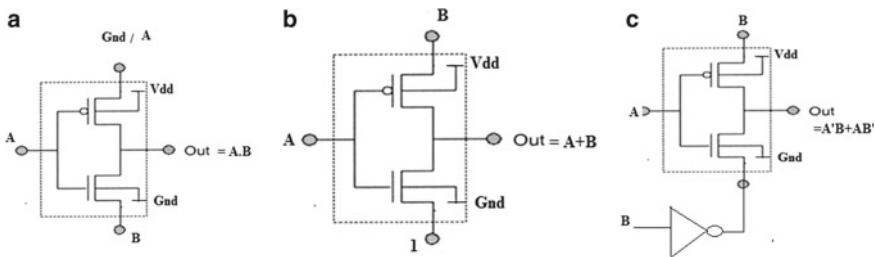


Fig. 1 a–c GDI AND, OR and XOR gate

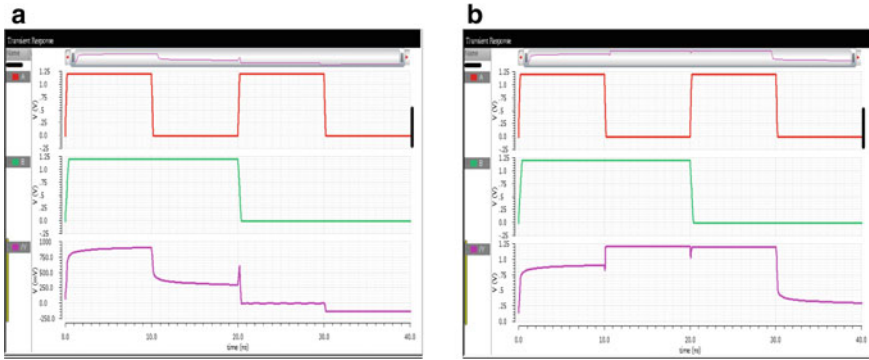


Fig. 2 a AND gate output, b OR gate output

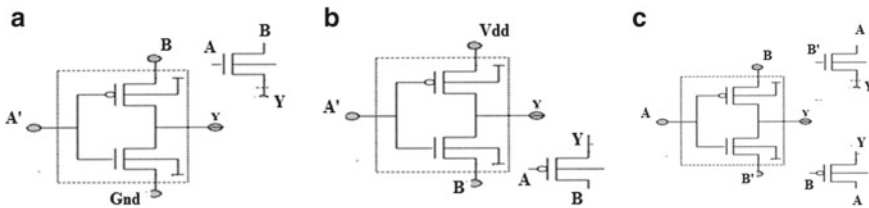


Fig. 3 a–c Full swing GDI AND, OR and XOR gate

NMOS transistor produces good logic 0 and weak logic 1. Similarly, PMOS transistor produces good logic 1 and weak logic 0.

The proposed design uses the additional transistor to improve the voltage swing, and hence, number of transistors are increased but the average power is still lesser than the CMOS logic gates. The proposed GDI gates are as shown in Fig. 3a–c. The average power of proposed basic gates are still lesser than the basic GDI cells.

The simulated output of proposed cells for an example of AND and OR gates are shown in Fig. 4a, b, respectively.

The average power of modified GDI cells are compared with CMOS logic. The modified GDI cells are more power efficient than the CMOS logic. The average power of both cells are shown in Table 1.

The next step is to develop a layout of each schematic and verify the layouts by Design Rule Check and Layout Vs Schematic.

2.2 Standard Cell Library

To create standard cell library, first step is to decide the height of the cell based on the largest circuit in the library. In the proposed library, the D-Flip Flop is the largest

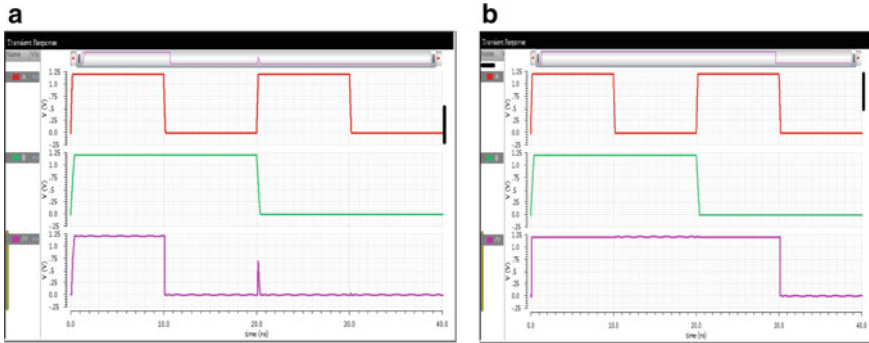


Fig. 4 a, b Modified GDI AND and OR gate output

Table 1 Average power of cells

Cell	Average power (uW)	
	CMOS logic	Proposed logic
AND	1.72	5.71
OR	1.92	5.37
XOR	2.8	10.3

cell, and hence, the height of the standard cell library is chosen as 15 μm . All the cells in the library is having the same height. The pitch is calculated for this technology as Pitch size = minimum spacing + width of the metal. There are 3 different ways in which pitch size can be measured [2, 3]. The line to line pitch calculation is used for layout and it is 0.6 μm for the 180 nm technology. This provides high routing density. The width of the cell varies according to the schematic. Both height and width parameters should be a multiple of pitch size. The N-well drawn for PMOS transistor is 10 μm , i.e., 2/3rd of the total height. To avoid the shorts and opens in the layout proper dimensions are measure and inverter layout is shown with the dimensions. Similarly, all the cells required in the library are developed with the proper measurements. The layout of NOT, AND, OR gates of proposed designs are shown Fig. 5a–c, respectively. The XOR gate and DFF layout is as shown in Fig. 6a, b.

3 Characterization

Using the Liberate tool from Cadence, the characterization of each logic cells is carried out. The characterization involves evaluating Input/Output Delay, Timing constraints (Setup/Holdtime, Recovery/Removal time), Input capacitance, logical expression and power dissipation. Before performing the characterization, verification of the layout and generating the av_extracted view from the layout is performed.

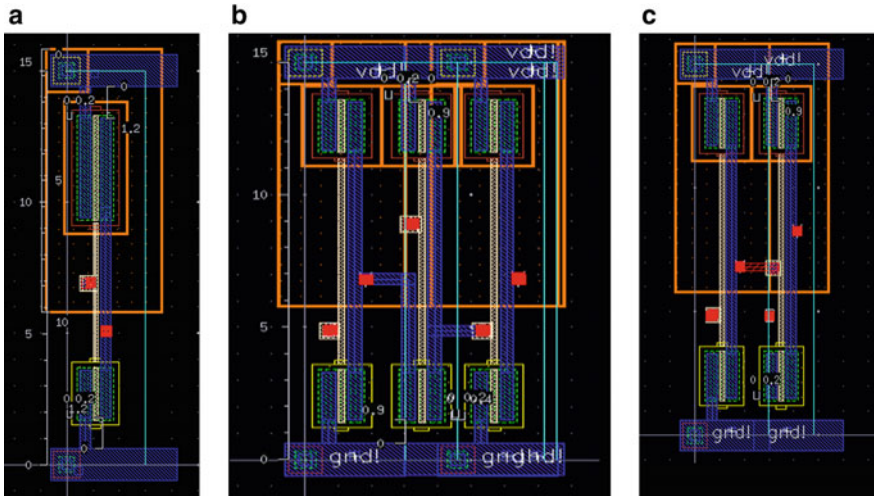


Fig. 5 a–c Layout of inverter, proposed AND and OR gate

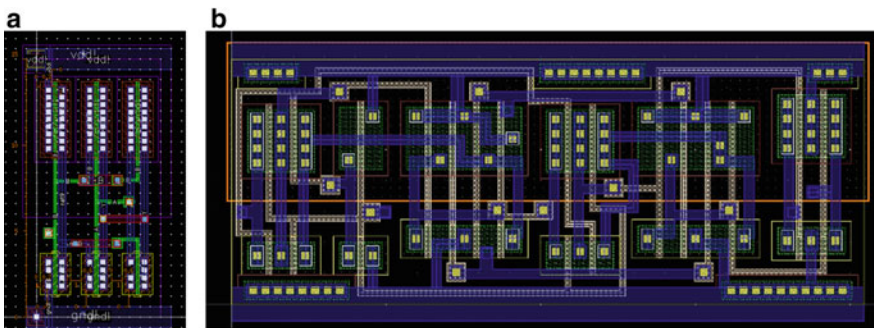


Fig. 6 a, b Layout of proposed XOR and DFF

The next step is generating the .lib file. There two files need to be generated for building the cell library. The first one is Liberty Timing file (LIB) and second is Liberty Exchange format (LEF) file [4, 5]. The LIB file is used during logical synthesis process, and LEF file is used during the Placement and Routing process. Additional files such as char.tcl, dut.scs, gpdk.scs files are needed. The dut.scs file generates the netlist from all the cells which are used in the library, which includes the information about the parasitics of each cells. The char.tcl file includes the information about the PVT, each cell format. Next step involves the synthesis of the digital design written in Verilog code using the standard cell library.

```

Operating conditions: PVI IP8V_25C (balanced_tree)
Wireload mode:      enclosed
Area mode:          timing library
=====
Instance Cells  Leakage  Dynamic  Total
                Power(nW) Power(nW) Power(nW)
-----
s713            248      8.048 278211.568 278219.608
  DFF_1          1        0.181  1867.767  1867.948
  DFF_15         1        0.181  1867.767  1867.948
  DFF_16         1        0.181  1867.767  1867.948

```

Fig. 7 Power report of S713 ISCAS sequential circuit

4 Synthesis

In electronics, logic synthesis is a process by which an abstract form of desired circuit behavior, typically at register transfer level (RTL), is turned into a design implementation in terms of logic gates, typically by a computer program called a synthesis tool. ISCAS sequential bench mark circuits such as S27, S713, etc., are synthesized using both proposed and CMOS standard cell library. The ISCAS benchmark circuit have been used by many researchers as a basis for comparing results in the area of test generation. The sequential bench marks circuits such as S27, S641 are considered for the analysis of power with respect to the proposed standard cell library.

5 Results

The proposed standard cell library is efficient for reducing the static power dissipation of the digital designs as compared to the CMOS logic. The power report of S713 ISCAS sequential is shown in Fig. 7.

The dynamic and static power of ISCAS bench mark circuits are reported as shown in Table 2.

6 Conclusion

Design of low power standard cell library is important for low power applications. Hence, developing own standard cells according to the requirement helps the designer to obtain the power efficient digital circuits. The low power standard cell has been created and used for the synthesis of ISCAS sequential circuits. The static power

Table 2 Dynamic and static power of ISCAS circuits

Circuits	Dynamic power (nW)		Leakage power (nW)		% of improvement in the leakage power
	CMOS	Proposed	CMOS	Proposed	
S27	5219.10	4974.48	0.495	0.378	23.4
S713	279,694.00	278,211.56	9.115	8.040	11.7
S641	282,901.17	279,991.29	9.035	8.029	11.1
S1488	275,048.71	275,469.83	27.085	21.687	20
S420	151,896.58	154,907.77	7.222	6.218	14

dissipation of these sequential circuits are less as compared to the CMOS standard cells. Upto 23% of static power reduction from the above listed ISCAS sequential circuits.

Future Scope: The cells in the library can be increased with different driving strength and can be used for synthesis of complex digital circuits.

References

1. Morgenshtein A, Fish A, Wagner IA (2002) Gate-diffusion input (GDI): a power-efficient method for digital combinatorial circuits. *IEEE Trans Very Large Scale Integr (VLSI) Syst* 10(05)
2. Sangmesh M, Sumukha M, Manasa CK (2019) Standard cell library development. *Int J Eng Res Technol (IJERT)* 8(07)
3. Ahmeduddin SK, Srividya P (2020) Standard cell library characterization and analysis of different liberty models during STA. *J Crit Rev* 7(14)
4. Patel C, Standard cell library/library exchange format (lef), *Advanced VLSI Design (CMPE 641)*, UMBC, [Online]. Available: https://www.csee.umbc.edu/~cpatel2/links/641/slides/lect04_LEF.pdf
5. Ha DS, Instructions for lef file generation process, Virginia Tech VLSI for Telecommunications, [Online]. Available: http://www.vtvt.ece.vt.edu/tutorial/Instructions_for_LEF_File_Generation_Process.pdf

Automatic Drainage Monitoring and Alert System Using IoT



K. R. Chairma Lakshmi, B. Praveena, K. Vijayanand, and S. Vijayalakshmi

Abstract The implementation of smart technology is used to improve the safety and efficiency in the drainage monitoring system, which decreases the need of man power and transport costs. The proposed system which accentuates on “SMART CITIES” is used to implement a smarter way of sewage management using smart sensors connected through IoT. The objective of this scheme is to develop a smarter way of conventional drainage monitoring system using smart sensors and IoT. Another key intention is to prevent gas poisoning in sewage maintenance work which can be deadly. An alert system is to be established which constantly monitors the flow, level and toxic gas amount of the sewage pipeline. The status of the particular drainage route is constantly monitored and displayed on the webpage. If any possibility of overflow is detected, the sewage is pumped to the alternate drain automatically until manual maintenance can be performed.

Keywords Sanitized workers safety · Flow sensor · IoT · Level sensor · Toxic gas sensor

1 Introduction

Drainage system plays a vital role in the metropolitan cities. One of the most critical issues of the recent times is the absence of proper sewage monitoring system. Monitoring of drainage status like overflowing of waste water in drainage, toxic gas present in drainage, etc., is not possible manually. Regardless of how well built a city’s architecture is the sewage system still proves to be poorly maintained. The most important priorities are to ensure a clean and healthy globe and to protect the urban environment. Drainage monitoring is a crucial task and a tedious one since the location of the leakage cannot be identified easily. The problem arising in such drainage lines like blockage due to waste solid and liquid, rapid rise in the water level as well as various harmful gases will be unavoidable if the appropriate cleaning actions are

K. R. Chairma Lakshmi (✉) · B. Praveena · K. Vijayanand · S. Vijayalakshmi
Department of Electronics and Instrumentation Engineering, R.M.K. Engineering College,
Gummidipoondi, India

not taken from time to time can root serious problems to the everyday routing of the city. The absence of efficient drainage management has caused serious environmental problems and cost issues. In the existing methods, IoT is used for various applications or fields like health monitoring, industrial automation, agriculture, transference, military, logistics, environmental and security purpose [1–5]. A long-term drain monitoring system designed using a variety of sensors, including water level sensor, blockage, and gas sensor. The flood can be classified as low, medium, or high depending on the water level sensor. This can help with flood detection in the early stages [6–10]. The gas sensors are designed to detect a variety of harmful gases so that drain workers may take measures when entering manholes [11–18]. Similarly, wireless sensor networks with flow and level sensor unit also used for monitoring the drainage. The drawback of the existing systems is even through drainage level and gas monitoring by system but it not providing intimation to works about location of the drainage, Cost of WSN is too high [19–21].

The aim of this proposed method is to present a smart drainage management system with help of Internet of Things and intelligent sensors to constantly monitor the drains in urban areas. Ultrasonic sensor is used to detect the level of the drainage. Similarly, a Hall Effect type flow metre is used to constantly measure the flow of drainage. Gas sensor is used to detect the presents of harmful gases like hydrogen sulphide, methane in the drainage. These sensors are connected to the internet through a Wi-Fi module and their status can be monitored using the webpage. The webpage is developed using PHP and HTML 5. The collected data are stored in MySql database.

2 Materials and Method

2.1 Proposed Methodology

The Photographic image and block diagram of proposed model are shown in Figs. 1 and 2, respectively. This arrangement provides an automatic mechanism to constantly monitor the drainage lines for various parameters such as drain level, flow rate of the sewage and the presence of various kinds of toxic gases. These sensors are intelligently connected to Internet of things through a sensor network. The various values of the sensors are constantly updated in every moment of time in an exclusively created IoT webpage. This webpage has a unique login page in which the respective municipality departments can login and view the status. New users can also create a login membership.

Fig. 1 Photographic image of the prototype

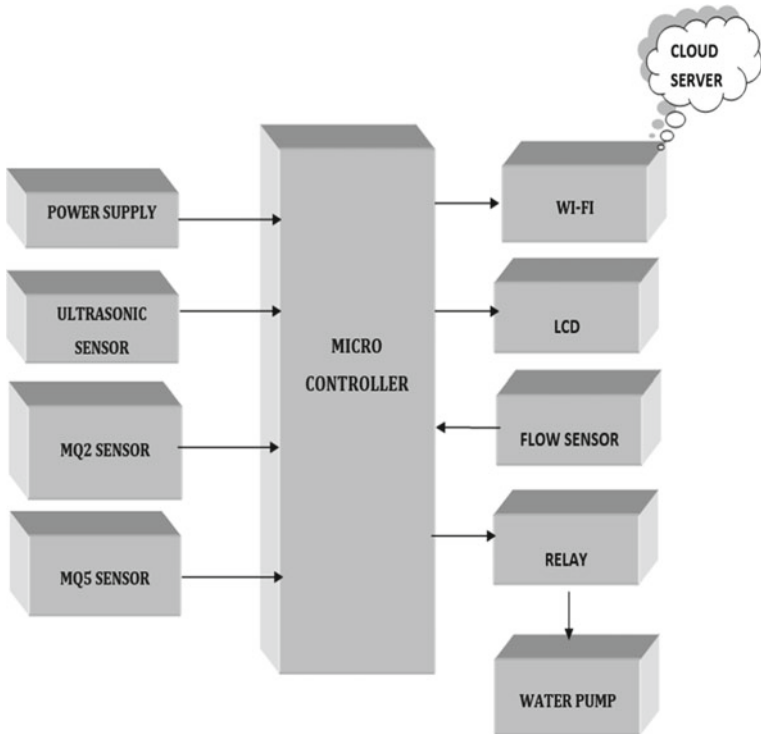


Fig. 2 Block diagram of the hardware section

2.2 Flow Chart of Proposed Methodology

Figure 3 shows the flow chart of proposed system. In case, the level of the drainage increases beyond the threshold limit an alert is immediately displayed on LCD display screen. Simultaneously, the sewage water is pumped to the next drain to prevent overflowing for a brief period of time until manual assistance can be deployed. In the webpage, each drain has the previous history of the particular route for future references. Along with that location of every particular drain is also prefixed, which enables the sanitation workers to identify the blockage easily.

3 Result and Analysis

The status of the drainage is constantly monitored using an IoT webpage. The webpage consists of an initial signup page in which the first-time users are required

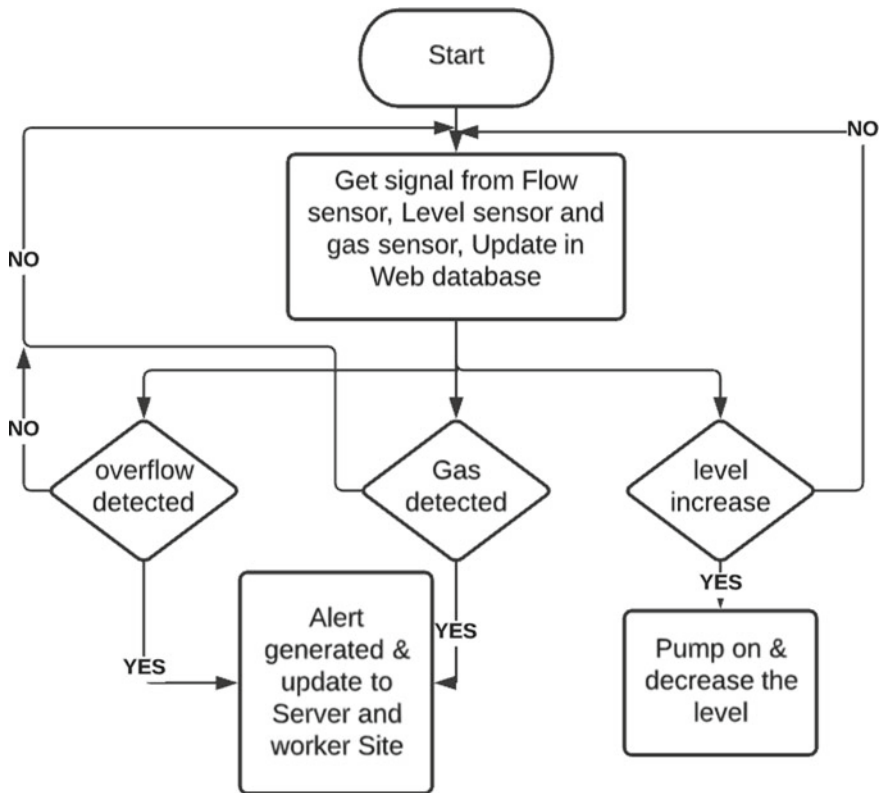


Fig. 3 Flow chart of the proposed system

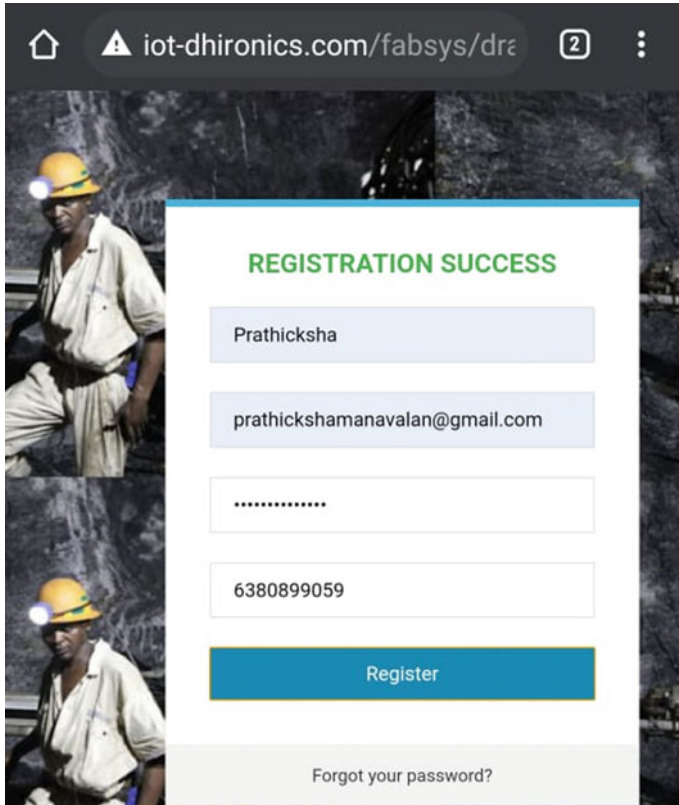


Fig. 4 User authentication

to create an account as shown in Fig. 4. Once the login credentials are generated, the concerned urban municipality departments can check the status. Once in, a series of drain routed are collectively visible in the following webpage, all of which status are constantly monitored as shown in Fig. 5. Each drainage monitoring webpage has various attributes which displays flow rate, total volume in litre, toxic gas levels and level status along with time stamp as shown in Fig. 6. By the use of MySQL RDBMS, the previous history of each drainage is also stored for future reference unless reset. The location of each drain is also visible in the Google map which helps in easy identification of blockage. In case any overflow occurs, an alert is shown in the LCD Screen as shown in Fig. 11.

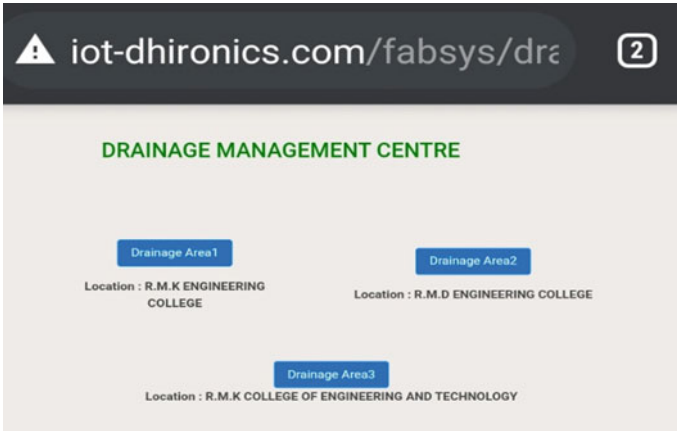


Fig. 5 Drainage selection webpage

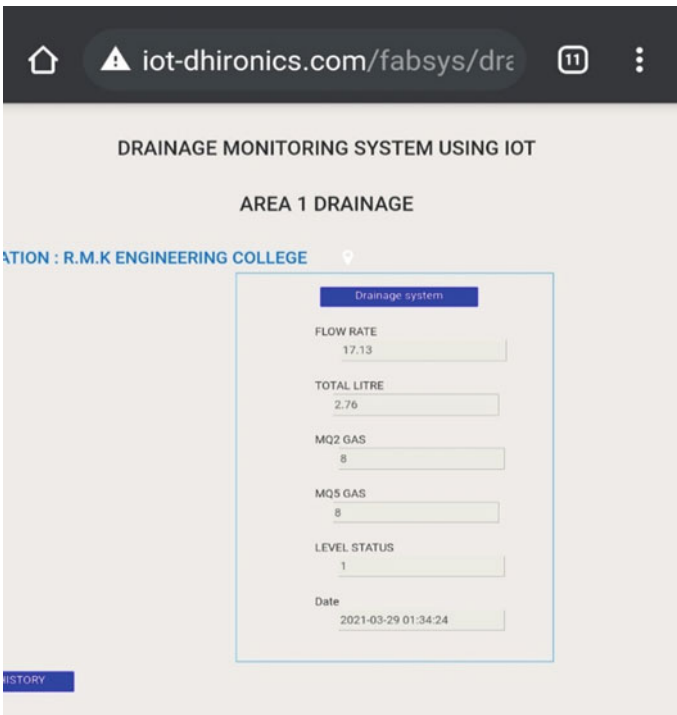


Fig. 6 Output indication in IoT page for case 1

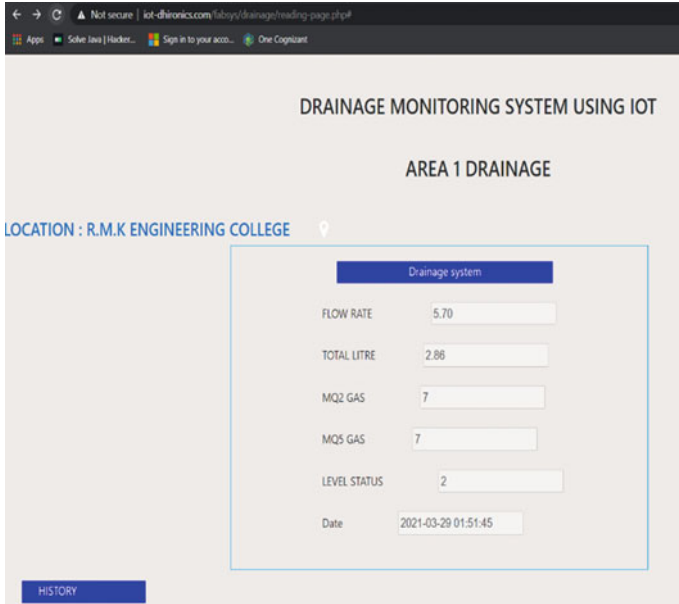


Fig. 7 Output indication in IoT page for case 2

3.1 Case 1: Drainage Status 1

If the drainage Level filled 75–100%, then it is indicated as level status = 1 in IoT page as well as the total litres of water present in the drainage along with the toxic gases is updated in the IoT webpage which is shown in Fig. 6.

3.2 Case 2: Drainage Status 2

In case 2, drainage water Level 50–75% filled which is indicated as level status = 2, toxic gas along with the flow rate is updated in the webpage as shown in Fig. 7.

3.3 Case 3: Drainage Status 3

In case 3, drainage water Level 25–50% filled which is indicated as level status = 3, the flow rate differs based on the litres present in the drainage and the toxic gas is detected and updated in the webpage as shown in Fig. 8.

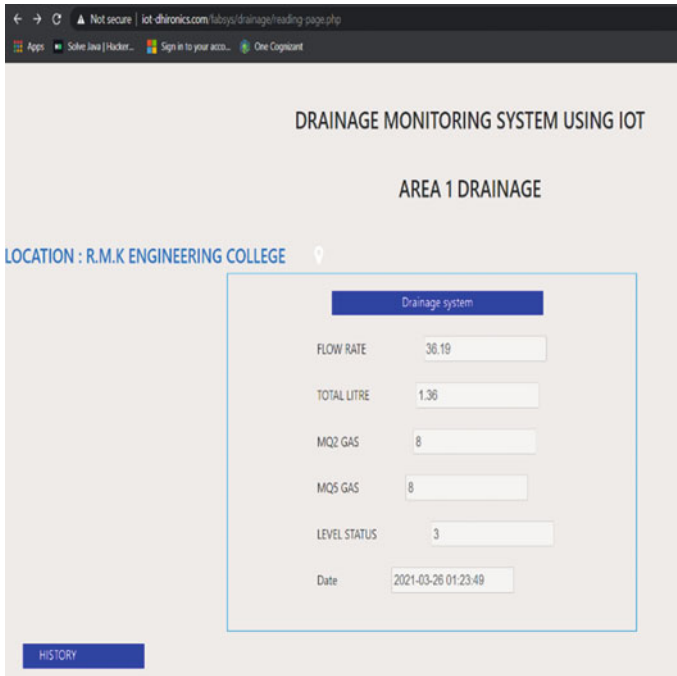


Fig. 8 Output indication in IoT page for case 3

3.4 Case 4: Drainage Status 4

In case 4, water Level 0–25% filled which is indicated as level status = 4 in IoT page which is shown in Fig. 9 and this case tested only for toxic gas and the water level condition and it's updated in the IoT webpage (Figs. 10 and 11).

When an overflow is detected, the connected pump starts to pump the sewage water to the next drain for a pre-mentioned brief period of time, until manual assistance can be deployed. This prevents any unwanted leakage that may cause public nuisance and accidents. The toxic gas detection is intimated in the LCD display as shown in Fig. 12.

If overflow or toxic gas detected, then the location of particular drainage is informed to the sanitation worker through message as well as the shortest path also shared with them using the map icon in IoT webpage. If the icon is selected, it will direct sanitation workers to the Google maps which provide optimal path to reach the affected area quickly.

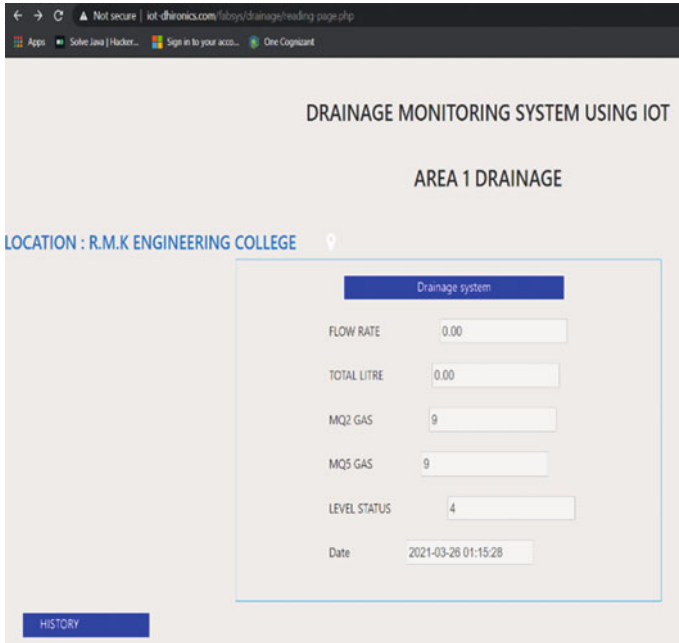


Fig. 9 Output indication in IoT page for case 4

4 Conclusion

The goal of our proposed system is to develop a safe and healthy environment by modernizing a smart drainage monitoring system with help of IoT applications for metropolitan cities. The real-time situation of the drainage system is monitored by using various sensors like gas sensor, ultrasonic sensor and flow sensor. With the support of smart and people friendly proposed drainage monitoring system, problem in the drainage can be detected early and the alert message or information will be update in IoT page for monitoring purposes. As well as, optimal path between location of blocked drainage and sanitation worker location is indicated using Google map to reach the affected area quickly.

HISTORY TO DRAINAGE MONITORING SYSTEM

TEST

Back

FLOWRATE	TOTALLITRE	MQ2 GAS	MQ3 GAS	LEVEL	DATE & TIME
35.19	1.36	0	0	3	2021-03-26 01:23:45
35.19	1.36	0	0	3	2021-03-26 01:23:45
35.19	1.36	0	0	4	2021-03-26 01:23:46
35.19	1.36	0	0	3	2021-03-26 01:23:46
35.19	1.36	0	0	4	2021-03-26 01:23:46
35.19	1.36	0	0	4	2021-03-26 01:23:46
35.19	1.36	0	0	3	2021-03-26 01:23:46
35.19	1.36	0	0	4	2021-03-26 01:23:47
35.19	1.36	0	0	2	2021-03-26 01:23:47
35.19	1.36	0	0	2	2021-03-26 01:23:47

Fig. 10 Previous history of a drainage route

Fig. 11 Overflow alert on LCD



Fig. 12 Toxic gas alert shown on LCD



References

1. S. Vuma, Internet of things (IoT) based smart agriculture in India: an overview. *J. ISMAC* **3**(1), 1–15 (2021). <https://doi.org/10.36548/jismac.2021.1.001>
2. S.R. Mugunthan, Decision tree based interference recognition for fog enabled IOT architecture. *J. Trends Comput. Sci. Smart Technol.* **2**(1), 15–25 (2020). <https://doi.org/10.36548/jtcsst.2020.1.002>
3. S. Manoharan, Sathish, Early diagnosis of lung cancer with probability of malignancy calculation and automatic segmentation of lung CT scan images. *J. Innov. Image Process.* **2**(4), 175–186 (2020). <https://doi.org/10.36548/jiip.2020.4.002>
4. D.S. Shakya, Computational enhancements of wearable healthcare devices on pervasive computing system. *J. Ubiquitous Comput. Commun. Technol.* **2**(2), 98–108 (2020). <https://doi.org/10.36548/jucct.2020.2.005>
5. A. Zanella, N. Bui, A. Castellani, L. Vangelista, M. Zorzi, Internet of things for smart cities. *IEEE Internet Things J.* **1**(1), 22–32 (2014). <https://doi.org/10.1109/JIOT.2014.2306328>
6. P. Tiwari, IoT based smart sewage monitoring system using GSM and Wi-Fi module. *Int. J. Innov. Sci. Res. Technol.* **6**(5), 955–958 (2021)
7. S. Foorginezhad et al., Recent advances in sensing and assessment of corrosion in sewage pipelines. *Process Saf. Environ. Prot.* **147**, 192–213 (2021). <https://doi.org/10.1016/j.psep.2020.09.009>
8. M.K. Naidu, N. Chowdary, L. Sai, P. Krishna, Smart drainage monitoring system using IoT. *J. Xi'an Univ. Archit. Technol.* **12**(4), 3186–3194 (2020)
9. S.V. Sai Bharath, T. Sidharth, S. Kaviti, B. Balachander, Drainage monitoring system using IoT (DMS). *Indian J. Public Health Res. Dev.* **8**(4), 1084–1087 (2017). <https://doi.org/10.5958/0976-5506.2017.00472.7>
10. G. Chandhini, B. Chithra, P. Kiruthikadevi, B. Sasi, V.K. Kumar, IoT based underground drainage monitoring system. *Int. J. Recent Technol. Eng.* **9**(3), 247–249 (2020). <https://doi.org/10.35940/ijrte.c4354.099320>
11. M. Gunasekaran, S. Pavithra, R. Priyanka, M. Reeva, IoT-enabled underground drainage monitoring system using water flow sensor. *Int. Res. J. Eng. Technol.* 2427–2430 (2019)
12. G. Sonawane, C. Mahajan, A. Nikale, Y. Dalvi, Smart real-time drainage monitoring system using internet of things. *IRE J.* **1**(11), 1–6 (2018). [Online]. Available: <http://irejournals.com/formatedpaper/1700668.pdf>

13. R. Vijayalakshmi, IOT based smart detection system for harmful gases in underground sewages. *Int. J. Res. Appl. Sci. Eng. Technol.* **V(XI)**, 604–614 (2017). <https://doi.org/10.22214/ijraset.2017.11095>
14. V.D.K. Ambeth, D. Elangovan, G. Gokul, J.P. Samuel, V.D.A. Kumar, Wireless sensing system for the welfare of sewer labourers. *Healthc. Technol. Lett.* **5(4)**, 107–112 (2018). <https://doi.org/10.1049/htl.2017.0017>
15. L.K. Hema, S. Velmurugan, S. Pa, R. Indumathi, Smart manhole toxic gas identification and alerting system. *Int. J. Recent Technol. Eng.* **8(3)**, 507–510 (2019). <https://doi.org/10.35940/ijrte.A2150.098319>
16. R. Girisrinivaas, Drainage overflow monitoring system using IoT (DOMS), in *2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering*, pp. 2133–2137, 5094 (2017)
17. K.V.D. Ambeth, Human security from death defying gases using an intelligent sensor system. *Sens. Bio-Sens. Res.* **7**, 107–114 (2016). <https://doi.org/10.1016/j.sbsr.2016.01.006>
18. P.M. Bach, W. Rauch, P.S. Mikkelsen, D.T. McCarthy, A. Deletic, A critical review of integrated urban water modelling—urban drainage and beyond. *Environ. Model. Softw.* **54**, 88–107 (2014). <https://doi.org/10.1016/j.envsoft.2013.12.018>
19. R.T. Wahyuni, Y.P. Wijaya, D. Nurmalasari, Design of wireless sensor network for drainage monitoring system. *Innov. Syst. Des. Eng.* **5(5)**, 6–14 (2014)
20. H.A. Obaid, S. Shahid, K.N. Basim, C. Shreeshivadasan, Modeling sewerage overflow in an urban residential area using storm water management model. *Malays. J. Civ. Eng.* **26(2)**, 163–171 (2014)
21. M. Sk, S. Rao, Automated internet of things for underground drainage. *Int. J. Inf. Comput. Technol.* **4(12)**, 1211–1220 (2014)

Earliest Deadline First (EDF) Algorithm Based Vaccination Management System



M. Karthigha, R. Pavithra, and C. Padmavathy

Abstract The primary goal of this research work is to develop a smartphone application that allows parents to learn about their children's vaccination information and keep track of their immunization schedule. Parents can log in with their credentials and upload information about their children. Appointments with paediatricians may be scheduled using the mobile application's real-time scheduling algorithm to get back the confirmation from the doctor. Further, a mobile a vaccination alert will be sent to the parents via the mobile application notification. Doctors can also know the child details and appointment details in their portal and Parents will be informed about their children's vaccination details from their portal. The developed mobile application is simple and user friendly.

Keywords Mobile application · Vaccination · Earliest deadline first · Scheduling

1 Introduction

A vaccination boosts our immune system without making us sick. Many deadly illnesses can be averted in this simple and efficient way. From birth, we are continually exposed to a broad variety of viruses, bacteria, and other microorganisms. Most of them are not harmful and beneficial but some can cause disease. The immune system of our body helps us to protect ourselves from many diseases. When we have a disease, we often develop lifelong immunity. The main aim of vaccination is to build the necessary immunity without any kind of risk to human life. For some diseases, vaccination provides lifelong protection, while for others the effect is reduced after a period of time. Then, the person must again take the second dosage of the particular

M. Karthigha (✉) · R. Pavithra · C. Padmavathy
Sri Ramakrishna Engineering College, Coimbatore, India
e-mail: karthiga.m@srec.ac.in

R. Pavithra
e-mail: pavithra.r@srec.ac.in

C. Padmavathy
e-mail: padma.dhansh@srec.ac.in

vaccine. In olden days, people were not aware of the importance of vaccination. Due to which many children were affected by various diseases like polio attack, measles, etc. Young children are more vulnerable to infectious illnesses because their immune systems have not yet acquired the essential defence to combat major infections and disorders. As a result, the illnesses like whooping cough and pneumococcal disease are highly hazardous, if not deadly, to infants and young children. Vaccinations start early in life to protect children before they are exposed to these diseases. Vaccines are provided for infants and children under the age of adolescence in a variety of ways, including attending school manually or at hospitals, and there is a chance that some children and infants will miss out on getting vaccinated due to some mandatory reasons that result in death or are affected by various diseases such as polio, etc. Vaccination usually begins at six weeks of age. The occurrence of infections in early life can lead to poor growth and stunting, which in turn adversely affect adult health, cognitive capacity, etc. [3, 4]. Malnutrition, infection, pregnancy and birth complications, and under-stimulation during the first 1000 days of life can have long-term consequences for health, cognitive, and economic outcomes well into old age. In addition to proper nutrition and nurturing, health interventions such as routine vaccinations could reduce the burden of infectious diseases in early childhood, thereby breaking the intergenerational cycle of poverty, poor health, and so on [5–7]. To overcome the above mentioned problems, the proposed application will be an optimal solution to track the vaccination schedule of the children and to get effortless appointment with the doctor.

2 Literature Review

Vidhya et al. [2] have utilized the infant's biometric traits (fingerprint) to store vaccination schedule details, thereby automating the infant's vaccination schedule. Biometrics traits are used since infant fingerprints have a high potential for accurately recording immunization and aid in the efficient search of data. Based on the vaccination schedule information that has been saved, the proposed method aims to create an application that sends out regular alerts to parents and Accredited Social Health Activist (ASHA) workers.

Negandhi et al. [1] have employed a novel implementation process. Methods: In 2015, quantitative data on vaccine supply chain management indicators were collected by using factsheets and dashboards, which represented the state of the vaccine supply and cold chain management system at regular intervals since its inception. In-depth interviews were conducted with programme specialists to learn about the initiative's origin, challenges, and strengths.

Penney et al. [14] demonstrated how the technology behind these Bluetooth exposure notification applications may be used to efficiently prioritize vaccination allocation. It has been demonstrated that a "hot-spotting" technique can produce herd immunity with less than half the number of vaccinations used as compared to dispersing doses equally throughout the population. Hasan et al. [10] emphasize the

significance of implementing a smarter mechanism to improve the situation. This paper has developed an Android application to address this issue. This software provides a means for sharing information, keeping track of records, and assisting parents in scheduling immunization visits for their children.

Abusair et al. [11] improved the client's experience when seeking for non-critical services. The report utilizes a queue management system to schedule appointments. Customers in line are split into numerous priority classifications, which are considered while calculating the projected wait time. We used healthcare vaccination to demonstrate the efficacy of the suggested technique.

Numnark et al. [12] created as a web application and have the following notable features. For starters, it includes graphical user interfaces that visualize different perspectives on vaccine-related data from social media. Second, it has a set of criteria that allow users to narrow their search to diseases, vaccinations, countries, and/or firms that they are interested in. Finally, it includes the assistance.

Odoom et al. [13] have proposed a solution based on blockchain and smart contracts that allows authorized entities to change the status of the system. They utilized Inter-Planetary File System (IPFS) distributed storage technology to store user encrypted records and retrieve them for verification requirements.

2.1 Existing System

Vaccination is an essential component of human life. Child vaccination administration is a time-consuming process. To protect a child from various diseases, he or she will require a large number of vaccinations. Vaccines are provided for infants and children under the age of ten in various ways, such as by attending school manually (or) at hospitals and there is a chance that some of the children and infants will miss out on getting vaccinated due to some mandatory reasons [8–13].

2.2 Proposed System

In order to make the child vaccination, management easy for a parent in the proposed system the parent can register in the application by giving the required details and apply for vaccination when required. Before applying for vaccination, the user must first log into the application. It is not necessary to give all details during every login. The user can login by entering the registered mobile number and password, which was specified at the time of registration. Users can apply for vaccinations and general consultations after logging in. While applying for vaccination, the user must specify which vaccination he wants or if he is applying for general consultation he/she must specify the reason for consultation. Application sent by the user will be saved in the portal and the confirmation of appointment will be sent as a message to the user

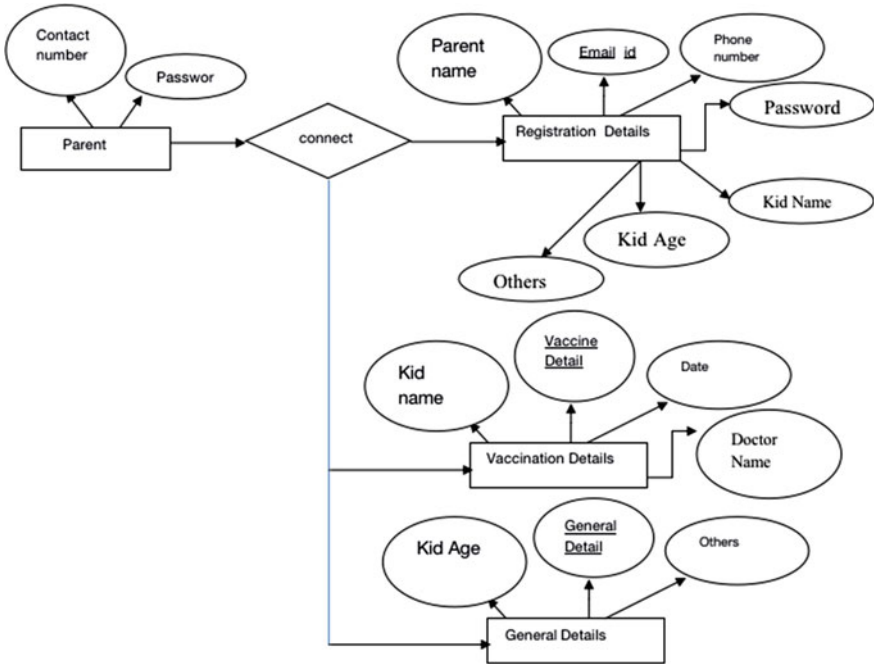


Fig. 1 Block diagram of vaccination management system

in real-time scheduling algorithm. The proposed application has following phases (Fig. 1).

2.3 Parent Registration

In this module, the user can register/sign up in the application by giving his/her details like name, e-mail id, contact number, password, etc. The user will be able to apply for vaccinations or general consultations after completing the registration process. After the user has entered his/her details, the doctor or any technical staff in hospital or nursing home can view the details entered by the user. It is not necessary of the user to enter all the details during every login. He/she can sign in into the application by entering the registered mobile number and the password, which was specified during registration.

2.4 Application for Consultation/Vaccine

The user sign into the app and request for a vaccine or general consultation. While fixing an appointment, the user must specify the name of the child, age, name of the parent and the reason for appointment. The request for appointment of the user will be saved in the database immediately. When making an appointment, the parent/user should specify the name of the vaccine that is required by his or her child. Adults can also apply vaccines for various infections such as malaria, dengue, etc. Not only may the user make appointments for vaccinations but he or she can also schedule appointments for general consultation.

2.5 Vaccination Details

The vaccination details will be uploaded, and later, it will be saved in the application. The vaccination chart for the babies will be uploaded. The chart consists of the list of vaccines to be given to the baby right from the first week upto 15 years. It also consists of the details regarding the number of vaccine doses. Uploading the vaccination chart into the application may help the parent when applying for the vaccine. Before submitting an application for vaccination, the user should review the vaccination chart. Along with the newborn immunization chart, the application will provide a list of infectious diseases and their vaccinations, which may be valuable for adults as well. For example, vaccinations can be used to prevent diseases such as malaria and dengue fever.

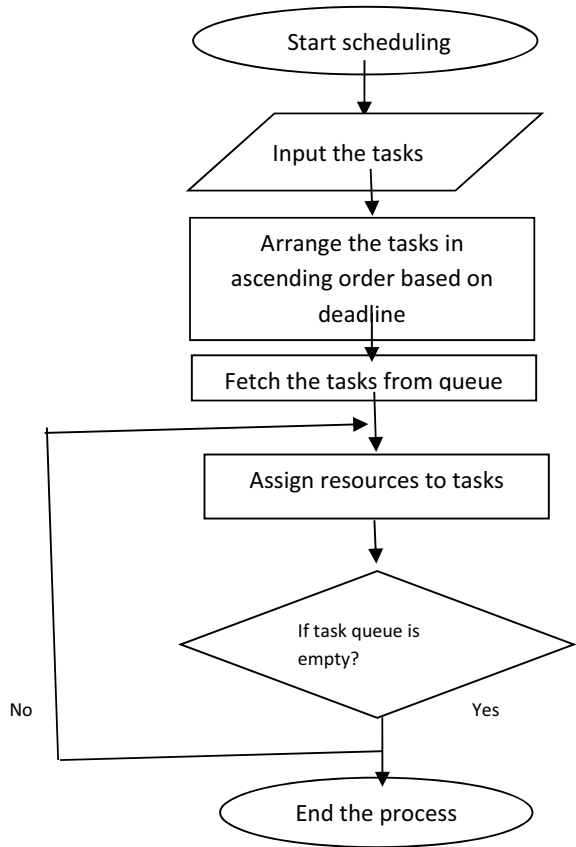
2.6 Appointment Using EDF

The appointments are based on dynamic priority scheduling algorithm and the earliest deadline first (EDF) algorithm. Priorities are assigned to jobs by EDF. It prioritizes the tasks based on the absolute deadline. The assignment with the shortest deadline receives the highest priority. Priorities are assigned and updated on a regular basis. When compared to other scheduling algorithms in real-time systems, EDF is extremely efficient. It may increase CPU utilization to nearly 100% while still ensuring security. When the user applies for an appointment, they are automatically notified with a message. If two or more applications come at the same time, the message will be sent on the EDF basis. The EDF schedulability requirement can be written as follows for a set of periodic real-time tasks T_1, T_2, \dots, T_n :

$$\sum_{i=1}^n e_i / p_i = \sum_{i=1}^n u_i \leq 1 \quad (1)$$

EDF is explained in Fig. 2. In this, the request from parent for child vaccination are considered as tasks. Based on the parent request, the previous vaccination details and the date of vaccination will be collected. Appointment with the doctor will be allocated based on priority for the available slots. The priority is given to the child, who is immediately supposed to take the vaccine as they are nearing their due date. For example, the second dose of Rotavirus vaccine will be administered in the fourth month. As a result, if more requests are submitted on a specific day, the child who is nearing the due date will be granted an appointment.

Fig. 2 Flow diagram of EDF algorithm



2.7 General Alert

On successful completion of this module, the user will get a notification message with confirmation along with date and time for the appointment. Here, the messaging system will work on the basis of the queue method.

3 Results

See Figs. 3, 4, 5, 6, 7, 8 and 9.

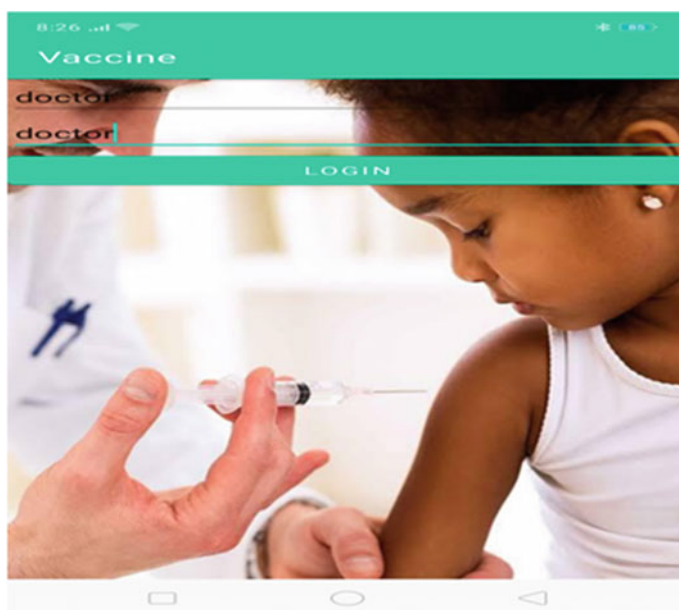


Fig. 3 Doctor login

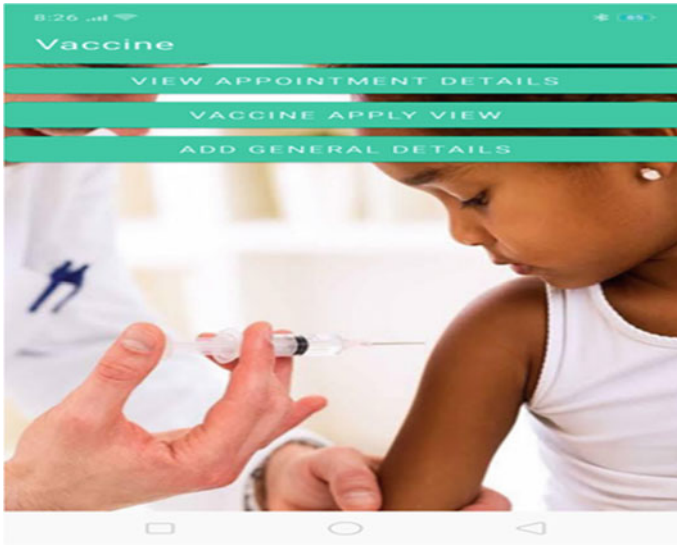


Fig. 4 Appointment details page

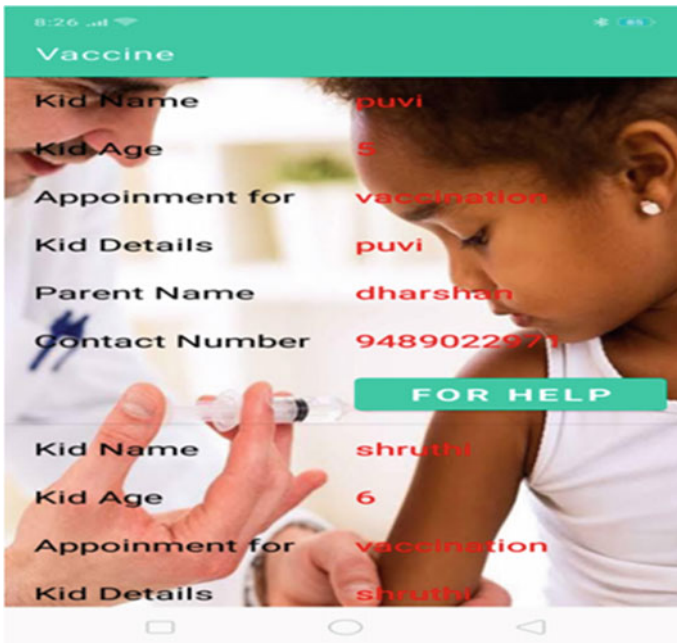


Fig. 5 Appointment details

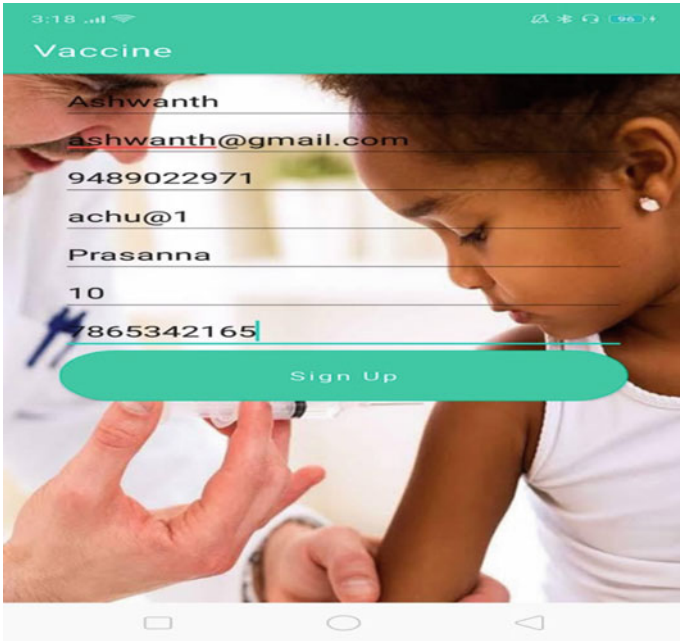


Fig. 6 Sign up page

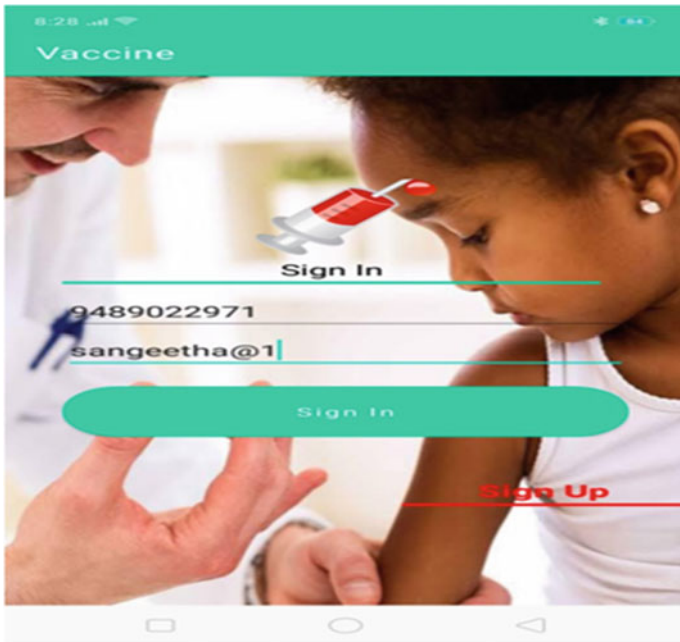
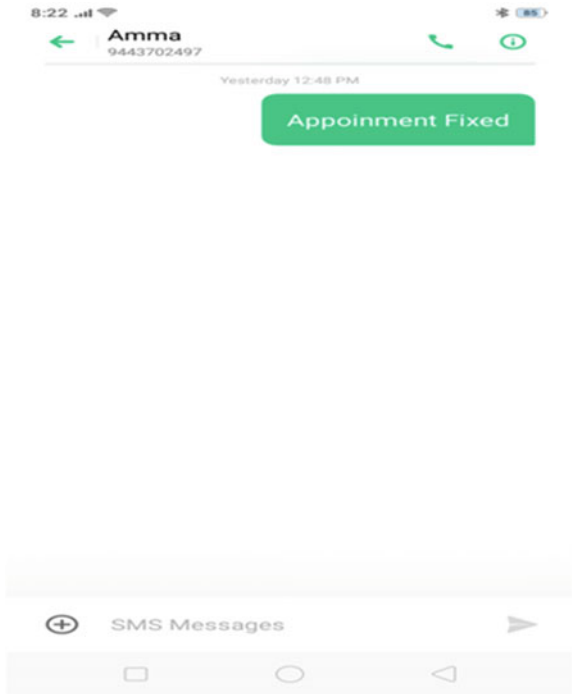


Fig. 7 Parent login

National Immunization Schedule		
When to give	Dose	Route
For Infants		
At birth or as early as possible till one year of age	0.1 ml (0.05ml until 1 month of age)	Intra -derm
At birth or as early as possible within 24 hours	0.5 ml	Intramusc
At birth or as early as possible within the first 15 days	2 drops	Oral
At 6 weeks, 10 weeks & 14 weeks	2 drops	Oral
14 weeks	0.5 ml	Intramusc
At 6 weeks, 10 weeks & 14 weeks	0.5 ml	Intramusc
At 6 weeks, 10 weeks & 14 weeks	5 drops	Oral
9 completed months-12 months. (give up to 5 years if not received at 9-12 months	0.5 ml	Subcutane

Fig. 8 Vaccination details

Fig. 9 SMS alert



4 Conclusion and Future Enhancement

Improper vaccination coverage remains the underlying cause of rising vaccine-preventable illnesses and, as a result, an increase in infant mortality rates (IMR). To maintain a track of infant vaccination schedules, an effective immunization programme is required. The suggested approach monitors infant vaccinations. The parents can apply and keep their children protected from various diseases. The proposed automatic messaging system has been successfully utilized to notify the user/parent of the date and time of the doctor's appointment via message. This approach will take less time than the traditional method, and the rate of immunization dropout will be lowered as well. With the further enhancement of this project, the rate of children taking vaccination can be increased. Further, it can be developed into a chatbot and even payments can be priorly executed. In today's fast moving world, people don't find time to register in an application and fix a manual appointment. The proposed application can be further enhanced to incorporating a chatbot.

References

1. P. Negandhi, M. Chauhan, A.M. Das, S.B. Neogi, J. Sharma, G. Sethy, Mobile-based effective vaccine management tool: an m-health initiative. Implemented by UNICEF in Bihar (2016)
2. S. Vidhya, B.A. Sabarish, J. Rajiv Krishnan, P. Sachin, e-vaccination: fingerprint based vaccination monitoring system (2018)
3. A.H.A. Almohamed, Y. Yusof, *Children Vaccination Reminder Via SMS Alert* (World Health Organization, 2020)
4. B.K. Siang, A.R.B. Ramli, V. Prakash, S.A.R.B.S. Mohamed, *SMS Gateway Interface Remote Monitoring and Controlling Via GSM SMS* (IEEE, 2000), pp. 84–87
5. C. Soriano, G.K. Raikundalia, J. Szajman, *A Usability Study of Short Message Service on Middle-Aged Users* (ACM, 2005)
6. M. Nasir, H. Hassan, N. Jomhari, The use of mobile phones by elderly: a study in Malaysia perspectives. *J. Soc. Sci.* **4**, 123–127 (2015)
7. L. Zhao, X. Chen, J. Ding, *Interference Clearance Process of GSM-R Network in China* (IEEE, 2010), pp. 424–428
8. R.T. Hasanat, M.A. Rahman, N. Mansoor, N. Mohammed, M.S. Rahman, M. Rasheduzzaman, An IoT based real-time data-centric monitoring system for vaccine cold chain, in *2020 IEEE East-West Design & Test Symposium (EWDTS)* (2020)
9. C. Antal, T. Cioara, M. Antal, I. Anghel, Blockchain platform for COVID-19 vaccine supply management. *IEEE Open J. Comput. Soc.* (2021)
10. S. Hasan, M.M. Yousuf, M. Farooq, N. Marwah, S.A.A. Andrabi, H. Kumar, e-vaccine: an immunization app, in *2021 2nd International Conference on Intelligent Engineering and Management (ICIEM)* (2021)
11. M. Abusair, M. Sharaf, T. Hamad, R. Dahman, S. AbuOdeh, An approach for queue management systems of non critical services, in *7th International Conference on Information Management (ICIM)* (2021)
12. S. Numnark, S. Ingriswang, D. Wichadakul, VaccineWatch: a monitoring system of vaccine messages from social media data, in *8th International Conference on Systems Biology (ISB)* (2014)

13. J. Odoom, R.S. Soglo, S.A. Danso, H. Xiaofang, A privacy-preserving Covid-19 updatable test result and vaccination provenance based on blockchain and smart contract, in *International Conference on Mechatronics, Remote Sensing, Information Systems and Industrial Information Technologies (ICMRSISIT)* (2020)
14. M. Penney, Y. Yargic, L. Smolin, Vaccine prioritisation using Bluetooth exposure notification apps (2021)

Using Computer Vision to Detect Violation of Social Distancing in Queues



Muhammed Ismail, T. Najeeb, N. S. Anzar, A. Aditya, and B. R. Poorna

Abstract On March of 2020, World Health Organization (WHO) declared COVID-19 as a global pandemic after months of infecting and claiming many victims. There are some ways by which we can safeguard ourselves against the virus and thereby controlling the spread of the virus. They are following proper sanitization by washing hands with soap regularly, wearing masks and following social distancing, while being present in public places. Social distancing refers to maintaining at least 6 feet of distance between other people. But the main problem is that most of the people ignore these rules and hence the spread of the virus can not be controlled. The project uses computer vision in order to ensure that social distancing is being followed properly, thus helping to reduce the number of victims that the virus may claim. Computer vision is a field of computer science that deals with how computers can gather knowledge and learn from images and videos. It is a rapidly growing field of science thanks to the many advancements in technology over the past few years such as increase in processing power of computers and the exponential increase in data being available nowadays. The system works by taking input from CCTV or other similar image source and then processing the input to find out if any people violate the rules of social distancing and if any violations are detected, the system will consist of an alert module which will alert the respective authorities regarding the violation so that they can do the needful.

Keywords Social distancing · Computer vision · YOLOv4

Supported by Mar Baselios College of Engineering and Technology.

M. Ismail (✉) · T. Najeeb · N. S. Anzar · A. Aditya · B. R. Poorna
Mar Baselios College of Engineering and Technology, Mar Ivanios College Rd, Nalanchira,
Thiruvananthapuram, Kerala 695015, India
e-mail: mbcet@mbcet.ac.in

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
S. Smys et al. (eds.), *Inventive Computation and Information Technologies*, Lecture Notes
in Networks and Systems 336, https://doi.org/10.1007/978-981-16-6723-7_57

771

1 Introduction

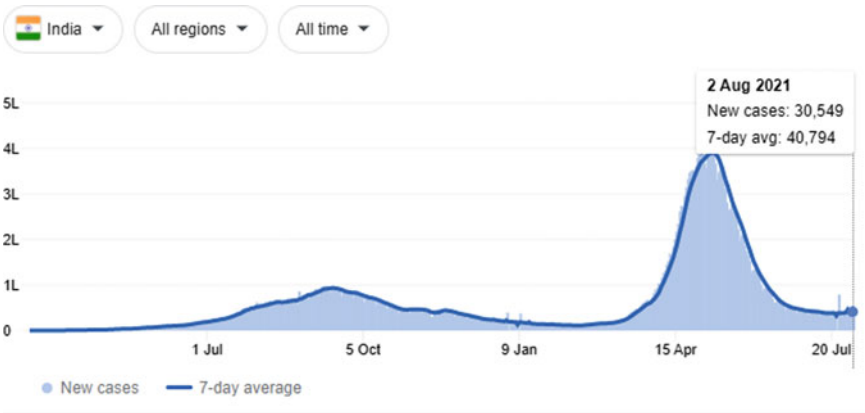
The first few cases of COVID-19 [1] started appearing in November 2019 in Wuhan province of China and by March 2020, the World Health Organization (WHO) declared COVID-19 as a pandemic, which refers to an outbreak or a disease that has crossed international borders and have affected many number of people. The virus spreads by means of contact which may be either direct or indirect, or by being in same breathing space as another infected individual.

Figure 1 shows the number of cases reported in India over the past month, and Fig. 2 shows the all time number of cases reported. From both the graphs, it is clearly visible that the daily cases being reported have still not gone down significantly after the second wave, while the death rate or the number of people who have died due to



Each day shows new cases reported since the previous day · [About this data](#)

Fig. 1 Graphical representation of reported cases of COVID-19 in India during past month



Each day shows new cases reported since the previous day · [About this data](#)

Fig. 2 Graphical representation of reported cases of COVID-19 in India of all time

COVID-19 has increased as the days passed by. There has been prediction of a Third Wave coming soon and if we take a closer look at the graph we can see that the peak has slowly started to rise again.

The Health department have issued many guidelines to follow so that the spread of virus can be controlled. Some of them are: Following proper sanitation by washing hands with soap regularly, only leave your homes in case of emergency and wear masks and follow social distancing, while being present in public places such as shops, railway stations, bus stands, banks, etc. But the main problem is that majority do not follow these guidelines properly, thus increasing the likelihood of spreading the virus.

The project aims to help the authorities reduce the spread of the virus by helping them find out if people are following the protocols or not and if any violation is detected, the system will alert the concerned personnel so that they can do the needful. We focus mainly on detecting violation of social distancing. Social distancing is a practice that requires an individual to maintain a minimum safe distance of at least 6 feet from others while being present in public places so as to reduce the risk of transmitting virus from one person to another.

This is done by using Computer Vision [2], which is a field of Computer Science that deals with processing images and videos and extracting data from these input so that it can be applied for solving real world problems. There has been many advancements in the field of Computer Vision mainly due to two reasons. One is the advancements happening in computational hardware that led to increase in processing power which further led to increase in processing speed and decrease in processing time. Another reason is the availability of huge amount of data which is way more compared to data available when computer vision started out as a concept thanks to the wide accessibility of both hardware and a medium to share data such as the Internet.

The project works by taking input from a source such as CCTV and processing the input to see if any pedestrians present in the input stream have violated the rules of social distancing by not maintaining the minimum safe distance of 6 feet and if any such violation is detected, the alert module present in the system will alert those concerned of the violation so that they can do the needful.

2 Literature Review

From the time when COVID-19 was declared as a global pandemic, there has been lot of research and study regarding the ways by which we the spread of COVID-19 virus could be controlled. Some researchers focused on a way to help develop the vaccine to the virus sooner by using deep learning and computational power, while some others focused on the socio-economic impacts due to the virus and on finding a way to bring back the society to the order and prosperity it enjoyed before the virus. Out of the many curious individuals, some of them also focused on social distancing and its effectiveness on reducing the spread of COVID-19 virus and developed certain systems for the same.

Ghodgaonkar et al. [3] focus on analysing how social distancing was carried out in different parts of the world and studying about how the various restrictions made by different governments affected the people's behavior in following social distancing. Rezaei and Azarmi [4] created a system which used YOLOv4 to develop a monitoring tool for social distancing and achieved an average precision score or AP score of 99.8 processing input feed. Yang et al. [5] is by far the system that shares most similarity with current proposal in the manner that their project detects breach in social distancing by using different object detectors such as Faster R-CNN, YOLOv3 [6], etc., and provides an audio/visual based alert system which gives out a general alert when the number of violations becomes moderate and a more severe alert when the number exceeds a threshold value.

The system's difference when compared to the already existing techniques mainly lies in the fact that it is intended to be deployed in public places where people gather such as shops, schools, railway stations, etc., and the project focuses on a particular type of scenario. The situation which we are trying to focus is that of a queue system such as ticket queue where people need to stand in a line for some time such as when visiting banks, waiting for checkout at shops, etc., and during this time, special care needs to be taken so as to avoid coming in close contact with others. This scenario needs to be addressed especially since in many states movie theaters have started functioning again which is one of the most common places where people usually stand in a queue. Thus, the proposed system aims to add additional functionalities to the already existing systems so that it could perform more tasks when deployed and do so with maximum efficiency.

3 Implementation

The system works by first taking input from an image/video source such as CCTV system or other similar alternatives. The output coming from the source will be either in .mp4 or .avi format. The input will be fed into the program using opencv which provides built-in functions for doing the same. The only problem is that the processing will be done in BGR color space, which also can be remedied by converting the color space of the video at a later stage. The video stream obtained will be processed for removing any noise or unwanted artifacts if present. This can be done using filters such as median filters. This processed input will be fed as input to an object detector for detecting pedestrians present in the input. The detector we chose for the system is YOLOv4 [7] since it runs extremely fast compared to other detectors such as Faster R-CNN [8] and Histogram of Oriented Gradients (HOG) [9], while not compromising on accuracy, hence a perfect candidate for applying to real world scenarios. After detecting the number of people present in the given input frame, distance will be calculated between each pair of pedestrian using a distance measure, which for the system we have decided to use Euclidean Distance measure. Then, we will check if any pair violate the minimum safety distance of 6 feet and for every violation present a timer variable will be maintained unique to that particular violation. If any timer

variable exceeds a threshold value, then that violation will be alerted to the concerned authorities so that they can take necessary action like warn them and remind them to follow COVID-19 protocols. A threshold value is set so that those contacts which may happen only for few seconds can be avoided and the system can focus on more severe violations.

3.1 Violation Detection Module

This is the main crux of the system. Firstly, we load the input video stream onto the program and then run it through YOLOv4 object detector. YOLOv4 is pre-trained to detect almost 83 classes of objects, but we are only looking for the pedestrians detected within the given input frame at any given time. So in the next step we filter out the detection to only those which fall under the label "person." We proceed to the next stage only if the number of detections is not null. From the output of detection done by the system, we get the x and y coordinates for the midpoint of the bounding box drawn around the person detected as well its width and height. From these points, we obtain the coordinates of the corner points of the bounding box, since they are needed later for drawing bounding box around the detected people in color, which will be green for those who are safe and red for those who are not 2 feet apart. After finding out the desired coordinates, they are added onto a dictionary with key being an integer initialized with 1 and incremented on addition of a new person onto the dictionary.

In the next stage, we find out all possible combinations of pedestrians present in a given input frame, with the number of pedestrians considered at a time set as 2, so as to find the distance between them using the combinations function provided in `itertools` package. From the dictionary that we created earlier, we calculate the difference between the x and y coordinates of the midpoint of the bounding boxes of the pedestrians currently in consideration and using this compute the distance between the said pedestrians using Euclidean Distance measure which was discussed in the previous section. Following step deals with checking if the distance is less than the minimum safe distance to be followed. Here, the minimum safe distance threshold is not a universal value and will vary according to the situation where the system is to used. This variance occurs due to the fact that when we use Euclidean Distance to calculate the distance measure between the pedestrians, we are actually calculating the number of pixels between them, which will vary according to the resolution in which the video is displayed, the location of the camera, while taking the video, etc. Once the system has been setup and the distance threshold has been identified manually, further detection of violations will happen automatically. If the pedestrians are not maintaining the minimum safe distance, they will be added to a separate list which indicates that they are in danger.

After creating the list which contains the key value and other necessary details of pedestrians from the original dictionary, the next step is to draw bounding box around them and display them in an output window. For this, we have already found

out the necessary coordinates for drawing the rectangular box and we input these coordinates into a function provided by OpenCV library along with other parameters such as thickness of the line, color of the lines, whether it be green for those who are safe or red for those who are in danger, etc. The total number of violations detected at any given time will also be displayed in the output window.

3.2 Alert Module

For the alert module, we make use of an Arduino Uno Microcontroller Board because of its functionality, support network and easiness to work with. The board consists of both analog and digital pins, although for this system we only require the use of digital pins. Normally, the boards manufactured by Arduino are used in conjunction with the IDE that they provide. But the IDE supports only C and C++ and the project is written entirely in python. So in order to interface the Arduino Board with the program, we used the PySerial package which helps programs to access the serial port. By specifying the port to which the Board is connected in the system, we can create a channel for communicating with the Board. Due to the COVID-19 and subsequent Lockdown restrictions, the initial plan of using an LCD display along with a buzzer system failed because of the lack of resources available. Instead we repurposed the alert module with the help of basic LEDs. The module will now blink the LED's connected to it when the number of violations have exceeded a predefined threshold value.

4 Results and Discussion

Before discussing about the results, there are certain factors that should be considered. The measure that we used to evaluate the system is the Average Frames per Second or FPS, which refers to the number of frames being processed in one second. This is because in real world situations, We gave frame rate more importance compared to other measures because depending upon the frame rate by which the input video is being processed, the output can be either fast enough to keep up with actual use case situations or be slow enough that it is not possible to use it for getting real time output. Another factor to consider is the system which is being used for running the project. Depending upon the processor of the system, the availability of accelerators such as GPU's and the type of GPU's used, the rate of processing videos differs rapidly with more powerful ones being the better option. The configuration of system we used for testing is shown in Table 1.

Using a system having the above specifications, we were able to achieve an Average Frame Rate of 25 FPS, which is considered an above acceptable standard compared to the 30 Frames per Second normally used for real time processing.

Table 1 Specification of system used for testing and evaluation

Processor	AMD Ryzen 7 3700x
Number of Processing Cores	8
RAM	32 GB DDR4 3200MHz
GPU	NVIDIA RTX 2070 Super
Number of CUDA Cores	2560

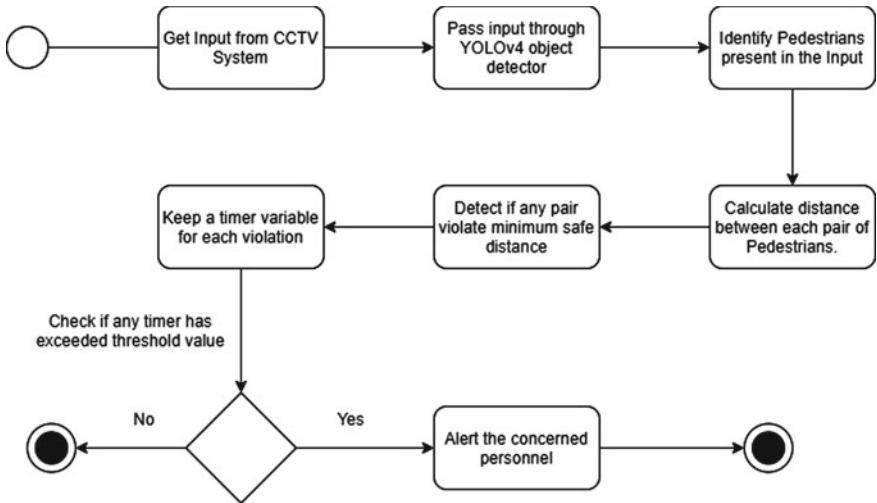


Fig. 3 Activity Diagram



Fig. 4 Output when minimum safe distance is maintained by the people



Fig. 5 Output when a violation is detected by the system

Figure 2 shows an example of a situation where everyone present in the given frame is maintaining the minimum safe distance to be kept. Hence, they are covered by green colored bounding boxes. In Fig. 3, 2 out of the 3 people present in the given frame are not maintaining the safety threshold distance between them and are thus in violation of social distancing. Hence they will be covered in red bounding box to indicate that they are in violation and danger (Figs. 4 and 5).

5 Conclusion

The expected output of the system is that it will be able to identify most if not all the people present in the given input and it is able to do so with a minimum required accuracy. It will also be able to calculate the distance between each pair of pedestrians detected, while accounting for real world parameters such as the angle and position at which the CCTV or input source is situated and the calculated distance value must be approximately equal to the real-world distance between them, provided within an error range of 0.5–1.5 feet. The system would also keep track of each violation and alert the authorities if any violation has exceeded the pre-set threshold value which normally would be set to 10s.

The expectation of developing the system is that the authorities such as health officers, police officers and the concerned people such as shop owners can use the system in order to control the virus from spreading and claiming a greater number of people than it already has. If we take the case of China where the virus first originated, they were able to contain the virus within 9–10 months because the government

enforced COVID-19 protocols strictly and the citizens also abided by the same. The expected outcome from using the system is that it helps the society to also do the same and break out from the grasp of COVID-19 virus or at the least reduce the rate at which the virus spreads until a proper vaccine has been developed, tested and made available to the public.

References

1. T.P. Velavan, C.G. Meyer, The COVID 19 epidemic. *Trop. Med. Int. Health* **25**(3), 278 (2020)
2. D. Barik, M. Mondal, Object identification for computer vision using image segmentation, in *2010 2nd International Conference on Education Technology and Computer*, vol. 2 (IEEE, 2010), pp. V2–170
3. I. Ghodgaonkar, S. Chakraborty, V. Banna, S. Allcroft, M. Metwaly, F. Bordwell, K. Kimura et al., Analyzing Worldwide Social Distancing through Large-Scale Computer Vision (2020). arXiv preprint [arXiv:2008.12363](https://arxiv.org/abs/2008.12363)
4. M. Rezaei, M. Azarmi, Deepsocial: social distancing monitoring and infection risk assessment in covid-19 pandemic. *Appl. Sci.* **10**(21), 7514 (2020)
5. D. Yang, E. Yurtsever, V. Renganathan, K.A. Redmill, Ü. Özgüner, A Vision-Based Social Distancing and Critical Density Detection System for Covid-19 (2020), pp. 24–25. arXiv preprint [arXiv:2007.03578](https://arxiv.org/abs/2007.03578)
6. J. Redmon, A. Farhadi, Yolov3: An Incremental Improvement (2018). arXiv preprint [arXiv:1804.02767](https://arxiv.org/abs/1804.02767)
7. A. Bochkovskiy, C.-Y. Wang, H.-Y.M. Liao, YOLOv4: Optimal Speed and Accuracy of Object Detection (2020). arXiv preprint [arXiv:2004.10934](https://arxiv.org/abs/2004.10934)
8. W. Wu, Y. Yin, X. Wang, X. De, Face detection with different scales based on faster R-CNN. *IEEE Trans. Cybern.* **49**(11), 4017–4028 (2018)
9. N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1 (IEEE, 2005), pp. 886–893

An Efficient Approach Toward Security of Web Application Using SQL Attack Detection and Prevention Technique



Vishal Bharati and Arun Kumar

Abstract SQL injection attacks are widely used by the imposters due to its less complexity and high flexibility. The proposed methodology is intended to perform detection and prevention of such malware scripted SQL queries using SVM. The model first trained with various malware strings and then tested with unknown scripts. It also performs prevention of web application from the SQL malware string using string analyzer and dynamic candidate evaluation. The string analyzer is a grammar-based algorithm that locates the context on the string using regular grammar. Dynamic candidate solution is used to dynamically identifies the malware script using review policy network in which it first generate the parse tree of the input query and then it analyze each node of the tree. It also finds the variation of detection time with respect to accuracy. For the prevention system, simplicity calculates ratio of prevented attack queries out of total number of input queries. The accuracy of the model is good and also the fault rate is minimal.

Keywords SQL injection attacks · Support vector machine · String analyzer · Dynamic candidate evaluation · Blind SQL query · Union query · Piggy-backed query · Cookie injection · Boolean injection

1 Introduction

Malware detection is now becoming very sophisticated task that require digital advancement for making the system protected. The nature of the malware is becoming versatile and is able to change its nature in the data stream. Proxy filters and firewalls are found to be obsolete framework to locate the modified malware. SQL injection malware is found to be very simple and logical malware that may resemble with benign query. It has very versatile nature and is able to join with any other benign query. Most of the SQL injection malware is universally true and provide logical

V. Bharati (✉) · A. Kumar
Centre for Advanced Studies, AKTU, Lucknow, India

A. Kumar
e-mail: drarun@cas.res.in

Table 1 Types of vulnerabilities

Type	Description
Type I	There is no proper distinction between benign and attacked query
Type II	This type of error will occur when there is a delay in analyzing the attacked query in the runtime phase in which variables are considered only
Type III	This type of error will occur when the vulnerability type has not been described. There is a lack to describe the specification of vulnerability
Type IV	This type of vulnerability occurs when the model is not able to validate the input SQL string

nature to procure data from database. The database storage is remain stationary and is located in cloud server. The exchange of the data via IOT devices uses some web application. SQL injection attacks can cause potential damage by exploiting the vulnerabilities of the web application. The SQL injection attack is used to search data from the database through the web application. The database manipulated can easily be done by SQL commands. But the concern rises when the SQL commands perform malicious activity in the back-end database. An example SQL malicious query has been given below:

```
SELECT * FROM employee WHERE category = 'salary' OR 1=1- -' AND transferred= 1
```

The above SQL query seems logical and benign but it says to retrieve all the data from the category 'salary' of the 'employee' table. Since the query says $1 = 1$, this logic is universally true and so system will also interpret it as true and grant the access. So the query will always return the data from the employee table. Here, 'transferred = 1' means that the query will display the salary that is transferred only. It provides the restriction to the un-transferred salary data. Table 1 contain the types of vulnerabilities that may exist in a web application or a system. Chapter scheme of the rest of the paper has been described as follows: Sect. 2 illustrates the proposed methodology. Section 3 accommodates the experimental results. Section 4 will summarize the conclusion/proposal. The final module is reserved for references.

2 Related Work

The process to keep the detection of strings of SQL query has been proposed by [1]. The model estimated the genuine and manipulated taints of the SQL string. Filters are made by the defensive programmer [2, 3] to prevent malicious input queries from the system. The API security has inbuilt features to monitor malware script through SQL injection. Patterns in the vulnerability have been identified by [4, 5] in which static analysis framework has been applied to locate SQL injection strings. It manually decrypts each string and discovers its meaning. This process is slow and has not been recommended for an advanced security framework. This system is also not able to

identify the pattern of SQL injection attack. References [1, 6] had discovered SQL injection preventive tool called AMNESIA that is able to locate the malware string at run time. The usage of this tool is expensive and its accuracy depends on the data-size. One another SQL injection detection framework named SAFELI has been introduced by [7]. This framework is able to locate all the fundamental SIA vulnerabilities before the run time. This framework is used to gather the information from the source code. This framework is used to locate very delicate vulnerability information from the SQL script that is not even caught by any other scanner. Its effect can be seen before sending the data for the execution. Various proxy filters [8] have been proven very effective against SQL injection attack. The filter uses proxy web application or server where it locates the malware script before the actual web application. These filters has certain rules and protocols based on which it applies searching into the SQL stream. Alenezi et al. [9] was coined by [10] that construct analogy of SQL statement with parse tree before the execution. The introduced framework is time consuming as it performs localization before the run time. Another methodology has been introduced by [11] who gave a procedure to detect SQL injection attacks using syntactical SQL query structure that are generated by web application. A framework named as 'Instruction set randomization' granted developers to generate SQL queries using randomized keywords. A tool named 'Swaddler,' proposed by Boyd et al. [12], perform the analysis of various states of web application and analyze the relationship states and the execution. Aliero et al. [13] introduced an approach in which strings are converted into tokens. These tokens are then classified into original query and the malware query. If both the generated tokens are found to be same then it is considered that the SQL injection attack has not been performed. Some machine learning techniques are also proven relevant to identify patterns in trained SQL strings. Kernel of tree data-structure has been proposed by [13] that perform analysis of SQL stream to generalize vectorization of feature space for input data. In a methodology proposed by [14], it has been seen that the SVM classifier has been trained feature vectorization in order to draw out the patterns to preserved the accuracy. Halfond and Orso [15] introduced a method based on token for which graphs are drawn by using nodes of the trained SVM. This approach uses encoding of training data of patterns in SQL injection stream. A methodology based on the combination of parse tree and the proxy has been introduced by [16] in which alignment of sequence of SQL syntax has been localized to identify the SQL injection attacked streams. Gould et al. [17] introduces the relationship among integrity, authenticity and the confidentiality that is affected by SQL injection attack. In today's era, the SQL injection attacks become revolutionized such that its detection becomes more challenging by the existing system. The advanced form of script becomes more prone to damage system security and hard to locate by filters.

By observing the concern and research gaps associated with SQL injection attacks, the proposed model put an effort to bring machine learning technology for the detection of SQL injection attacks. The machine learning and artificial intelligence are some buzz words that are mostly applied in advanced security system. The proposed model uses support vector machine (SVM) for the classification of benign and malicious query. The proposed model uses standard dataset [18]. First in order to get

trained with all the SQL malicious streams. The dataset contain all types of SQL malware streams. The model is then tested under unknown query for which the model classifies the benign and attacked query. The proposed model also uses Candid method for the prevention of SQL injection attacks [19]. This method is also called dynamic candidate evaluation method in which each stream of SQL query has been transformed into parse tree and each node of the tree has been analyzed to locate the malicious query [20]. The prevention system is dynamic and allows the prevention at run time. This technique replaces the malicious nodes with the dummy variable so that it may look un-recognized to the system. One of the models is proposed which is based on the concept of deep learning. It can find irregularities in the data collected on the Fog network. The proposed model based on performance indicators such as detection rate and accuracy; prove the capabilities and scalability [21]. In order to optimize the impact on social media, an interest-based algorithm with parallel social behavior was developed. In the beginning, a novel parallel architecture was built to exclude the least influential nodes and choose possible candidate nodes. The proposed technique is very fast, uses less memory, is time efficient, and eliminates the trade-offs that occur in existing algorithms [22]. The given method solves the problems of delayed training and over-fitting. In addition, it also improves the efficiency and accuracy of the classification model. Least squares minimization is used to solve basic regression problems, thereby providing a more effective output weight vector for layer-to-layer comparison. In the improved version of ELM, input weights and biases are randomly selected [23]. The given research uses a hybrid method to generate domain ontology in the field of citizenship issuance. The model is inspired by the lack of domain ontology in the citizen domain. The citizen domain ontology will be a useful supplement to the ontology library and can be used to achieve interoperability between different systems by using a common representation of data [24].

3 Proposed Methodology

Figure 1 showing the flow chart of the proposed methodology. The flow chart clearly depicts that the proposed scheme applied SVM on the pre-processed SQL input query and classify it into type of SQL threats are shown in Table 2.

As shown in the flow chart, the proposed scheme first takes the input SQL string from web client. Then, the model performs its type checking where the type of the variables in the query has been checked. Then the query has been normalized in which unwanted string, spaces, comments, etc., are removed. The aim of the normalization is to extract the require SQL command from the stream of string. Then, the model eliminates the unwanted parameter from the extracted query. The model then applied SVM for the detection and the classification of the SQL query type. After the classification, the proposed model applies candid method for the prevention of the attacks. The candid method replaces the attacked strings with the other variable that becomes unrecognizable to the system. Figure 2 showing the parse

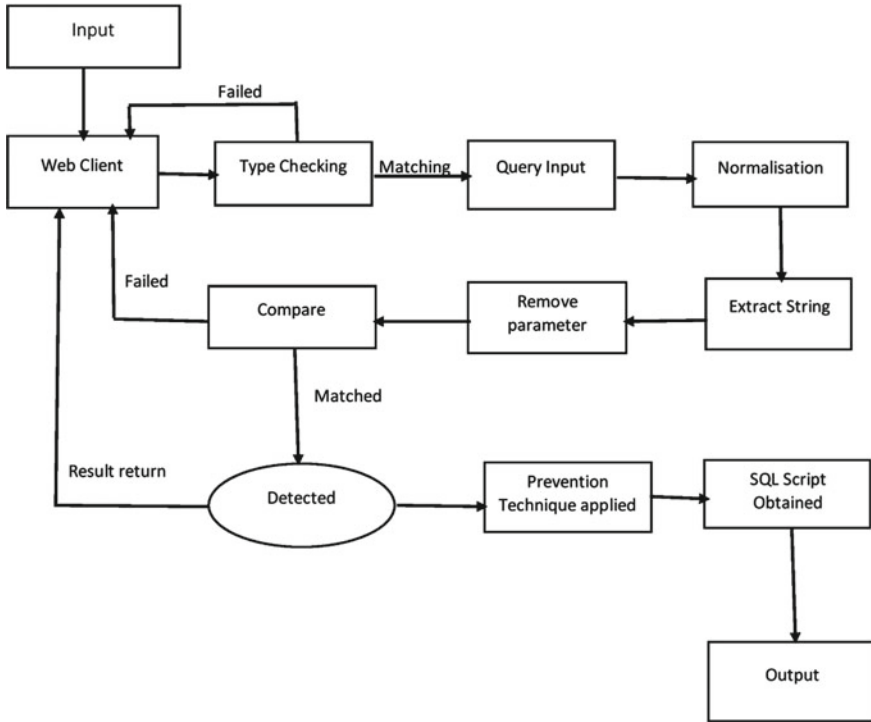


Fig. 1 Flow chart of the proposed model

tree of sample SQL attack. The candid method applied that screens out each string then replaces the malicious query.

4 Experimental Results

SVM efficiently perform detection of SQL injection attacks. Figure 2 shows the graph between true positive rate (TPR) and the data-size. The graph shows the change in the recognition of the number of SQL injected strings with respect to data-size. As we increase the data-size, the variation can be seen in TPR for the SQL strings are captured by the model. The average true positive rate for the recognition of SQL malware string is found to be more than 97%.

Figure 3 shows the graph between true negative rate (TNR) and the data-size. The graph shows the change in the rejection rate of benign queries with respect to data-size. As we increase the data-size, the variation can be seen in TNR. TNR shows that the model does not accept the benign query as malicious query. The average true negative rate for the rejection in the recognition of benign query is found to lie in the range of 94–95%.

Table 2 Different types of SQL injection attacks

Attack types	Description
Vulnerability based on tautologies	SQL queries contain some logical conditions that remain universally true
Logically incorrect SQL queries	Injecting SQL malware query using the error message thrown by system repeatedly
Union SQL query	SQL injected malware query has been joined with other genuine SQL query using the UNION operator to find out the data from the database table
Stored procedure	Many databases have built-in stored procedures. The attacker executes these built-in functions using malicious SQL injection codes
Piggy-backed SQL queries	SQL malware has been inserted with the benign SQL query to obtain illegal database access
Blind SQL query	Blind SQL injection is a type of SQLI attack that asks the database true or false questions and determines the answer based on the application's response
SQL injection based on alternate encodings	SQL injection attack query has been encoded and then inserted into the fields of web application to obtain unauthorized database access

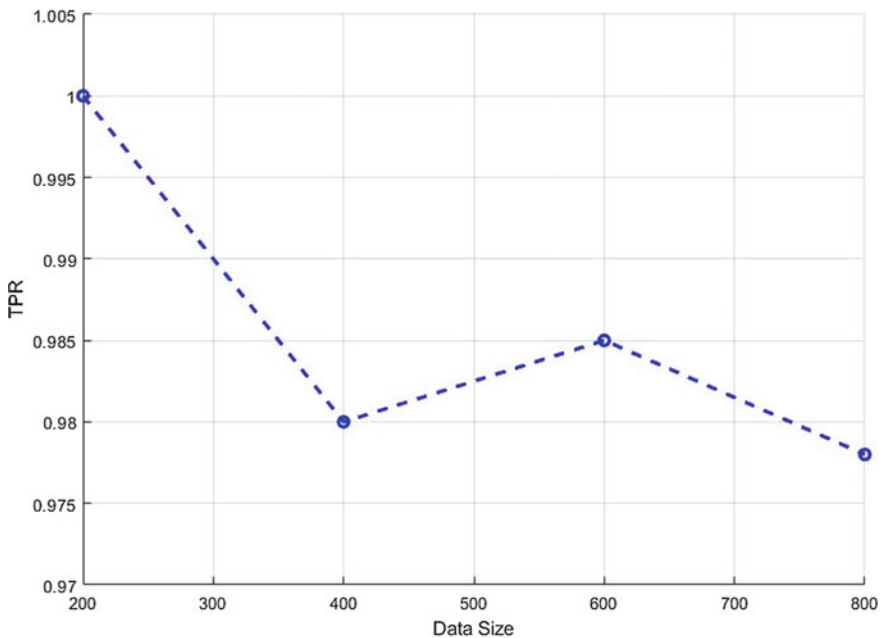


Fig. 2 Graph between TPR and the data-size

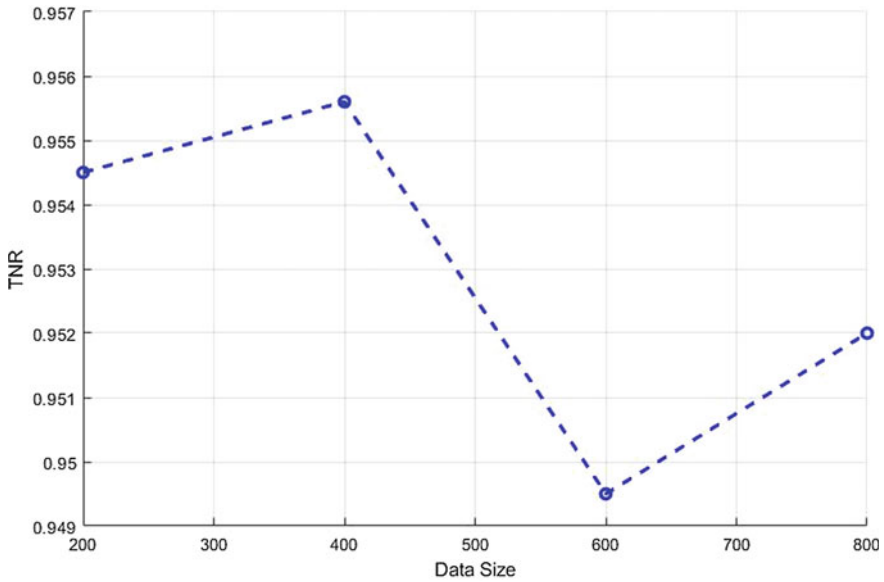


Fig. 3 Graph between TNR and data-size

Figure 4 shows the graph between false positive rate (FPR) and the data-size. The graph shows the change in the acceptance of benign query as malicious query with respect to the data-size. As we increase the data-size, the variation can be seen in FPR. The rate shows the undesired error in the model. The average false positive rate for the acceptance of benign query as malicious query is found to lie in the range of 1.2–1.6%.

Figure 5 shows the graph between false negative rate (FNR) and the data-size. The graph shows the change in the rejection of recognition of malicious SQL query as malicious with respect to the data-size. As we increase the data-size, the variation can be seen in FNR. The rate shows that the model is unable to recognize the malicious SQL query as malicious. This rate shows the error present in the model to recognize the malicious query. The average false negative rate for the recognition of malicious query as malicious is found to lie in the range of 0–8%.

Figure 6 shows the detection time utilized by the SVM with respect to the accuracy of the model. The average accuracy of the detection of SQL injection attack is found in the range of 95–96%. The detection time also lies in the range of 0.006–0.024 s.

Figure 7 shows the graph between detection time and the training time of the proposed model. As seen in the graph, the proposed model shows the detection time in the range of 0.006–0.024 s. Graph also shows that the model takes training time in the range of 0–3.5 s.

From Table 3, it is clear that the proposed model is efficient as compared to some recent models. The proposed system can be utilized for real-time in cyber forensics, filtering, etc., applications.

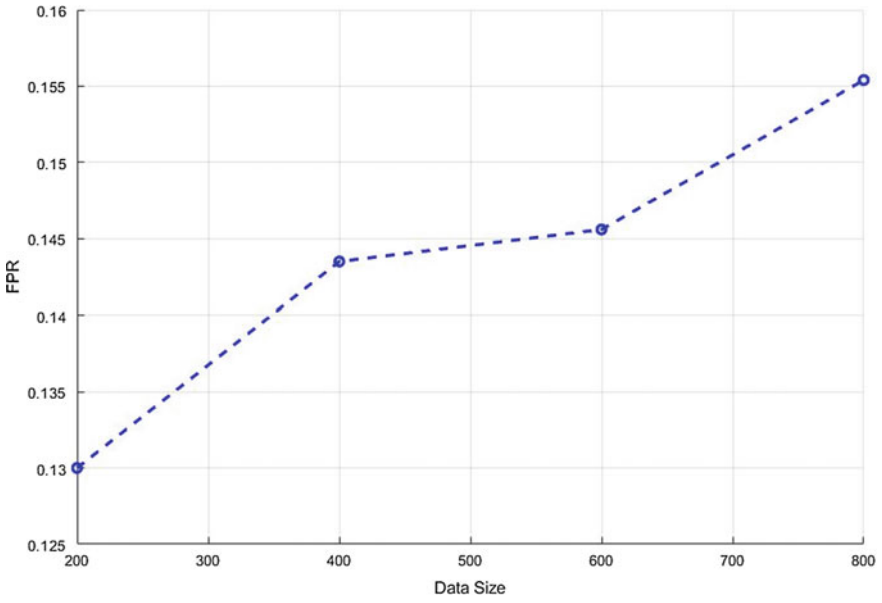


Fig. 4 Graph between FPR and data-size

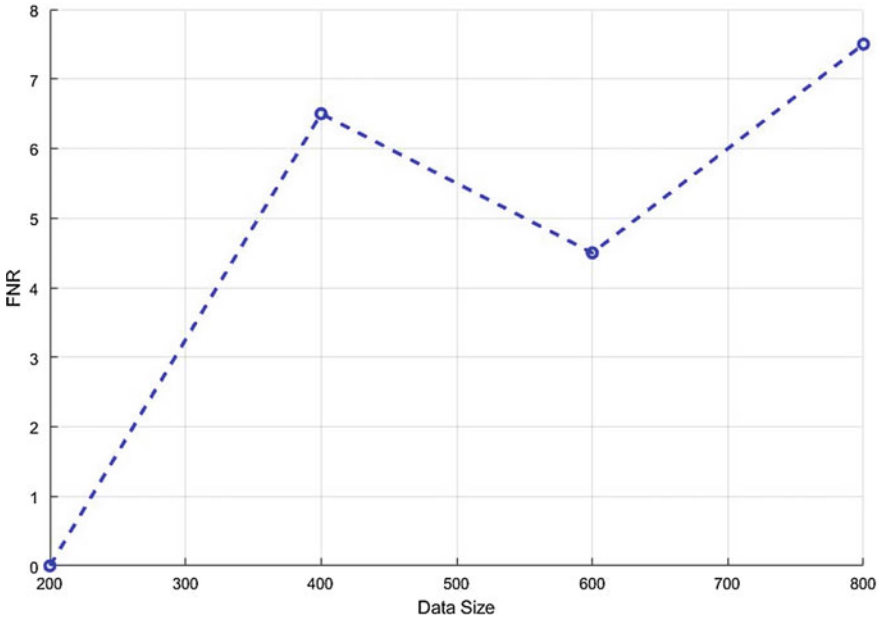


Fig. 5 Graph between FNR and data-size

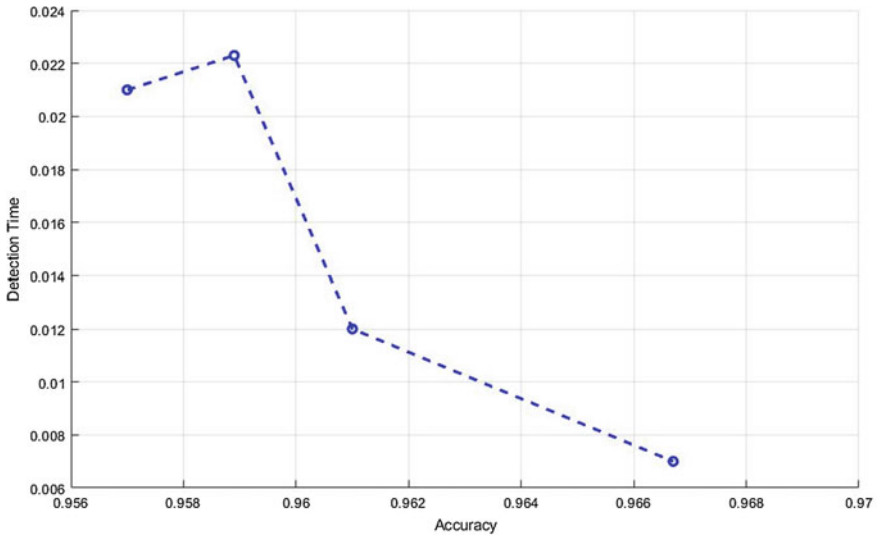


Fig. 6 Graph between detection time and the model accuracy

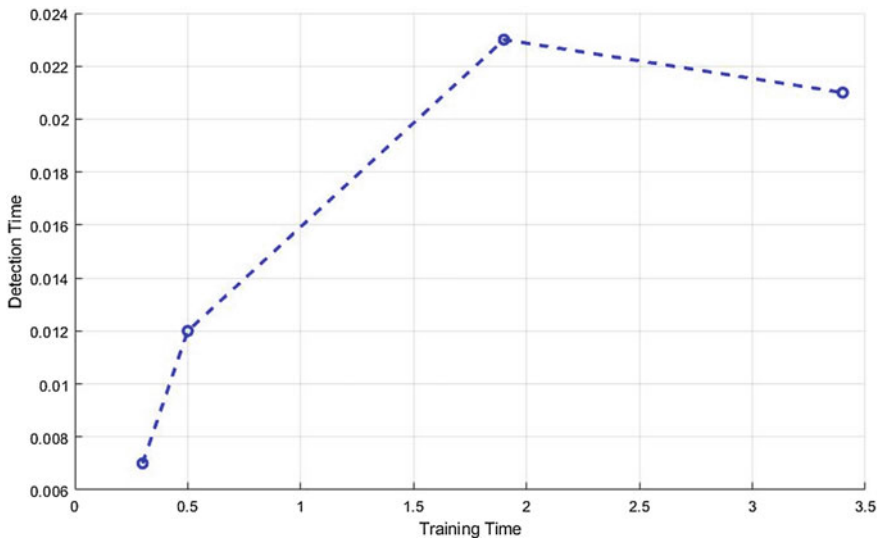


Fig. 7 Graph between detection time and the training time

5 Conclusion

This paper investigated the efficient and robust algorithms for the detection of SQL injection attack input SQL stream. The model applies SVM model to perform

Table 3 Comparison table presenting comparison with other recent publications

Techniques	Classifier	Average accuracy (%)	Dataset
Ladole and Phalke [25]	SVM	94.25	SQL queries
Joshi and Geetha [20]	Back proportion neural network	94.6	Over 13,000 URL address
Hasan et al. [26]	ANN	93.45	Over 25,000 URL address
YawAsabere and Kwawu Torgby [27]	SVM and decision tree	92.87	Used 1800 malicious query
Appelt [28]	Decision tree	93.65	NSL-KDD
Proposed methodology	SVN (detection and prevention)	95.25	Standard dataset

the detection of SQL malicious strings. The proposed model also utilized candid approach for the prevention from SQL malicious strings. The candid method simply generates the parse tree of the input string and performs the analysis of each node of the tree. The model tries to replace the malicious stream of SQL query with some dummy variable that are unable to recognize as SQL command and hence the system will be prevented from the script. The average detection accuracy of the malicious SQL script is found to be 95.25%.

References

1. R. Muhammad, R. Muhammad, R. Bashir, S. Habib, Detection and prevention of SQL injection attack by dynamic analyzer and testing model. *Int. J. Adv. Comput. Sci. Appl.* **8**(8), 209–214 (2017). <https://doi.org/10.14569/ijacsa.2017.080827>
2. A. Ciampa, C.A. Visaggio, M. Di Penta, A heuristic-based approach for detecting SQL-injection vulnerabilities in web applications, in *Proceedings of International Conference on Software Engineering* (2010), pp. 43–49. <https://doi.org/10.1145/1809100.1809107>
3. P. Bisht, P. Madhusudan, V.N. Venkatakrishnan, CANDID: dynamic candidate evaluations for automatic prevention of SQL injection attacks. *ACM Trans. Inf. Syst. Secur.* **13**(2), 1–38 (2010). <https://doi.org/10.1145/1698750.1698754>
4. R.A. McClure, I.H. Krüger, SQL DOM: compile time checking of dynamic SQL statements, in *Proceedings of 27th International Conference on Software Engineering ICSE05* (2005), pp. 88–96. <https://doi.org/10.1109/icse.2005.1553551>
5. G. Buehrer, B.W. Weide, P.A.G. Sivilotti, Using parse tree validation to prevent SQL injection attacks, in *SEM 2005—Proceedings of 5th International Workshop on Software Engineering and Middleware*, Sept 2005, pp. 106–113. <https://doi.org/10.1145/1108473.1108496>
6. W.G.J. Halfond, A. Orso, Preventing SQL code injection by combining static and runtime analysis. *Distribution* (2008)
7. B.A. Pham, V.H. Subburaj, An experimental setup for detecting SQLi attacks using machine learning algorithms. *J. Colloq. Inf. Syst. Secur. Educ.* **8**(1), 1–13 (2020). [Online]. Available: <https://cisse.info/journal/index.php/cisse/article/view/124>
8. P.S. Naidu, R. Kharat, *Security in Computing and Communications*, vol. 625 (2016)

9. M. Alenezi, M. Nadeem, R. Asif, SQL injection attacks countermeasures assessments. *Indones. J. Electr. Eng. Comput. Sci.* **21**(2), 1121–1131 (2020). <https://doi.org/10.11591/ijeecs.v21.i2.pp1121-1131>
10. Y. Kosuga, A study on dynamic detection of web application vulnerabilities, Aug 2011, p. 113
11. S.O. Uwagbole, W.J. Buchanan, L. Fan, An applied pattern-driven corpus to predictive analytics in mitigating SQL injection attack, in *Proceedings of 2017 7th International Conference on Emerging Security Technologies EST 2017*, Sept 2017, pp. 12–17. <https://doi.org/10.1109/EST.2017.8090392>
12. S.W. Boyd, A.D. Keromytis, SQLrand: preventing SQL injection attacks, in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3089 (2004), pp. 292–302. https://doi.org/10.1007/978-3-540-24852-1_21
13. M.S. Aliero, A.A. Ardo, I. Ghani, M. Atiku, Classification of SQL injection detection and prevention measure. *IOSR J. Eng.* **06**(02), 6–17 (2016). [Online]. Available: www.iosrjen.org
14. D. Appelt, N. Alshahwan, L. Briand, Assessing the impact of firewalls and database proxies on SQL injection testing, in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8432, Nov 2013 (2014), pp. 32–47. https://doi.org/10.1007/978-3-319-07785-7_2
15. W.G.J. Halfond, A. Orso, AMNESIA: analysis and monitoring for neutralizing SQL-injection attacks, in *20th IEEE/ACM International Conference on Automated Software Engineering ASE 2005* (2005), pp. 174–183. <https://doi.org/10.1145/1101908.1101935>
16. Z. Su, G. Wassermann, The essence of command injection attacks in web applications, in *Conference Record of the Annual ACM Symposium on Principles of Programming Languages* (2006), pp. 372–382. <https://doi.org/10.1145/1111037.1111070>
17. C. Gould, Z. Su, P. Devanbu, JDBC checker: a static analysis tool for SQL/JDBC applications, in *Proceedings of International Conference on Software Engineering*, vol. 26 (2004), pp. 697–698. <https://doi.org/10.1109/icse.2004.1317494>
18. S. Panda, S. Ramani, Protection of web application against SQL injection attacks. *Int. J. Mod. Eng. Res.* **3**(1), 166–168 (2013)
19. N. Shah, Securing Database Users from the Threat of SQL Injection Attacks (2017). [Online]. Available: <http://digitalrepository.smu.edu>
20. A. Joshi, V. Geetha, SQL injection detection using machine learning, in *2014 International Conference on Control, Instrumentation, Communication and Computational Technologies ICCICCT 2014*, no. 2 (2014), pp. 1111–1115. <https://doi.org/10.1109/ICCICCT.2014.6993127>
21. D.S. Shakya, D.S. Smys, Anomalies detection in fog computing architectures using deep learning. *J. Trends Comput. Sci. Smart Technol.* **2**(1), 46–55 (2020). <https://doi.org/10.36548/jtcsst.2020.1.005>
22. D. Sivaganesan, Novel influence maximization algorithm for social network behavior management. *J. ISMAC* **3**(1), 60–68 (2021). <https://doi.org/10.36548/jismac.2021.1.006>
23. S.R. Mugunthan, T. Vijayakumar, Design of improved version of sigmoidal function with biases for classification task in ELM domain. *J. Soft Comput. Paradig.* **3**(2), 70–82 (2021). <https://doi.org/10.36548/jscp.2021.2.002>
24. R. Bastola, S. Shakya, Developing domain ontology for issuing certificate of citizenship of Nepal. *J. Inf. Technol. Digit. World* **2**(2), 73–90 (2020). <https://doi.org/10.36548/jitdw.2020.2.001>
25. A. Ladole, D. Phalke, SQL injection attack and user behavior detection by using query tree, fisher score and SVM classification. *Int. Res. J. Eng. Technol.* **03**(06), 1505–1509 (2016)
26. M. Hasan, Z. Balbahaith, M. Tarique, Detection of SQL injection attacks: a machine learning approach, in *2019 International Conference on Electrical and Computing Technologies and Applications ICECTA 2019* (2019). <https://doi.org/10.1109/ICECTA48151.2019.8959617>
27. N. YawAsabere, W. Kwawu Torgby, Structured query language injection (SQLI) attacks: detection and prevention techniques in web application technologies. *Int. J. Comput. Appl.* **71**(11), 29–39 (2013). <https://doi.org/10.5120/12404-8908>

28. D. Appelt, Automated security testing of web-based systems against SQL injection attacks (SOFIA), June 2016, p. 140

String Matching Algorithm Based Filter for Preventing SQL Injection and XSS Attacks



Abhishek Kumar Yadav and Arun Kumar

Abstract Injection attacks are most often experienced computer security breaches. Among them, SQL injections with Boolean type of queries and cross-site scripts are mostly done. Probabilistic models fail to adapt to most of the new kind of attacks, due to the changing nature of these attacks. This paper proposes a novel technique for filtering SQL Boolean queries, and these queries are capable of bypassing the existing models but were trapped in this new model. The model uses Rabin-Karp algorithm which is based on the concept of string matching. An SQL query passed as user input is evaluated by the proposed query filter, which is designed to separate malicious queries from the normal queries and flag them as malicious. The proposed model of filtering was experimented on a JavaScript-based web application, designed to test the model with a number of Boolean queries. The results were promising with a maximum accuracy of 96%. Moreover, the model has proved to be effective against the Boolean SQL queries which shows that the SQL injection attacks could be prevented using this model.

Keywords SQL injection · Cross-site scripting · Query filter · Rabin-Karp algorithm · String matching algorithm · Rolling hash function · Data-driven web applications · Attacks

1 Introduction

The year 2020 marked by the deadly pandemic also marked a surge in cyber-crimes, and statistics explains the methods used were 7% SQL injections and 25% as cross-sites scripts [1]. There is a constant need for a new method to defend the web applications against such attacks. Our model proves to be a better one at giving results

A. K. Yadav (✉) · A. Kumar

Department of Computer Science and Engineering, Centre for Advanced Studies, Dr. A.P.J. Abdul Kalam Technical University, Lucknow, Uttar Pradesh 226031, India
e-mail: abhishek06041996@gmail.com

A. Kumar

e-mail: drarun@cas.res.in

and might shape the future of web security in the near future. We have used a mechanism that filters out all the Boolean SQL injection-based malicious requests. These requests could be as damaging as any database crash. This mechanism uses a module called query filter to separate all the SQL requests which are of malicious in nature and consist some specific pattern of Boolean SQL queries.

The World Wide Web or the WWW has emerged as a multi-service network involving a wide variety of components and technologies including the client-side technologies and the server-side technologies. These technologies are full proof in terms of the guarantee they provide with the services. But, there exist flaws in the system as well as new discrepancies which create a void. These voids pose the biggest challenges in terms of security of the web-based systems. There are many vulnerabilities in the client-side technologies as well as the server-side technologies. There are elite group of hackers who are expert in exploiting these vulnerabilities. Client-side attacks discussed here are SQL injection attacks and the cross-site scripting (XSS).

The code injection allows attackers to execute malicious scripts on the web browsers of victims, whereas the SQL injections inject malicious query in the database of the web applications. The SQLi is termed as plague for databases, in the modern web-based applications. In worst cases, data that can be compromised via SQLi or XSS include stealing of passwords, cookies, debit-credit card details [2]. Also, securing Web sites of domains like banking, health care, financial services, defense are tough challenges. This paper presents a novel technique to prevent the SQL injection attacks and the cross-site scripting attacks using the Rabin-Karp string matching algorithm-based query filter.

The primary research goal for us was to solve the SQL vulnerabilities in the present web-based systems by using a query filter model. The performance of such a unique method is verified through a series of experiments performed on the web application, and the results are recorded on the browser with the help of a TTFB graph. The effectiveness of the query filter in preventing the vulnerabilities is proved by this graph which records the request to response time. After a particular query is detected, there is an output return value, which is summarized into a thorough analysis report and then it is stored in the database to keep all the records. Later on, it is used for verification. For validating the results, here, JavaScript and Apache Tomcat server are used in a web application for implementing the query filter-based detection system. The research presents a method that would in the future be the pioneer of more simple designs for detector tools in the computer, database, and network security. Web servers are one of the important components of the web technology, and all the flow of data is regulated via the servers. Our query filter is embedded or bridged with the web server, and the malicious Boolean SQL requests can be filtered and separated from the genuine queries, and the web security can be ensured. Many researches solve the problems of web security vulnerabilities with the help of the machine learning algorithms, but these models are probabilistic. The design of models like the one presented in this research could be an important factor for the future of web security.

2 Background

2.1 SQL Injection Attacks

Attackers try to manipulate the queries that an application makes to its database; this is called an SQL injection attack. This manipulation is a key for the attackers to access data which is generally not available. These data are the sensitive and private information of users or any data which the application would need to access [3]. Attackers use such a technique to gain the data and try to modify it permanently or temporarily as per the need. In crucial cases, they get the full control in their hands by modifying the original form of data, behavior, and contents [4]. In many case studies, there were incidents recorded in which attackers escalated SQL injection attacks to get through the server and compromise the back-end architecture, which leads to a DOS attack [5]. Jemal classifies the injection string of SQL according to their source of injection. According to his research, the malicious string is injected, through the user input forms, through browser cookies, through intermediary server variables, via stored injections [6].

The blind or Boolean SQL injection is the type of SQL injection studied in this research. It tries to manipulate the system based upon the application's response.

2.2 Cross-Site Scripting Attacks

Cross-site scripting attacks are basically malicious codes that are executed on the client side of the web applications like the web browser. XSS attacks are of persistent, non-persistent, and DOM-based XSS attacks. A persistent XSS attack is also known as stored attack. It can be found in those web applications where users are permitted to enter a HTML or JavaScript code, e.g., the web browser's search box, forms to be filled. A non-persistent XSS attack is also known as reflected XSS attack. Here, no code or script needs to be stored in the web application. DOM-based XSS attack is an alternative of both the reflected and the stored XSS attacks. Complete analysis of domains was presented by Gupta and Gupta in a 2017 paper mentioning the defense mechanism of cross-site scripting [7]. They prepared a score card of the domains on the parameters such as always vulnerable, frequently vulnerable, regularly vulnerable, occasionally vulnerable, rarely vulnerable.

2.3 Rabin-Karp Algorithm

It has been quite a time since the computer science industry is using the Rabin-Karp algorithm for many purposes. The algorithm has been used for comparing patterns based on the hash values. There were researches, in which image was recognized

using the algorithms. Using the k-gram component and the relationship of adjacent pixels in each image, patterns were identified [8]. The algorithm is identified as one of the proficient methods to identify the similarity index in the documents. It seeks for the substring patterns in texts with the help of hashing. The algorithm has been consistently used in the various plagiarism check software. It was invented by Rabin and Karp and named after them too. The hash values of texts are compared in order to match the level of similarity in the document. It works very well in case of multiple pattern search. Concept is simple which means if two strings are equal, then their hash values must also be equal. If the hash values are not equal, then the algorithm must seek for other characters and their associated hash values.

3 Related Work

Acunetix 2020 web application vulnerability report identifies a total of 26% of the Web sites are severely vulnerable, while 63% of Web sites have medium-type vulnerabilities present in them. It reports that 25% of the total web applications on the Internet are vulnerable to XSS. Whereas, nearly 8% of them are vulnerable to SQL injection attacks. It has named the SQL injection and XSS attacks among the high-severity vulnerabilities and also defines high severity as a level where an attacker is capable of fully compromising the confidentiality, integrity, and availability of a

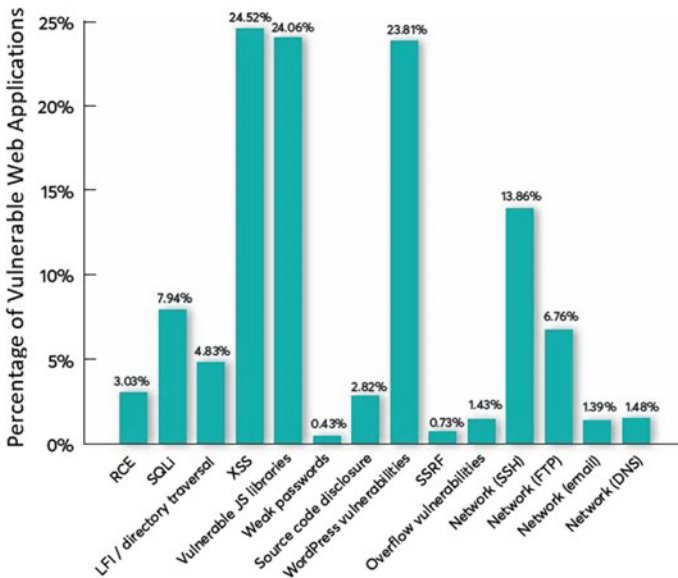


Fig. 1 Percentages of type vulnerabilities in sites

system without using any special access grant mechanism. Figure 1 shows the graph with reference to above report from acunetix [9].

Abikoye in [10] used a method similar to that of string matching, but the algorithm used was Knuth-Morris-Pratt string matching algorithm. This research brought a technique for preventing SQLi and XSS attacks using string matching. They used to match the input string with stored patterns of injection strings to detect malicious code.

Tariq et al. [11] used a method of genetic algorithm in combination with reinforcement learning. The research used real datasets. Claims were made that the method had better performance when compared to other existing methods. Also, the method was flexible to the increment in XSS payloads. Which means, it can survive even if the scale of attack becomes large and the amount of injection is increased rapidly without any knowledge of the same.

Aliero et al. [12] presents his work as focused on minimizing the false positives and false negatives in a machine learning-based method to detect and prevent SQLi. Use of object-oriented approach was emphasized, and the method was called black box testing-based method. Web applications were developed to test the accuracy and validate results.

Jemal in [6] showed the overview of all the machine learning-based methods to prevent SQLi attacks. It describes methods to classify SQLis also the mitigation methods based on concept of ontology and machine learning. Interestingly, it highlighted the performance of all possible solutions based on accuracy, recognition rate, and precision. The methods were all good though, but the false positives were major concern among all of them. It also states that statistics-based methods like machine learning-based need a clearly refined dataset also that statistics show trends only, and 100% accuracy is still awaited.

Gu et al. [13] used a traffic-based framework for SQLi detection, named DIAVA. It claims to accurately identify malicious SQLi among the suspected ones. It used a GPU-based dictionary attack along with its DIAVA framework. The research earns a badge of state-of-the-art method and better than other firewalls. It is capable of managing leaked data as soon as it is out in public domain. Again, this method does not use machine learning algorithms and techniques, yet it outperforms all the trending methods. Network traffic-based SQLi monitoring was the other aspect of this research which gives a boost of morale to other researches out and ready to deviate from common trends right now.

Authors of [14] presented a technique of classifying the SQL queries on the features of all the initial query string. A gap weighted string and a subsequence function are used in order to classify all the similar types of queries into groups. This function was the core of the authors' research as this function computes the similarity of queries that are unknown to preselect query string used for training. It also uses an SVM classifier for similarity measurements to identify a normal and a malicious query. But since, SVM is a probabilistic classifier, so even if queries are not malicious but have a slight similarity with the trained query set, so unnecessary flaw is identified here.

IEEE Transactions on Vehicular Technology volume 68 published an article by Qi Li which brought a totally new aspect of need of detecting SQLis [15]. He devised a machine learning-based method for detection of SQL injection attacks in intelligent transport system. However, this paper also clearly mentions that collection and selection of data for training are a huge problem in artificial intelligence-based detection methods. So, the training part becomes tough, in order to get guaranteed accurate results; the topmost refined SQLi datasets must be used which again becomes a hectic task. Hundred percentage protection of SQLi is not possible even with the best machine learning algorithms.

Avancini in [16] used a model of combining genetic algorithm with symbolic execution for automatic generation of security test cases. These test cases were those which exposed a vulnerability by making an application control flow as a satisfying vulnerability condition. This method earned huge attention.

In another article by Batista et al. [17] published in 2018, fuzzy neural network was used to create an expert and full proof system of detection of SQLi. Fuzzy neural networks are one of the most advanced and complex neural networks usually used where high computing power is needed. This system proposes hybrid models, which via fuzzy rules allow formation of full proof systems in cybernetic data attacks, specially focused on the SQL injection attack. This model claims to be 94% close to results as of fuzzy sets. Since, it uses cup dataset from the version 1999. So, the results cannot be thought of being very accurate as it just reaches 94% near to older fuzzy neural network based system. The overall selection mechanism was good, but the dataset used was old, and hence, it is understood that neural networks are good choice for fuzzy logic-based SQL injection classification.

4 Methodology and Working

We have introduced a “**query filter**” in this model. The problems discussed in case of SQL injections are solved using the query filter, and the results are better. The query filter accepts requests which are made to the server, then it is checked with all the possible malicious queries and their combination through the Rabin-Karp string matching algorithm. The scale of the project is adjusted according to the Kaggle SQL injection dataset.

Figure 2 shows the overview of the system enhanced with the proposed filter. The overview demonstrates a normal web system bridged with the proposed query filter module. If a Boolean combination is found to exist in the requested query, then the filter understands that these requests are malicious and action should be taken. Binding the server with the query filter actually serves the purpose of:

- **Filtering the requests:** Separate queries which are malicious combination of patterns and in the future such requests would be blocked right away.

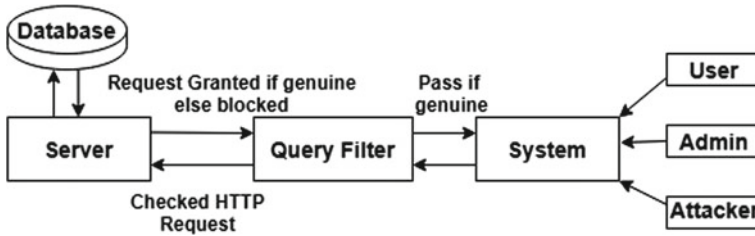


Fig. 2 Overview of proposed system with query filter

- Keeping and maintaining record of all malicious patterns, after the requested query is verified to be malicious or manipulated from the pre-existing repository of queries and their combination.
- Verifying queries which are genuine so that malicious queries are recorded, and all possible combination of the queries and pattern of queries is kept for verifying; if it is a genuine request or an attacker’s request.

4.1 Hash Function()

The hash value of a pattern is defined by the hash function chosen for the algorithm. The hash function called the rolling hash function is used in this method. The advantages of this hash function are:

- It gives a large output which makes it difficult for two different strings to have same hash value, i.e., less chance of spurious hits.
- It is fast and easy to implement.
- Calculation is based on the ASCII value and base 10 exponent of the character’s place value.

4.2 Hash Value Evaluation

To compare to this hash value, the algorithm searches for the same hash-valued SQL injection string window of size 7. This is the current window of the string for comparison with the pattern. The Rabin-Karp algorithm saves time by comparing only the hash values. $H(P)$ = hash value of pattern and $H(W)$ = hash value of window. If for now the slide window is on $W = PQ'OR'1$, the hash value of current window is calculated as: $H(W) = 80 * 10^6 + 81 * 10^5 + 39 * 10^4 + 79 * 10^3 + 82 * 10^2 + 39 * 10^1 + 49 * 10^0$. Table 1 shows the ASCII values of corresponding characters.

Table 1 ASCII values of characters in a window

Character	ASCII value
P	80
Q	81
,	39
O	79
R	82
,	39
1	49

4.3 Working of the Query Filter

In an event of SQL injection attack, the attacks are performed using an SQL injection query like **PQ'OR'1' = '1'#**. In this research, for the purpose of storing SQL injection string, a repository is created, which stores all the SQL injection queries. There is a whole bunch of queries, and a wide variety of them are from Kaggle’s popular SQL injection dataset [18]. There is a separate repository for XSS injection strings. This repository is the base of test and experimentation in this research for Boolean SQL queries. These are tried to be injected as user inputs. The pattern which is visible here is **1 = 1** which means it is a “Boolean-based” SQL injection string. It always gives a true value for the user input. As the filter designed for this research is based on the Rabin-Karp algorithm, so the role of this algorithm in the filter is discussed in Fig. 3. The algorithm finds whether this pattern is present in the injected query or not.

Figure 3 shows the work flow diagram of the query filter as it receives the SQL injection string **PQ'OR'1' = '1'#** as user input. It starts the process of finding the pattern, length of the pattern **L** and calculates the hash value **H(P)** of the pattern. The hash value is calculated using the rolling hash function **H()**.

SQL injection string: **PQ'OR'1' = '1'#**; **pattern = '1' = '1'**.

Length of pattern = 6.

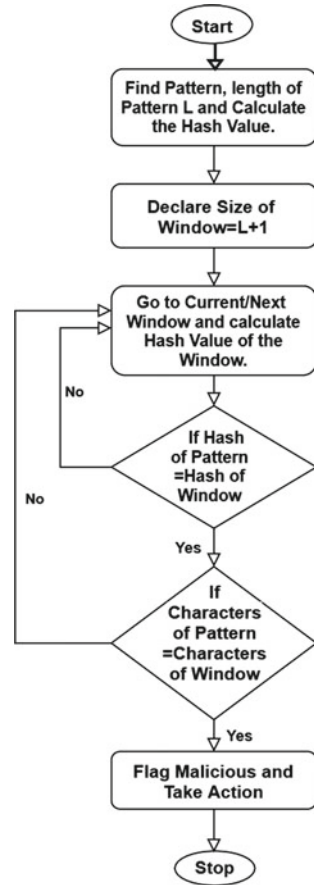
Hash value of pattern is **H(P) = 39 * 10⁶ + 49 * 10⁵ + 39 * 10⁴ + 61 * 10³ + 39 * 10² + 49 * 10¹ + 39 * 10⁰ = 44,355,429**.

Next, according to the algorithm, the size of the window is declared as length of pattern + 1. Hence, size of window = Length of pattern + 1 = 6 + 1 = 7.

The current window is on **W = PQ'OR'1** **H(W) = 80 * 10⁶ + 81 * 10⁵ + 39 * 10⁴ + 79 * 10³ + 82 * 10² + 39 * 10¹ + 49 * 10⁰ H(W) = 88,577,639. H(W) != H(P)**. Comparison of hash values is done because this makes the process faster without getting involved in comparing the character by character, and here, the two hash values are not equal, so slide to the next window.

The next window is **W = Q'OR'1** **H(W) = 81 * 10⁶ + 39 * 10⁵ + 79 * 10⁴ + 82 * 10³ + 39 * 10² + 49 * 10¹ + 39 * 10⁰ H(W) = 85,776,429. H(W) != H(P)**. Sliding to the next window, since the hash did not match here too, which means the characters would definitely not match.

Fig. 3 Working of the query filter



The next window is $W = 'OR'1' = H(W) = 39 * 10^6 + 79 * 10^5 + 82 * 10^4 + 39 * 10^3 + 49 * 10^2 + 39 * 10^1 + 61 * 10^0$ $H(W) = 47,764,351$. $H(W) \neq H(P)$, hash values are not equal, so slide to the next window, and repeat the steps of comparing the hash values of window and pattern.

The next window is $W = 'OR'1' = 'H(W) = 79 * 10^6 + 82 * 10^5 + 39 * 10^4 + 49 * 10^3 + 39 * 10^2 + 61 * 10^1 + 39 * 10^0$ $H(W) = 87,643,549$. $H(W) \neq H(P)$, so slide to the next window.

The next window is $W = R'1' = '1$ $H(W) = 82 * 10^6 + 39 * 10^5 + 49 * 10^4 + 39 * 10^3 + 61 * 10^2 + 39 * 10^1 + 49 * 10^0$ $H(W) = 86,435,539$. $H(W) \neq H(P)$. Since, the hash values of the window and pattern are not equal and so the loop of again going to next window continues.

Now, the next window is $W = '1' = '1' H(W) = 39 * 10^6 + 49 * 10^5 + 39 * 10^4 + 61 * 10^3 + 39 * 10^2 + 49 * 10^1 + 39 * 10^0$ $H(W) = 44,355,429$. Here, $H(W) = H(P)$, the hash value of string in the window equals the hash value of string in the pattern. So, now to confirm the availability of the malicious SQL Boolean query, the

Fig. 4 Matching the characters of pattern one by one

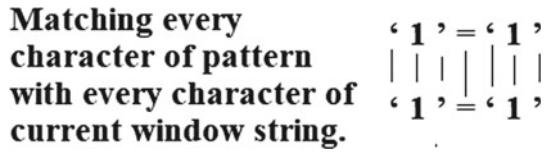


Table 2 SQL injection strings from Kaggle datasets

SQLi query	Boolean value
a' or 1 = 1; -	1
? or 1 = 1 -	1
anything' or 'x' = 'x'	1
PQ'OR'1' = '1'#	1
1' or '1' = 1	1
" or "a" = "a	1

filter matches every character of pattern with every character of current window one by one; Fig. 4 shows the matching.

All characters in the window match with the characters in pattern; Fig. 4 shows, hence, query PQ'OR'1' = '1'# is a malicious query and the filter flags it as malicious Boolean query and takes further actions of resetting http connection. The Rabin-Karp algorithm was used to verify the pattern in the user's input string with all the Boolean SQL injection queries that were combined and used. Table 2 shows ample list of SQL queries from Kaggle SQLi dataset.

Actions taken after matching patterns:

$$I = \sum_{x=0}^n F$$

```

Filter(I) {
info= dehash(convert ASCII to String)
if(info<>" ") {
X1= checkBoolSQLi(info);           //Check for Boolean SQLi
X2= checkXSSi(info);               //Check for XSS script
if( true(X1 || X2 ||) ) {           //if any condition is true
blockUser();markMalicious();       //Mark as Malicious
resetHTTP(); warningMessage(); }    //Take required action reset HTTP etc.
else { AccessGrant(); } }           //if both X1 X2 conditions fail, query is genuine
    
```


5 Result and Discussion

The query filter mechanism uses the rolling hash function. The detection tools were used to validate the results. The technique was tested on a Java-based web application, developed for this purpose with Apache Tomcat server, and MySQL database was used. The total number of queries taken for the experiments was nearly 250. These Boolean strings were taken from the Kaggle SQL injection dataset; Table 2 shows us the same and then added to the local repository created in the eclipse IDE. The results showed that the method performed great if the detection rate is concerned. An inbuilt simulation graph in Fig. 5 drawn from the web browser for demonstrating the output requests and the TTFB (time to first byte) measurement is used for the indication of the responses of the web server and the other resources from the network and server. The number of successful query detection is 240, when the experiments were run several times.

The time to first byte (TTFB) graph in Fig. 5 shows the waiting time gap between the user’s request and the first byte issued back as a response from the web server to the user’s web browser. In this case, TTFB waiting time was recorded as 7.12 ms. Since the queries were blocked by the query filter, hence, the graph shows zero requests sent back as responses, and this assures us of the working of the filter. It can be inferred from Fig. 5 that after requests were made and these were not immediately started, so the malicious input was being evaluated for about 3.06 s and then after confirming the malicious nature of the above SQL injection string. The http reset request was sent with a warning message. The resources that were requested were rescheduled after 4.74 ms of the http reset. The connection took almost 0.16 ms of time to be established again after confirmation of the maliciousness of the injection string. The number of responses sent back to the user is marked 0 because after the confirmation and reset, no request was granted.

Table 3 shows the comparison and analysis of this method with the previous methods. The accuracy of our model was 96%, which in case of encryption-based methods was 86–96% as they use probabilistic models. Efficiency was 95% which



Fig. 5 TTFB graph for time between request and response

Table 3 Comparative analysis with existing encryption-based methods (Boolean queries)

S. No.	Parameter of comparison	Proposed method	Existing methods
1.	Accuracy	96%	86–94%
2.	Efficiency	95%	95%
3.	Execution time	7.12–12 ms	11–20 ms
4.	Percentage improvement	10% lower limit 2% upper limit	
5.	CPU utilization	39%	>60%

is equal to that of encryption-based methods. Execution time of the model was between 7.12 and 12 ms which is better as compared to other models. Percentage improvements as compared with the previous models is 10% in lower limits and 2% in the upper limits. These limits refer to the minimum and maximum accuracy. The CPU utilization in our model is 39%; better than other models, it is due to the model being less complex and its efficient utilization of resources.

Figure 6 shows http request is reset leading to unavailability of requests and restricts unauthorized login in the Web site “etalenthunt” developed for testing the query filter. The http request was reset and the query filter stopped any further request grant.

Figure 7 shows how the connections were reset and the requests were halted. Apache version 9 offered these services via CATALINA, and these services can be stopped via the above query filter mechanism. The request was initiated in the same session, and the resources were made unavailable as it was a malicious query after the filter understood its malicious nature. Apache took the usual time to start and establish the connection with the port. The dedicated port was 8080, and the http request from the port 8080 was reset. This explains how the server was stopped from

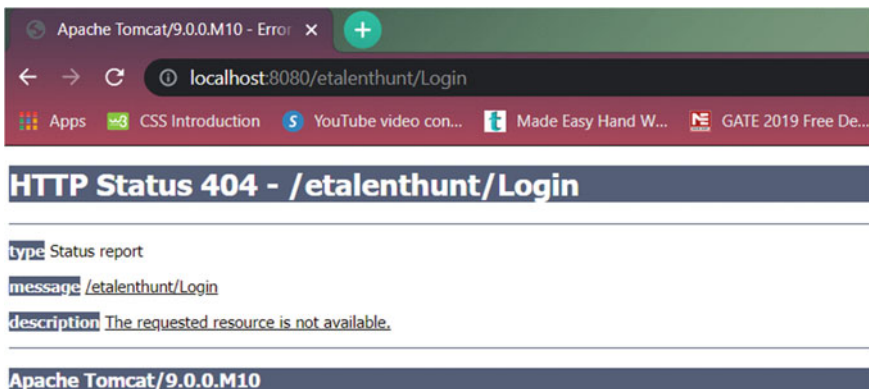


Fig. 6 Browser output after the filter was applied

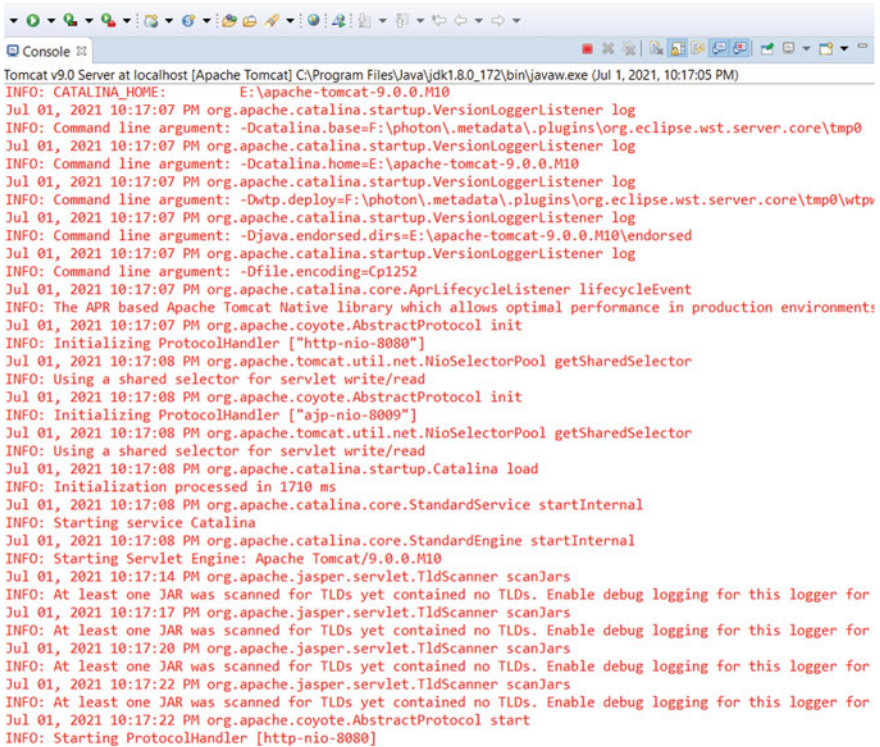


Fig. 7 Console result; Apache server did not grant any request from port 8080

granting the requests against the malicious queries. Apache took usual time but did not grant any request.

6 Conclusion and Future Scope

This model of query filters is effective in preventing against the Boolean SQL injection attacks. In this paper, we showed the effectiveness of the model and results which are better than the probabilistic models. Probably, this is the most user-friendly method of defending against SQL injection attacks and the cross-site scripts. Although cross-site scripts were not implemented in this paper, but the same methodology is to be followed against them too. The outputs and the functioning of the filter are a great as it showed results where only a certain level of advanced methodology has reached. The idea here was to bring a less complex method into light which could solve a bigger problem easily and without burdening the system. The problem of Boolean SQL injections was identified and then a repository of these strings was made. Next, the proposed query filter model was designed which filters

various types of Boolean-based queries and identifies the patterns present in them. Later on, when the maliciousness of the query was proved, the query filter marks these queries as suspicious and harmful. The full working of the system was ensured, and the results were recorded via graphs and performance of the system. Overall, this method showed better results than the existing models.

References

1. Acunetix Web Vulnerability Report, *The Invicti AppSec Indicator Spring 2021 Edition: Acunetix Web Vulnerability Report Introducing the Invicti AppSec Indicator* (2021). [Online]. Available: <https://www.acunetix.com/wp-content/uploads/2021/04/Invicti-AppSec-Indicator-Spring-2021-Edition-Acunetix-Web-Vulnerability-Report.pdf>
2. J. Li, Vulnerabilities mapping based on OWASP-SANS: a survey for static application security testing (SAST). *Ann. Emerg. Technol. Comput.* **4**(3), 1–8 (2020). <https://doi.org/10.33166/AETiC.2020.03.001>
3. D. Patel, N. Dhamdhare, P. Choudhary, M. Pawar, A system for prevention of SQLi attacks, in *Proceedings of the International Conference on Smart Electronics and Communication ICOSSEC 2020* (2020), pp. 750–753. <https://doi.org/10.1109/ICOSSEC49089.2020.9215361>
4. O. Ben Fredj, O. Cheikhrouhou, M. Krichen, H. Hamam, A. Derhab, An OWASP top ten driven survey on web application protection methods, in *LNCS*, vol. 12528 (Springer International Publishing, 2021)
5. F.Q. Kareem, S.Y. Ameen, A. Ahmed, A.A. Salih, SQL injection attacks prevention system technology: review (2021). <https://doi.org/10.9734/AJRCOS/2021/v10i330242>
6. A.M. Ines Jemal, O. Cheikhrouhou, H. Hamam, SQL injection attack detection and prevention techniques using machine learning. *Int. J. Appl. Eng. Res.* **15**(6), 569–580 (2020)
7. S. Gupta, B.B. Gupta, Cross-site scripting (XSS) attacks and defense mechanisms: classification and state-of-the-art. *Int. J. Syst. Assur. Eng. Manag.* **8**(s1), 512–530 (2017). <https://doi.org/10.1007/s13198-015-0376-0>
8. A.P.U. Siahahan, Rabin-Karp elaboration in comparing pattern based on hash data. *Int. J. Secur. Appl.* **12**(2), 59–66 (2018). <https://doi.org/10.14257/ijssia.2018.12.2.06>
9. Acunetix, *Web Application Vulnerability Report 2020* (2020). [Online]. Available: <https://www.acunetix.com/acunetix-web-application-vulnerability-report-2020>
10. O.C. Abikoye, A. Abubakar, A.H. Dokoro, O.N. Akande, A.A. Kayode, A novel technique to prevent SQL injection and cross-site scripting attacks using Knuth-Morris-Pratt string match algorithm. *Eurasip J. Inf. Secur.* **2020**(1), 1 (2020). <https://doi.org/10.1186/s13635-020-00113-y>
11. I. Tariq, M.A. Sindhu, R.A. Abbasi, A.S. Khattak, O. Maqbool, G.F. Siddiqui, Resolving cross-site scripting attacks through genetic algorithm and reinforcement learning. *Expert Syst. Appl.* **168**, 114386 (2021). <https://doi.org/10.1016/j.eswa.2020.114386>
12. M.S. Aliero, I. Ghani, K.N. Qureshi, M.F. Rohani, An algorithm for detecting SQL injection vulnerability using black-box testing. *J. Ambient Intell. Humaniz. Comput.* **11**(1), 249–266 (2020). <https://doi.org/10.1007/s12652-019-01235-z>
13. H. Gu et al., DIAVA: a traffic-based framework for detection of SQL injection attacks and vulnerability analysis of leaked data. *IEEE Trans. Reliab.* **69**(1), 188–202 (2020). <https://doi.org/10.1109/TR.2019.2925415>
14. P.R. McWhirter, K. Kifayat, Q. Shi, B. Askwith, SQL injection attack classification through the feature extraction of SQL query strings using a gap-weighted string subsequence kernel. *J. Inf. Secur. Appl.* **40**, 199–216 (2018). <https://doi.org/10.1016/j.jisa.2018.04.001>
15. Q. Li, F. Wang, J. Wang, W. Li, LSTM-based SQL injection detection method for intelligent transportation system. *IEEE Trans. Veh. Technol.* **68**(5), 4182–4191 (2019). <https://doi.org/10.1109/TVT.2019.2893675>

16. A. Avancini, M. Ceccato, Comparison and integration of genetic algorithms and dynamic symbolic execution for security testing of cross-site scripting vulnerabilities. *Inf. Softw. Technol.* **55**(12), 2209–2222 (2013). <https://doi.org/10.1016/j.infsof.2013.08.001>
17. L. Batista et al., Fuzzy neural networks to create an expert system for detecting attacks by SQL injection. *Int. J. Forensic Comput. Sci.* **13**(1), 8–21 (2018). <https://doi.org/10.5769/j201801001>
18. Kaggle Great SQLi Dataset, <https://www.Kaggle.com/karamsara/greatsql-dataset>

Modelling the Inhibitors of Online Learning Over 4G Networks: ISM-MICMAC and FMICMAC Analysis



L. Kala, T. A. Shahul Hameed, and V. R. Pramod

Abstract Online learning is a well-proven application in all walks of the teaching–learning process that depends on wireless mobile technology and Internet connectivity. It is accomplished through mobile handheld devices and wireless networks. It has proved its importance in this COVID pandemic era for distant learning to facilitate online classes for all university students. The first evolution of remote Internet-based learning was termed electronic learning (e-learning) with wired networked desktop computers. Mobile learning or m-learning is accomplished with wireless Internet connectivity-enabled laptops or mobile handheld devices. Now, this technology of the teaching–learning process is widely accepted as the education system of pandemic-affected world and is termed online learning. Online learning is a subset of mobile learning over wireless networks using portable electronic devices. Inhibitors caused due to the existing 4G wireless networks, mobile handheld devices, and the outlook of end users are the main factors selected for this particular study. Inhibitors of online learning over 4G networks were identified through an intensive study conducted by discussions with experts, end users, designers and learners. The methodology adopted for this research work is the fuzzy matrix of cross-impact multiplications applied to classification (MICMAC). The fuzzy-MICMAC method is used to analyse the selected inhibitors of online learning over wireless networks. A digraph with all possible interconnections, correlation study and the comparative analysis of fuzzy logic approach to the binary logic ISM-MICMAC method were implemented. More outlined results were achieved with the fuzzy analysis. The driving power-dependence abilities of these inhibitors were recorded, which will resolve many undue variables in the future implementation of the system.

L. Kala (✉) · V. R. Pramod
NSS College of Engineering, APJ Abdul Kalam Technological University, Kerala, India
e-mail: lkala@nssce.ac.in

V. R. Pramod
e-mail: pramodvr@nssce.ac.in

T. A. S. Hameed
Thangal Kunju Musaliar College of Engineering, APJ Abdul Kalam Technological University,
Kerala, India
e-mail: shahulhameed@tkmce.ac.in

Keywords 4G inhibitors · Online learning · ISM · Fuzzy · MICMAC · Driving power · Directional graph

1 Introduction

Online learning is an ICT-enabled wireless teaching–learning process that overrules the disadvantages of time, learning materials and distance. It is a 4-A learning system, where 4 A's are any time, anything, anywhere and anybody. It is possible to access this system at any time, or it is a 24×7 learning system. For a user, it is possible to connect to anywhere over the world from his present location if he has a mobile handheld device with Internet connectivity.

Further, it is possible to avail any idea, whether new or old, at his fingertip. Anybody indicates, irrespective of age, gender and profession, any person can use this technology. This system provides any formal and informal information around the world to its end users. A person interested in a particular field has abundant opportunities to understand and study his field of interest. Wireless networks used in nearly 70% of the world are 4G, and 3G is still in use in very few countries. It has many applications in information and communication technology-enabled education systems. The world witnessed a new mode of education technology with significant changes in the existing course delivery for online education. The never observed COVID-19 pandemic further accelerated this trend to combat the disruption caused in the classroom teaching–learning process. These days, many people use smartphones as handheld devices for their earnings through hobbies as bloggers, whether travel or beauty, with millions of subscribers. It is currently possible to share any multimedia information over the Web instantaneously; extensive use of mobile handheld devices and laptops enhanced the feasibility of online learning and the entertainment industry. All these parameters lift the need to adopt m-learning as an advanced technology, for the online teaching–learning process.

Most of the educational institutions and universities approved online classes on various platforms. Google Classroom, Google Meet, Webinar, MOOC, Moodle, NPTEL/MIT lectures, ZOOM, WebEx, Microsoft classroom, etc. Most software companies allowed work from home, and this platform is extensively helpful in conducting official meetings. In this work, the authors analysed the inhibitors of the most used 4G wireless networks in enabling online learning. Here, analysis was performed on all inhibitors of online learning, caused due to the influences of wireless mobile networks, trepidations with Internet access, and influences of 4G technology, user ability, issue's with mobile handheld devices and network connectivity. Fuzzy-based interpretive structural modelling (ISM) was used to analyse, as it is an appropriate tool to solve highly interrelated problems in complex situations. Fuzzy-ISM (FISM) can be effectively utilised to analyse the inhibitors while mobile handheld devices are used over the Internet. This analysis was executed to find out the interrelationships and associations among inhibitors of online learning [1].

Warfield [2] first put forward this method of ISM as a widely accepted and well-recognised analytical tool to model a selected system and analyse the influential variables and the interrelationships among them. It is a well-accepted technique for establishing interconnections of influencing elements. McQuiggan et al. [3], in Chapter 1 of their book, narrated how mobile devices such as smartphones and tablets have transformed human lives in many different ways and revolutionised education by using this promising technology in the classroom. The chapter also covers the concepts, challenges, benefits and associated inventiveness of mobile learning. Chapter 7 clarifies how developers can make mobile learning unique, occupy students and induce learning to a new-fangled format. It also dictates the hardware facilities like camera, GPS, video, Wi-Fi 3G connectivity, gyroscope, etc. and software features like accessibility, content, integration with other apps, security, etc. of mobile devices over former technologies. Its capability to augment functionality through the usage of accoutrements, iBeacons, Apple TV, wearable devices etc. are other exhilarating structures of mobile technology. Yang and Park [4] studied how various customer characteristics affect airline mobile application services' ease of use and usefulness. Customer demand for services is conducted with a survey of passengers using airline mobile applications by applying structural equation modelling and a maximum likelihood estimator. It is observed that a user's mental model has maximum effect on the study of the selected applications and concluded that it has a significant statistical impact on customer reception and resistance. Hwang and Fu [5] conducted an extensive review of the 'mobile technology-assisted language learning studies' in print from 2007 to 2016 in selecting SSCI journals. They included many research issues, methods, learning types, outcomes and language for this study.

Moreover, in terms of vocabulary, pronunciation and speaking in selected languages, the effectiveness of mobile learning was conducted. The authors reported that further studies are required to authorise its impressions on reading, listening and language learning as a whole. In this paper, big data collected from social media sites, and surveys are analysed to identify factors influencing customers' purchasing decisions [6]. It is employed using ISM and Fuzzy-MICMAC analysis. Authors gave recommendations to support retailers to plan strategies on the consumer-oriented supply chain. Traxler and Kukulska-Hulme [7], in their textbook *Mobile Learning*, explicated various fields related to context-aware mobile learning, with mobile and pervasive personal technologies and cell phones in 12 different chapters. The book covers topics on 'approaches to the integration of mobile contextual learning in diverse contexts; challenges, barriers, issues of sustainability; like as conceptions of context; technologies and applications'. Sushil [8] modified the comparatively interpretation of ISM using a method of total interpretive structural modelling (TISM). Firstly, the author first implemented an ISM model, and then it is integrated into TISM. It is displayed in a step-by-step style, and specific tests for authenticating total interpretive structural models are also recommended. An illustration in the perspective of organisational research is also executed. Pramod and Banwet [9] accomplished the modelling of the customer receptivity aspects of telecom service providers with the fuzzy-ISM method. Authors observed that qualitative benchmarks are time and again conveyed by obscurities and imprecision, and the outcomes are reinforced

with administrative repercussions. Wairiya et al. [10] described the importance of mobile handheld devices in teaching and learning processes and analysed the social and cultural implications of adopting m-learning in India. A survey was conducted with 57 instructors and 390 students across government and private educational institutions in India.

The authors observed that both of them have constructive perceptions in the direction of m-learning and then acknowledged that m-learning supports and enhances teaching–learning practice. Moreover, they identified several obstacles in the execution of m-Learning. Cai and Xia [11] explained the security and privacy concerns associated with the 5G network. This paper also deliberated significant developments of security issues of 5G and beyond and encounters for upcoming research. Swati and Abhishek [12] explored both public and private management education in India using interpretive structural model (ISM) and validated using TISM. Variables that affect the dynamics of college rankings were considered for this study and proposed. A study was completed on green supply chain management (GSCM) to isolate its barriers.

Further, a fuzzy matrix of cross-impact multiplications applied to classification (MICMAC) was used to reveal and identify the effect on direct and indirect factors by Dube and Gawande [13, 14]. A distinct work identified different key enablers of GSCMEs and developed an integrated interpretive model using the fuzzy-MICMAC approach. Authors coined that this study maintains substantial practical consequences for academicians, managers and practitioners to emphasise recognised GSCMEs for strategy formulation and implementation. A study on innovation enablers was implemented to analyse its influence to boost the effectiveness and perform a qualitative analysis. Using fuzzy-MICMAC (Matrice d’Impacts Croisé’s Multiplication Appliquée a UN Classement) analysis, Dewangan et al. [15] identified the prominent IEs to raise the competitiveness of 100 manufacturing sections across India.

Many papers on mobile learning were reviewed to collect ideas about the parameters to be selected for this study. Matzavela and Alepis [16] discussed the use of digital tools, the experience of teachers and learners and the challenges faced during online learning. Shrestha et al. studied involvement in online classes with the help of handheld digital devices such as smartphones and laptops through online platforms such Google Meet, telegram and WhatsApp [17]. Mohammadi et al. [18] evaluated the acceptance of mobile learning which was evaluated among a group of faculty members both qualitatively and statistically in their paper. Even though many such recent studies are noted in the literature, the technological aspects of handheld devices, wireless networks, and Internet connectivity that support such an online teaching–learning process were not evaluated. So the authors decided to think in that direction and proposed a further study with the technological parameters of mobile handheld devices. Bashar [19] conducted a comparative analysis of various networks with the essential parameters and tabulated the outcome with special mention to specific applications of 4G technology like mobile web broadcasting, video conferencing, IP telephony, cloud computing and gaming. Duraipandian [20] discovered the reasons behind the sudden call terminations in configuring the self-organising 4G-LTE and the methods to minimise the related factors and

improve voice communication quality. A study from a customer point of view was performed, emphasising the reliability study of the energy-efficient data transmission methods, with high security over an IoT, endowed wireless mobile network, with enhanced throughput [21]. Xixiang et al. [22] conducted and represented a detailed analysis of various representations and parameters of a triangular fuzzy model. Few most effective approaches, methods learner ability, learner behaviour and high order thinking learning skills. Also, suggestions for further research in the field were proposed by Chu-Lin [23]. Criollo-C et al. [24] well addressed the use of mobile learning in education, technology, behavioural use of mobile devices, and development of practical educational applications. It also covered the possibility of reducing complications in implementing it in education.

2 Research Methodology

The primary objective of this research is to analyse inhibitors of online learning, using mobile handheld devices over 4G wireless networks.

Subsequent objectives are

- To identify prominent inhibitors of online learning over 4G wireless networks.
- To establish interrelationships amongst inhibitors of online learning
- To apply fuzzy-MICMAC analysis for classification of inhibitors of online learning
- To evaluate driving and dependence power rules of online learning parameters.
- To discuss the repercussions of driving power and dependence.
- To propose guidelines for upcoming research work.

Fuzzy logic is beheld as a tool for addressing uncertainty related to a system, especially in the field of engineering and technology. It accounts for a third level apart from logic YES or NO opinions, which is termed a membership function (MF). This logic answers to the intricacies with which human thoughts are processed with so many conditional statements. Some sort of approximation is preferred over the exact reasoning approaches. One-to-one mapping of such rules are acquainted with fuzzy logic, which allow space for tolerance levels too. MF takes on real values between 0 and 1, indicating how much the variable associate to that fuzzy set. Most commonly used membership functions are triangular, trapezoidal and Gaussian. For this particular work, a triangular membership function is selected. MICMAC analysis was proposed with fuzzy data to establish complex interrelationships among a selected set of inhibitors of online learning over 4G networks. It is also expected to meet the objectives of this work to classify the selected factors based on the rules of driving and dependence powers. ISM is a Boolean algebra-based matrix theory under discrete mathematics. It models the relations between a selected set of elements under study. This methodology is proposed by Warfield to identify relations between various parameter sets [2]. Multiple factors are generally associated with

such complex problems. Intended to perform a comparison study of ISM-MICMAC with fuzzy-MICMAC analysis.

2.1 Implementation

The authors identified twelve inhibitors of online learning after an intensive literature review and after accumulating opinions from experts in the fields. They are indicated as inhibitors of online learning (IOL) and labelled as IOL1 to IOL 12 and recorded in Table 1.

The procedure for implementing the analysis of inhibitors of m-learning is carried out in various steps. A structural self-interaction matrix (SSIM) is developed with alphabet entries *V, A, X* and *O* and logic rules used are illustrated below. It is employed by bearing in mind the pair-wise correlation between the designated elements. Anecdotal relationships among a couple of elements were examined. To establish this circumstantial relationship, a law of ‘leads to’ is applied. Hence, four different conditions were examined

- *V*: parameter *i* leads *j*;
- *A*: parameter *j* leads *i*;
- *X*: parameter *i* and *j* leads to each other; and
- *O*: parameters *i* and *j* are unrelated.

The following statements explain the use of symbols *V, A, X* and *O* in SSIM. Here, variable 1 leads to 12, 10, 08, 07, 06, 05, 04 and 03; hence, (*V*) is allotted against those pairs in the table. Some other examples are variable 3 and 12 leads to each other, hence allotted (*X*); and since variables 09 and 11 are unrelated, (*O*) is assigned. Likewise, in this study, the relationship between each pair, $n(n-1)/2$

Table 1 Inhibitors of online learning

CODE	Inhibitors of online learning
IOL1	Rapid technological advancements
IOL2	Learning pattern
IOL3	Connectivity issues
IOL4	Acceptance time
IOL5	Attitude of learners
IOL6	Need for training
IOL7	Screen Size of handheld devices
IOL8	Cost of equipment
IOL9	Undue dependence hardware devices
IOL10	Security aspects
IOL11	Imprecise curriculum
IOL12	Lack of remote area coverage

pairs (132 sets), was checked with $n \times n$ question sets. An $n \times n$ square matrix with binary values 0 and 1 entries was formed using the logic. For any (i, j) pair, (1, 1) is assigned if 'X' is there in SSIM. (0, 0) is assigned for 'O' and (1, 0) for 'V' and (0, 1) for 'A' for (i, j) and (j, i) positions respectively. This binary adjacent matrix is termed as reachability matrix. After checking the transitivity over all pairs of this 12×12 matrix, the final reachability matrix was formed, and the transitivity links were identified. Transitivity matrix was created with binary 1 for transitivity links and zero elsewhere. The final reachability matrix (FRM) was created by doing a binary EX-OR operation for each cell.

2.2 Micmac Analysis

MICMAC is a matrix operational method, where multiple power matrices were created. In ISM-MICMAC, it was performed over binary FRM, and for each power of it, the row sum and column sum of all 12 rows and 12 columns were calculated and tabulated in Table 2 [1]. At a particular stage, the row sum and column sum levels, start repeating and then the matrix is stated to be at its stagnation state. The above step is repeated till stagnation is reached, and the matrix thus obtained is termed the stagnation matrix (Table 3). From the stagnation matrix, the driving power and dependence were calculated and the parameters with high driving-dependence powers were identified as the prominent elements of the system.

Table 2 IOL—final reachability matrix (FRM)

Binary FRM matrix											
1	0	1	1	1	1	1	1	1	1	0	1
0	1	0	1	1	0	0	0	0	0	0	0
0	0	1	1	1	0	0	0	0	0	0	1
0	0	1	1	0	0	0	0	0	1	0	1
0	1	1	1	1	0	1	0	1	1	1	1
0	1	0	1	1	1	0	1	1	1	0	1
0	1	0	1	1	0	1	0	1	0	0	0
0	1	1	1	1	1	1	1	1	0	0	0
1	0	0	1	1	1	1	1	1	0	0	1
0	0	0	1	1	1	0	1	1	1	0	0
0	1	0	1	1	1	0	0	1	0	1	0
0	0	1	0	0	1	0	0	0	0	0	1

Table 3 IOL—ISM-MICMAC stagnation matrix

R1	C1	R2	C2	R3	C3	R4	C4	R5	C5	R6	C6
10	2	65	10	390	54	2278	304	13,204	1740	76,374	57,862
3	6	16	35	81	196	457	1124	2614	6474	15,083	216,053
4	6	20	41	100	252	560	1482	3193	8617	18,402	288,528
4	11	17	67	95	393	540	2277	3115	13,165	17,989	439,844
9	10	48	56	281	326	1617	1884	9354	10,888	57,076	363,748
8	7	49	36	292	215	1704	1248	9874	7244	54,111	242,488
5	5	29	30	165	167	949	949	5480	5448	31,668	181,432
8	5	49	27	283	152	1624	875	9368	5055	54,103	168,960
8	8	55	44	327	250	1917	1436	11,105	8282	54,240	276,470
6	5	43	35	261	204	1539	1192	8941	6905	43,596	230,921
6	2	38	12	223	68	1299	394	7534	2278	43,596	76,097
3	7	15	51	84	305	476	1795	2740	10,426	15,807	349,072

2.3 Fuzzy-MICMAC Analysis

The objective of fuzzy-MICMAC analysis is to categorise the selected variables as per the driving power and dependence. This research work was proposed by incorporating the fuzziness of interrelationships among the inhibitors of online learning. The initial analysis with binary states was already performed. Inhibitors of online learning were classified into four clusters based on their driving power and dependence. Inter dependent levels in the fuzzy ranges were also considered in this work, where only two logic levels of 1’s and 0’s were considered for the former ISM-MICMAC analysis. Fuzzy levels between 0 and 1 were considered, which removed the ambiguities faced in the previous case. The triangular fuzzy levels were selected in progression steps of size 0.2 starting from 0.1. The normal values of a fuzzy system are incremented in steps in the order of 0.1, 0.3, 0.5, 0.7 and 0.9, with boundary values 0 and 1.

2.3.1 Triangular Fuzzy Number

The triangular fuzzy number (TFN) is represented by $\mu_{\tilde{M}}(x)$ and are defined by the Eqs. (1) and (2), where $\mu_{\tilde{M}}(x)$ is the membership values of the triangular fuzzy function corresponding to an element x [22]. Here, l and u represents the extreme points of the triangle and m the midpoint corresponds to the peak or maximum value of the MF function.

$$\tilde{M} = \sum_i^n \frac{\mu_{\tilde{M}}(x_i)}{x_i} \text{ OR } \tilde{M} = \{(x, \mu_{\tilde{M}}(x))\}_{x \in X}$$

\tilde{M} is a regular fuzzy convex set in the real domain follow the conditions:

- (i) $\mu_{\tilde{M}}(x_0) = 1$, for only one element x_0
- (ii) $\mu_{\tilde{M}}(x)$ is continuous

then \tilde{M} is a fuzzy number, which means ‘a real number that is approximated to x_0 ’.

\tilde{M} can be expressed as follows:

$$\mu_{\tilde{M}}(x) = \begin{cases} L(x), & l \leq x \leq m, \\ R(x), & m \leq x \leq r \end{cases} \tag{1}$$

where $L(x)$ is a continuous and increasing function to the right and $R(x)$ is also a continuous but decreasing function to the left. With the following conditions

$0 \leq L(x)$, $L(x)$ is expressed as $(x-l)/(m-l)$

$R(x) \leq 1$. $R(x)$ is represented as $(x-r)/(m-r)$

$\mu_{\tilde{M}}(x)$ can also be represented by left, middle and right threshold values. Here, the left threshold value a^l , the midpoint a^m , and the right threshold value a^u are used to represent a triangular fuzzy number \tilde{M} ,

$\tilde{M} = (a^l, a^m, a^u)$, and its membership function is as follows:

$$\mu_{\tilde{M}}(x) = \begin{cases} \frac{x-a^l}{a^m-a^l}, & a^l \leq x \leq a^m, \\ \frac{x-a^u}{a^m-a^u}, & a^m \leq x \leq a^u, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

For this particular work, the membership functions are selected as per Table 4, and the TFM is plotted in Fig. 1.

‘If Rule’ was assigned for fuzzification. If the condition satisfies for this linguistic statement range, then decision of fuzzified values corresponding to the entries in the binary FRM matrix is entered in Table 5. Fuzzy levels between 0 and 1 were considered, which removed the ambiguities faced in the previous case. The triangular fuzzy levels were selected in progression steps of size 0.2 starting from 0.1. Since the

Table 4 Assigned fuzzy layers for the model

Sl. No.	Linguistic statement (influence of inhibitor)	If rule (range of x)	FTMF
1	Extremely low	$x \leq 0.05$	0
2	Low	If $0.05 \leq x \leq 0.2$	0.1
3	Very low	If $0.15 \leq x \leq 0.45$	0.3
4	Medium	If $0.35 \leq x \leq 0.65$	0.5
5	High	If $0.55 \leq x \leq 0.85$	0.7
6	Very high	If $0.75 \leq x \leq 0.95$	0.9
7	Extremely high	$x \geq 0.95$	1.0

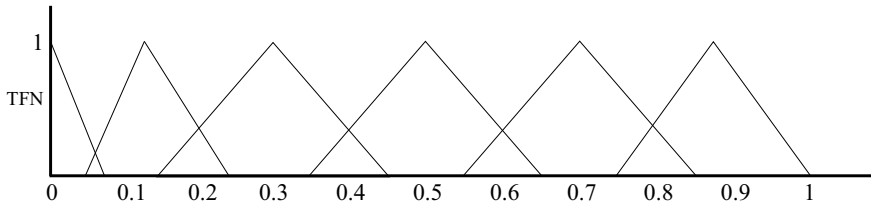


Fig. 1 TFN plot

Table 5 IOL—fuzzy-FRM

No.	1	2	3	4	5	6	7	8	9	10	11	12
1	1	0.1	0.7	1	0.9	0.9	0.7	0.5	0.9	0.3	0.3	0.9
2	0.3	1	0	0.7	0.7	0	0.3	0.5	0	0.7	0.5	0.1
3	0.5	0.4	1	0	0.9	0	0	0	0	0	0	1
4	0.8	0.9	0	1	0	0.5	0	0.1	0	0	0.3	0.1
5	0.7	1	0.3	1	1	0	1	0	1	1	1	1
6	0.9	0.7	0	1	1	1	0	0	0	1	0.1	0
7	1	0.1	0	0.2	0.3	0	1	0.5	0	0	0	0
8	0.5	0	0	0.1	0.2	0	0.1	1	0	0	0	0
9	0.3	0.1	0	0.5	0.9	0.7	1	0.7	1	0.2	0	0.5
10	0.1	0	0	0.3	0.4	0.2	0	0.1	1	1	0	0
11	0	1	0	0	0.5	0.9	0	0	0	0	1	0
12	0.3	0.7	1	0.1	0	0	0	0.1	0	0	0	1

data is collected through an extensive survey and from expert’s opinions, the mean value sometimes happens to be in even terms and values like 0.2, 0.4, 0.6, 0.8, but limited to very few.

The first step in the fuzzy-MICMAC analysis was to formulate a stagnation matrix. It was attained by taking the powers of FRM and calculated its row sum and column sum for every power. Row–column sums of this 12 × 12 FRM matrix’s powers at each stage were calculated as columns of a matrix and was denoted as R_i and C_i . The matrix thus obtained was termed the fuzzy-stagnation matrix. Values thus obtained are tabulated in Table 6. The first column of $R1$ corresponds to the row sum of all rows of the first power matrix of the final reachability matrix. $R2, R3, R4, R5$ and $R6$ corresponds to the row sums of second to the sixth power of FRM. Similarly, $C1–C6$ corresponds to first to sixth column sums of the powers of FRM. In this work, at the sixth power, stagnation was observed.

Table 6 IOL—fuzzy-MICMAC stagnation matrix

R1	C1	R2	C2	R3	C3	R4 (10 ³)	C4	R5 (10 ³)	C5 (10 ³)	R6 (10 ⁴)	C6 (10 ⁴)
8.2	6.4	41.53	31.78	209.785	158.638	1.0481	792.6225	5.2288	3.9571	2.6084	1.9744
4.8	6	22.22	30.11	108.553	151.603	0.5401	757.0496	2.696	3.7773	1.3457	1.8846
3.8	3	21.12	14.12	102.925	69.38	0.5076	343.3241	2.5212	1.7075	1.2556	0.8509
3.7	5.9	18.96	32.34	96.007	162.108	0.4821	810.4131	2.4092	4.0478	1.2028	2.0203
9	6.8	43.08	32.38	212.337	159.735	1.0585	797.3182	5.2797	3.9774	2.6341	1.9843
5.7	4.2	32.58	19.36	165.194	98.413	0.8271	493.7225	4.1333	2.4655	2.0634	1.2308
3.1	4.1	16.17	21.43	80.878	107.344	0.4051	535.3331	2.0244	2.6694	1.0104	1.3316
1.9	3.5	8.48	15.95	41.374	79.846	0.2064	399.8247	1.0309	1.9949	0.5145	0.9952
5.9	3.9	29.43	20.66	147.137	103.742	0.7341	514.8342	3.665	2.5667	1.8292	1.2808
3.1	4.2	15.86	22.1	79.727	108.583	0.3988	541.1929	1.9918	2.7029	0.9944	1.349
3.4	3.2	17.83	17.11	90.912	85.717	0.4543	427.3186	2.2681	2.1334	1.1324	1.0648
3.2	4.6	13.38	23.3	65.257	114.977	0.3208	570.1083	1.5898	2.8383	0.7911	1.415

3 Results and Discussion

Interrelations among the selected 12 inhibitors of online learning are shown in the directional graph (Digraph) in Fig. 2. Here, 88×2 edges are shown, which indicates all possible inter-linkage among the nodes. All incoming and outgoing links indicate the directivity and driving factors of the selected parameters. Here, the number of links are different for various nodes. It marks the depth by which each one influences

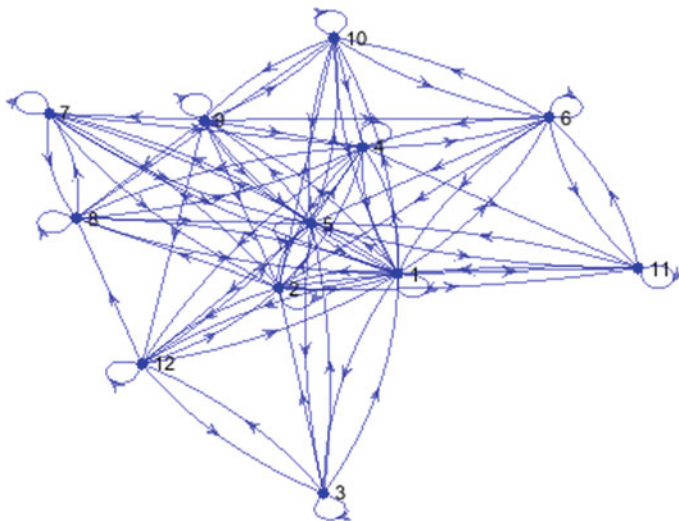


Fig. 2 IOL—directional graph

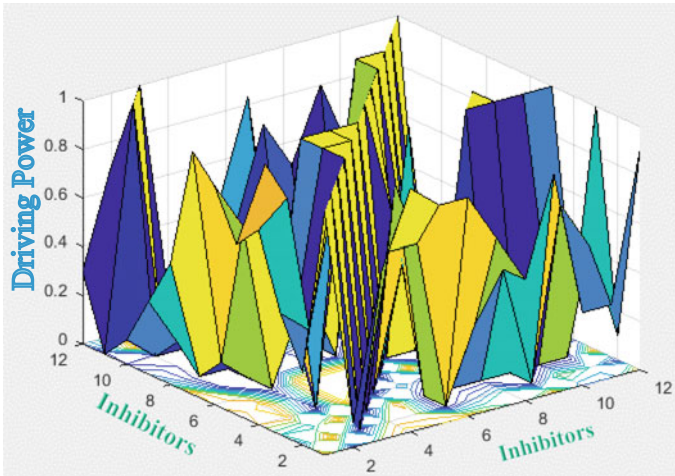


Fig. 3 IOL—fuzzy correlation plot

the other taken in pairs. Incoming arrows to say node 6 indicate 1, 4, 9, 10, 11 and self. Likewise, outward links indicate that node 6 influences 1, 2, 4, 5, 10 and 11. The bidirectional arrowheads show the bidirectional influence among specific pairs. In the case of node 6, they are 1, 10 and 11. Likewise, all possible transitions from all 12 nodes are depicted.

Further, analysis was done to study the correlation property of this system. A correlation matrix is created, and the corresponding correlation plot is made. It indicates that the elements are highly correlated. This analysis was performed with MATLAB, and the results were plotted with graphic functions. A 3D fuzzy-MATLAB plot is given in Fig. 3. Diagonal values show covariance, and the rest of the values represents the pair-wise correlation between the fuzzy values of the selected components. In the plot, the diagonal values are identical since it indicates a one-to-one relation, while others are of the comparison taken in (i, j) and (j, i) pairs along with the inhibitors. Values vary from 0 to 1, in steps of 0.1 for data points other than the extreme ones.

Level assignments performed from the stagnation matrix, corresponding to the stagnation level, and the ranking of elements from 1 to 12 row-wise and column-wise was taken and assigned across the corresponding inhibitor of online learning and is shown in Table 7. The bar plot, in Fig. 4, is termed as the driving power-dependence (DD) plot. Also, inhibitors 3, 8, 11 and 6 are the dependent elements of the system having maximum respective top-level values. It means that the four parameters, connectivity problems, equipment cost, ambiguity about curriculum, training requirements, are those highly dependent on other parameters.

Further, a comparative study was performed between the results obtained from ISM-MICMAC methodology [1] and this fuzzy-MICMAC method. In fuzzy, as said earlier, intermittent values were also included. Plots of the driving power of both the methods are separately shown in Fig. 5a, b. Driving power (DP) of 2, 3 and

Table 7 IOL—fuzzy-MICMAC DD levels

No.	Inhibitors of online learning	Driving power	Dependence
IOL1	Rapid technological advancements	2	3
IOL2	Learning pattern	5	4
IOL3	Connectivity issues	6	12
IOL4	Acceptance time	7	1
IOL5	Attitude of learners	1	2
IOL6	Need for training	3	9
IOL7	Screen size of handheld devices	9	7
IOL8	Cost of equipment	12	11
IOL9	Undue dependence hardware devices	4	8
IOL10	Security aspects	10	6
IOL11	Imprecise curriculum	8	10
IOL12	Lack of remote area coverage	11	5

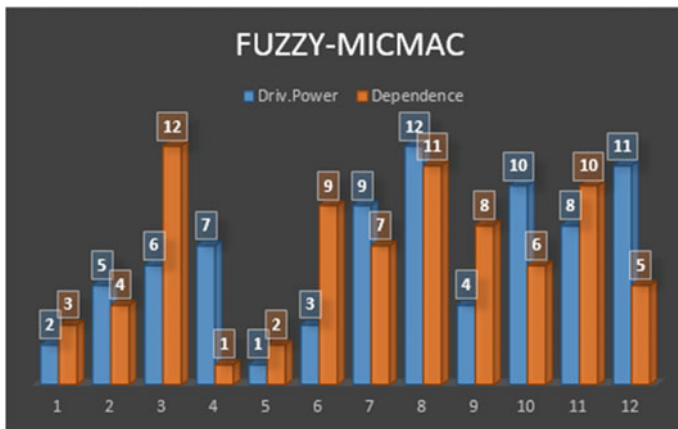


Fig. 4 IOL—DD plot

4 reduced and DP increased for 8 and 10 and remains unaltered for other factors. Dependence values are also plotted and shown in Fig. 5c, d. It is inverted for the first three parameters and observed plus or minus 1 to 2 variations for elements 6 to 12, while 4 and 5 remain unaltered. So with the fuzzy-MICMAC method, due to the effect of more levels of judgments included at the input, relative reflections or more condensed variations observed in the output.

At the next step, the clustering of elements into four different quadrants was accomplished. The driving power vs dependency plot showing autonomous, dependent, linkage and independent clusters, respectively, beginning from I to IV quadrants, taken in the anticlockwise direction and is displayed in Fig. 6. Consider

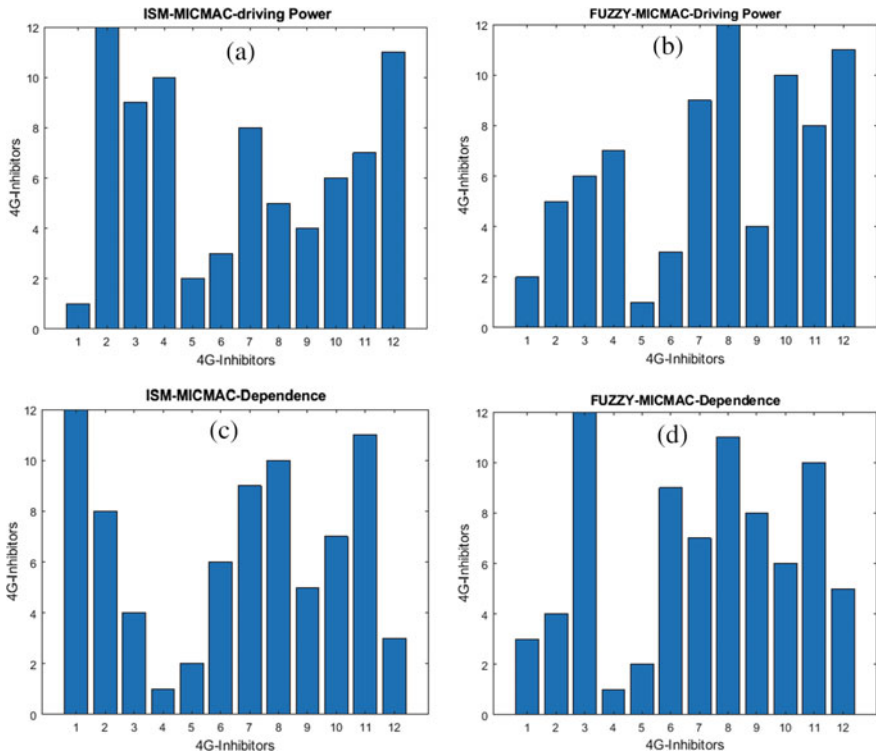


Fig. 5 IOL comparison plot **a** ISM-MICMAC driving power, **b** Fuzzy-MICMAC driving power, **c** ISM-MICMAC dependence, **d** fuzzy-MICMAC dependence

inhibitor 10 in this graph. Its point corresponds to driving power 10 and dependence 6. It indicates that IOL10 is with high driving power and moderate dependence. So it is allocated the point at (10, 6). Likewise, all other inhibitors are positioned at their own corresponding points. It is inferred that IOL1, IOL2, IOL5 are the autonomous elements of the system, with the lowest driving power and dependence. So all these parameters of cluster I become detached from the system. Cluster II elements are IOL3, IOL6 and IOL9, which were observed with low driving power and high dependence. Cluster III elements, IOL7, IOL8, and IOL11 are the linkage elements that exhibited high dependence and high driving powers, and hence the most critical parameters under consideration. Cluster IV with IOL4, IOL10 and IOL12 is termed independent cluster, with the independent elements of the system. Any variations in those parameters will drastically affect the system. Hence, they are listed as the most influential parameters of the system. So a design engineer should take care while making further changes in those parameters for optimum design. Those parameters having the maximum driving power influence other parameters of the system. A shaded portion of the transition region is indicated, where the fuzziness overlapping occurs over the system elements. Fuzzy-MICMAC analysis further substantiated

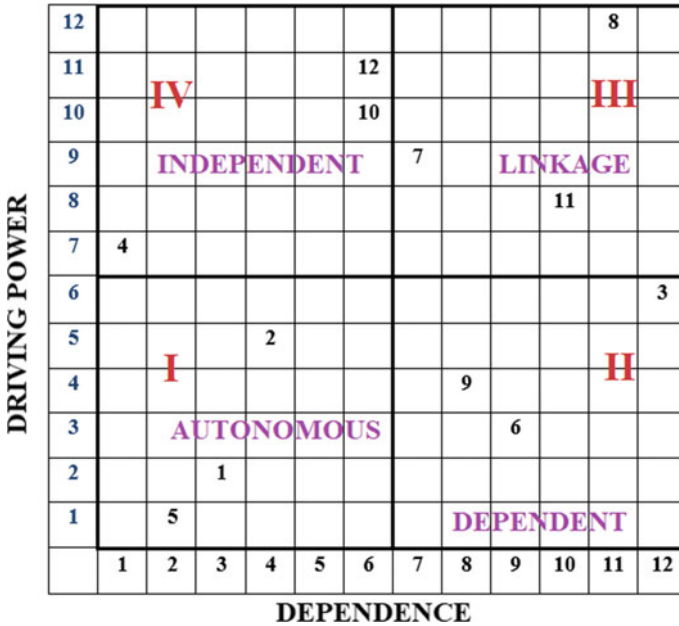


Fig. 6 IOL DD diagram—fuzzy-MICMAC

the results of ISM-MICMAC performed earlier. With this analysis, it is proved that certain IOLs in and near the transition ranges are still in the well-regulated and manageable positions.

Rapid technological advancements, learning patterns and learners’ attitudes are the autonomous parameters with the lowermost dependence on other parameters. High-dependence parameters are connectivity issues, need for training and undue dependence on hardware devices. Screen size of handheld devices, cost of equipment, imprecise curriculum are the linkage parameters with high driving power and high-dependence values. Acceptance time, security aspects and remote area coverage are the driving parameters factors of the system. In this case, the most prominent parameter is the lack of remote area coverage. Network design engineers, mobile device manufacturers, software developers, service providers etc. should consider these facts while designing this system that enables the online teaching–learning process over the wireless networks with mobile handheld devices. An end user using the system setup should also be aware of the inhibitors of the system.

3.1 Limitations

The variables for this study were collected from literature and expert opinion. After further deliberations and many discussions, 18 variables were identified, and some

were omitted and finalised to 12. The experts' opinion was not statistically evaluated for this work. It can be incorporated with appropriate tools. Also, as per the advances in technology, the importance and influence of the variables may get affected. All of a sudden, the number of people accessing the network increased, and the bandwidth deficiency or connectivity problem is reported by many users. In most places where 4G networks are in use, live streaming with many participants is not possible. Even some service providers limit the number of persons who access the system at a particular time for single use. Google Meet can be given as a specified example. So, more points can be added or deleted as per the technological changes. Even with all these listed limitations, the authors hope that this study will direct to further research in this area.

3.2 Future Scope

Nowadays, the existing technology limits the number of persons who can log in to an online class due to bandwidth problems. When 5G technology is implemented worldwide, it will revolutionise the way we do things, not only the teaching–learning process but also any real-time application. With abundant bandwidth, high data rate and low latency of less than one millisecond, the next-generation 5G network expects to address and revolutionise how we are using this system for any selected real-time application like online learning. 5G technology is already implemented in 30% of world countries, and the rest of the countries will switch over to the new technology shortly. Further research in this field is anticipated with the advent of 5G technology in all real-time application fields such as IoT and vehicle-to-vehicle communication, as a way out to many such solicitation human beings stumble across.

4 Conclusion

4G technologies are the recent day wireless technology used in most of the countries over the world. COVID-19 pandemic constrained people to bind to the four walls of their residence for all repercussions in their life, including the teaching–learning process. The advancements in Internet technology, wireless networks and mobile devices made things possible in this extraordinary juncture of human life. This work aimed to bring out the inhibitors of adopting online learning in this pandemic era, using the fuzzy-MICMAC method. This form of technological advancements may lead to adopting the teaching–learning process fully in an online mode for knowledge delivery. This will open up ample scope for any students attending the top-rated academicians at the same time from anywhere over the world. A comparative study of ISM-MICMAC and fuzzy-MICMAC methods was executed. The results were given to experts for validation as the modelling of this online learning with this selected tool is a novel approach. Hence, there was no other way to compare with any existing

results. In this context, the proposed method will help the stakeholders recognise and consider the main inhibitors of any system as a crucial point of consideration. The authors hope these results will help the designers and service providers think in that direction to eliminate the most prominent inhibitors in the future networks and enhance this application to benefit end users.

References

1. L. Kala, H.T.A. Shahul, V.R. Pramod, Analysis of inhibitors of mobile-learning over 4G wireless networks, with interpretive structural modelling (ISM) and ISM-MICMAC methods. *Int. J. Adv. Res. Eng. Technol.* **12**, 77–94 (2021)
2. J.N. Warfield, Developing interconnection matrices in structural modeling. *IEEE Trans. Syst. Man Cybern.* **4**, 81–87 (1974)
3. S. McQuiggan, J. McQuiggan, J. Sabourin, L. Kosturko, *Mobile Learning: A Handbook for Developers, Educators, and Learners* (Wiley Publications, 2015)
4. H.S. Yang, J.W. Park, A study of the acceptance and resistance of airline mobile application services: with an emphasis on user characteristics. *Int. J. Mobile Commun.* **17**, 24–43 (2019)
5. G.J. Hwang, Q.K. Fu, Trends in the research design and application of mobile language learning: a review of 2007–2016 publications in selected SSCI journals. *Interact. Learn. Environ.* **27**, 1–15 (2018)
6. N. Mishra, A. Singh, N.P. Rana, Y.K. Dwivedi, Interpretive structural modelling and fuzzy MICMAC approaches for customer centric beef supply chain: application of a big data technique. *Prod. Planning Control* **28**, 945–963 (2017)
7. J. Traxler, A. Kukulska-Hulme, *Mobile Learning: The Next Generation*, vol. 45284 (Routledge, 2016), pp. 1–236
8. Sushil, Interpreting the interpretive structural model. *Global J. Flexible Syst. Manage.* **13**, 87–106 (2012)
9. V.R. Pramod, D.K. Banwet, FISM for analysing the interrelationships between customer receptivity aspects. *Int. J. Bus. Excell.* **7**, 549–564 (2014)
10. M. Wairiya, A. Shah, G.P. Sahu, Mobile learning adoption: an empirical study, in *Proceedings of 10th International Conference on Cloud Computing, Data Science and Engineering* (2020), pp. 757–761
11. Y. Cai, C. Xia, Interpretive structural analysis of interrelationships among the elements of characteristic agriculture development in Chinese rural poverty alleviation. *Sustainability* **10** (2018)
12. Y. Swati, B. Abhishek, Benchmarking model for management education in India: a total interpretive structural modeling approach. *Benchmarking: Int. J.* **24**, 666–693 (2017)
13. A.S. Dube, R.R. Gawande, Analysis of green supply chain barriers using integrated ISM-fuzzy MICMAC approach. *Benchmarking: Int. J.* **23**, 1558–1578 (2016)
14. A.S. Dube, R.R. Gawande, ISM-fuzzy MICMAC approach for analysis of GSCM enablers. *Int. J. Logistics Syst. Manage.* **24**, 426–451 (2016)
15. D.K. Dewangan, R. Agrawal, V. Sharma, Enablers for competitiveness of Indian manufacturing sector: an ISM-fuzzy MICMAC analysis. *Proc. Soc. Behav. Sci.* **189**, 416–432 (2015)
16. V. Matzavela, E. Alepis, M-learning in the COVID-19 era: physical vs digital class. *Educ. Inf. Technol.* 1–21 (2021)
17. S. Shrestha, S. Haque, S. Dawadi, R.A. Giri, Preparations for and practices of online education during the Covid-19 pandemic: a study of Bangladesh and Nepal. *Educ. Inf. Technol.* 1–23 (2021)
18. M. Mohammadi, M.S. Sarvestani, S. Nouroozi, Mobile phone use in education and learning by faculty members of technical-engineering groups: concurrent mixed methods design. *Front. Digital Educ.* **5** (2020)

19. A. Bashar, An efficient cell selection approach in 4G networks. *J. Trends Comput. Sci. Smart Technol. (TCSST)* **2**, 188–196 (2020)
20. M. Duraipandian, Long term evolution—self organizing network for minimization of sudden call termination in mobile radio access networks. *J. Trends Comput. Sci. Smart Technol. (TCSST)* **2**, 89–97 (2020)
21. N. Bhalaji, Reliable data transmission with heightened confidentiality and integrity in IOT empowered mobile networks. *J. ISMAC* **2**, 106–117 (2020)
22. X. Zhang, W. Ma, L. Chen, New similarity of triangular fuzzy number and its application. *Sci. World J.* **2014** (2014)
23. L. Chu-Lin, Trends of mobile learning: a review of the top 100 highly cited papers. *Br. J. Edu. Technol.* **51**, 721–742 (2019)
24. S. Criollo-C, A. Guerrero-Arias, A. Jaramillo-Alcázar, S. Luján-Mora, Mobile learning technologies for education: benefits and pending issues. *Appl. Sci.* **11**, 1–17 (2021)

An Improved Model for Clarification of Geospatial Information



Khudov Hennadii, Butko Igor, Makoveichuk Oleksandr, Khizhnyak Irina, Khudov Vladyslav, Yuzova Iryna, and Solomonenko Yuriy

Abstract The article formulates an improved mathematical model for clarification of imagery for an area of the earth's surface. An improved mathematical model for clarification of geospatial information has been formulated in general form, which can be presented as a result of the action of an operator that transforms coordinates, operators that conduct image clustering, and operators that carry out zoning according to some criterion. The co-ordinates transformation operator, clustering operators, zoning operators, and their explicit form are presented. This model, based on the model of forming imagery, performs the inverse transformation of the imagery coordinates into spatial coordinates and clustering imagery into particular classes according to their texture and color. This takes into account geographic zoning. A model for clarification of geospatial information in operator form is obtained.

Keywords Geographic information system · Clarification · Imagery · Model · Geospatial information and coordinates

K. Hennadii (✉) · K. Irina · Y. Iryna · S. Yuriy
Ivan Kozhedub Kharkiv National Air Force University, Kharkiv, Ukraine
e-mail: 2345kh_hg@ukr.net

K. Irina
e-mail: khizh_ia@ukr.net

Y. Iryna
e-mail: uzik25@ukr.net

S. Yuriy
e-mail: solomon69@ukr.net

B. Igor
State Enterprise "State Land Cadastral Center", Kyiv, Ukraine
e-mail: butko_igor@ukr.net

M. Oleksandr · K. Vladyslav
Kharkiv National University of Radio Electronics, Kharkiv, Ukraine

1 Introduction

Today, Earth remote sensing data are geospatial information presented in digital form in the form of raster images [1]. This geospatial information is now freely available to a large number of users and is used not only in scientific research, but also in other spheres of human activity [2–6]. Such data are also the main source of up-to-date and operational data for geographic information systems. A geographic information system contains data about spatial objects in the form of their digital representations. A geographic information system is also a large class of information system that allows you to work with spatial data. In modern geographic information systems, complex processing of such information is carried out—from collection to storage, presentation, and updating [1].

Solving the problem of thematic processing of imagery is an analysis stage, that is, image clarification after preliminary digital processing. The end result of solving such a problem is the implementation of the process of identifying and recognizing objects. In practice, this process is called image decryption [4].

The result of the decryption is a thematic map of the territory, which was presented in the imagery with the interpreted recognized objects applied. Interpretation can be carried out using visual and automated models of clarification of imagery. Interpretation of information usually comes down to transforming coordinates, clustering images, and highlighting objects of interest. Examples of interpreting various information are discussed in many articles.

Paper [7] discusses the complaints registration system. The main focus of the project is about the pothole-related complaints. The application [7] will give easy access to people to put their complaints toward the government. But the results [7] cannot be applied to the interpretation of information in views.

In order to maximize the influence in social networks, an interest-based algorithm with parallel social action has been proposed in the paper [8]. This algorithm enables identifying influential users in social network. But the results [8] cannot be applied to the interpretation of information in views.

Extreme learning machine is one of the latest trends in learning algorithm, which can provide a good recognition rate within less computation time [9]. In [9], the extreme learning machine method has been designed with the presence of sigmoidal function of biases in the hidden nodes to perform the classification task. The modified version of extreme learning machine has been developed to obtain better accuracy and minimize the classification error. Paper [9] includes the mathematical proof of sigmoidal activation function with biases of the hidden nodes present in the networks. But the results [9] cannot be applied to the interpretation of information in views.

Paper [10] is mainly focused on development of domain ontology for issuing Citizenship of Nepal as e-government is widely considered as a good example of heterogeneous system. Protégé is used as ontology editor, and Web Ontology Language is used for representation of the concepts. A hybrid methodology with Unified Modeling Language is used to get in-depth concept of the domain. The domain ontology is verified using built-in reasoner of Protégé and validated with answering competency

questions of the domain using SPARQL query. But the results [10] cannot be applied to the interpretation of information in views.

2 Problem and Presentation Materials Researching

2.1 General Form a Mathematical Model for Formation of Imagery

In general form, a mathematical model for formation of imagery I for a plot of the earth's surface g can be presented as a result of the action of two operators:

- the operator \widehat{T} that performs coordinate transformation:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \widehat{T} \begin{pmatrix} \varphi \\ \theta \\ h \end{pmatrix}, \tag{1}$$

where $\begin{pmatrix} \varphi \\ \theta \\ h \end{pmatrix}$ —vector of geographic coordinates (longitude, latitude, and altitude);

$\begin{pmatrix} x \\ y \end{pmatrix}$ —vector of coordinates on the image.

- the operator $\widehat{\Phi}$ that determines the brightness of the corresponding image element for a given element of the earth's surface g in the spectral channel c :

$$I_c = \widehat{\Phi}_c(g). \tag{2}$$

A mathematical model of thematic clarification of imagery (constructing a set of binary masks M for geosigns from imagery I) in general form can be presented as a result of the actions of the following operators:

- the operator \widehat{T}^{-1} (inverse to the operator \widehat{T}) that performs the transformation of coordinates:

$$\widehat{T}^{-1} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \varphi \\ \theta \\ h \end{pmatrix}, \tag{3}$$

- operators \widehat{C}_k that cluster the image according to some criterion k :

$$M_k = \widehat{C}_k(I), \tag{4}$$

- operators \widehat{Z}_l that zoning to some criterion l :

$$M_k = \widehat{C}_k(I), \tag{5}$$

Let us consider in more detail the introduced operators and their explicit form.

2.2 The Coordinates Transformation Operator

This operator \widehat{T}^{-1} is the inverse of the operator \widehat{T} . The explicit form of this operator is given by expression (6):

$$\begin{aligned} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} &= \begin{pmatrix} f/p_x \tan \chi & c_x \\ 0 & f/p_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{R}^T & -\mathbf{R}^T \mathbf{t} \\ \mathbf{0}^T & 1 \end{pmatrix} \\ &\times \left(\left[R_{Earth} \begin{pmatrix} \sin \theta \cos \varphi \\ \sin \theta \sin \varphi \\ \cos \theta \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ h \end{pmatrix} - \begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \end{pmatrix} \right] \right), \end{aligned} \tag{6}$$

where p_x, p_y —pixel dimensions of the scanner matrix;

χ —the tilt angle of pixel;

(c_x, c_y) —coordinates of the main point;

f —the focal length of the scanner;

R_{Earth} —the radius of the Earth.

In (6), \mathbf{R}^T —the transposed 3×3 matrix of the scanner plane rotation is relative to the Earth’s surface (the angle between the corresponding normals); \mathbf{t} —the 3-vector of the displacement of the scanner center relative to its projection onto the Earth’s surface, $\mathbf{0}^T$ —the transposed zero 3-column vector (i.e., the zero three-row vector).

To reflect the fact that the matrix $\begin{pmatrix} \mathbf{R}^T & -\mathbf{R}^T \mathbf{t} \\ \mathbf{0}^T & 1 \end{pmatrix}$ is formed by complex objects, they are highlighted in bold (as opposed to the usual number one in the lower right corner of the matrix).

Note that finding the inverse function by expression (6) is impossible for one separate point (due to the loss of information). Therefore, it requires a set of points. The solution is then found as a solution to the optimization problem. Finding an

operator \widehat{T}^{-1} that allows identifying image points with the corresponding points on the Earth's surface is called the georeferencing task [11–13].

The task of georeferencing to a certain extent is inverse to the calibration task: having sets of corresponding points (7):

$$\begin{pmatrix} x \\ y \end{pmatrix}_i = \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}_i, \quad (7)$$

since the earth's surface is not flat, we compensate for nonlinear distortions (8)–(11) [14]:

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix}_i = \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}_i, \quad (8)$$

$$X'_i = (X_i - \bar{X}_i)(1 + k_1 r_i^2 + k_2 r_i^4) + \bar{X}_i, \quad (9)$$

$$Y'_i = (Y_i - \bar{Y}_i)(1 + k_1 r_i^2 + k_2 r_i^4) + \bar{Y}_i, \quad (10)$$

$$r_i^2 = (X_i - \bar{X}_i)^2 + (Y_i - \bar{Y}_i)^2, \quad (11)$$

where X'_i, Y'_i —coordinates with compensated nonlinearity;

\bar{X}_i, \bar{Y}_i —average values for X_i, Y_i , respectively;

k_1, k_2 —parameters of radial distortions.

And, we find a matrix P^{-1} such that it minimizes the square of the norm. (Note that we do not invert the matrix. This designation simply emphasizes the connection with the previously introduced matrix P):

$$\min_P \sum_i \left\| P^{-1} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}_i - \begin{pmatrix} X' \\ Y' \\ Z \\ 1 \end{pmatrix}_i \right\|^2. \quad (12)$$

So,

$$\begin{pmatrix} X' \\ Y' \\ Z \\ 1 \end{pmatrix} = P^{-1} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}. \tag{13}$$

and since the parameters of radial distortions k_1, k_2 are known,

$$X = \frac{X' - \bar{X}}{1 + k_1 r_i^2 + k_2 r_i^4} + \bar{X}, \tag{14}$$

$$Y = \frac{Y' - \bar{Y}}{1 + k_1 r_i^2 + k_2 r_i^4} + \bar{Y}, \tag{15}$$

where r_i^2 is found by expression (11).

Let us move from coordinates $\begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$ to coordinates $\begin{pmatrix} \varphi \\ \theta \\ h \end{pmatrix}$:

$$\varphi = a \tan \frac{Y}{X},$$

$$\theta = a \sin \frac{\sqrt{X^2 + Y^2}}{R_{\text{Earth}}}, \tag{16}$$

$$h = Z - R_{\text{Earth}} \cos a \sin \frac{\sqrt{X^2 + Y^2}}{R_{\text{Earth}}}.$$

Thus, taking into account all of the above, it is proposed to set the explicit detailed form of the coordinate transformation operator \hat{T}^{-1} in the form (17):

$$\begin{pmatrix} X' \\ Y' \\ Z \\ 1 \end{pmatrix} = P^{-1} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix},$$

$$X = \frac{X' - \bar{X}}{1 + k_1 r_i^2 + k_2 r_i^4} + \bar{X},$$

$$Y = \frac{Y' - \bar{Y}}{1 + k_1 r_i^2 + k_2 r_i^4} + \bar{Y}, \tag{17}$$

$$\begin{pmatrix} \varphi \\ \theta \\ h \end{pmatrix} = \begin{pmatrix} a \tan \frac{Y}{X} \\ a \sin \frac{\sqrt{X^2+Y^2}}{R_{Earth}} \\ h = Z - R_{Earth} \cos a \sin \frac{\sqrt{X^2+Y^2}}{R_{Earth}} \end{pmatrix},$$

where the matrix P^{-1} is found as a solution to the georeferencing task.

2.3 Clustering Operators

Consider the explicit detailed form of the operator $\widehat{\Phi}$ introduced above. It determines the brightness of the corresponding image element for a given element of the Earth’s surface (18):

$$I(X, Y, c) = S \left(\alpha_c + \beta_c \sum_{k'} \rho_{k,c} \delta_{k,k'} \cdot \frac{1}{4\pi} \iint_{\Omega} \left(\frac{1 + \Omega_x \frac{\partial h}{\partial X} + \Omega_y \frac{\partial h}{\partial X}}{\sqrt{1 + \left(\frac{\partial h}{\partial X}\right)^2 + \left(\frac{\partial h}{\partial X}\right)^2}} \right) d\Omega \right), \tag{18}$$

where c —the spectral channel;

$S(i)$ —the function of the sensitivity of the scanner;

ρ —the coefficient of reflection of light from the surface. It depends on which class k the object belongs to and on the spectral channel c , that is $\rho = \rho(k, c)$.

At first, consider the multiplier

$$w = \frac{1}{4\pi} \iint_{\Omega} \left(\frac{1 + \Omega_x \frac{\partial h}{\partial X} + \Omega_y \frac{\partial h}{\partial X}}{\sqrt{1 + \left(\frac{\partial h}{\partial X}\right)^2 + \left(\frac{\partial h}{\partial X}\right)^2}} \right) d\Omega.$$

This multiplier obviously depends only on the surface geometry and does not depend on the reflection coefficient. In this case, in the model for clarification, it will be responsible for the texture of the imagery. If each surface object has its own geometry, then w will depend on the class of the object k , that is, $w = w_k$. And in this case (18) can be rewritten as:

$$I(X, Y, c) = S \left(\alpha_c + \beta_c \sum_{k'} \rho_{k,c} w_k \delta_{k,k'} \right). \tag{19}$$

So, operators \widehat{C}_k that cluster an image should return binary masks of an object’s class k depending on the color of the image and texture. This is done for all spectral

channels and is implemented using multispectral image segmentation techniques. It should be noted that these methods require a preliminary training stage. This stage is needed in order to identify each class of image segmentation (the corresponding binary mask) with a specific geo-sign.

Most often, the k -means segmentation method is used for this purpose, the essence of which is the division of n observations into k clusters. The division is performed in such a way that each observation belongs to the cluster with the closest mean value [15–18].

Let us consider this method in more detail.

Suppose we have a set of observations $\mathbf{x} = (x_1, x_2, \dots, x_n)$ where each observation is a d -dimensional real vector. The goal of clustering is to divide n observations into k sets (moreover $k \leq n$) in such a way as to minimize the sum of the squared distances within the cluster (i.e., variance). Formally, the goal is to find such a decomposition into sets S :

$$S = \arg \min_S \sum_{i=1}^k \sum_{x \in S_i} \|\mathbf{x} - \mu_i\|^2 = \arg \min_S \sum_{i=1}^k |S_i| \text{var} S_i, \tag{20}$$

where μ_i —average over set S_i . $|S_i|$ denotes the number of elements in a set of S_i .

This is equivalent to minimizing the pairwise standard deviations for each cluster (expression (21)):

$$S = \arg \min_S \sum_{i=1}^k \frac{1}{2|S_i|} \sum_{x, y \in S_i} \|\mathbf{x} - \mathbf{y}\|^2. \tag{21}$$

This expression follows from the fact that

$$\sum_{x \in S_i} \|\mathbf{x} - \mu_i\|^2 = \sum_{x \neq y \in S_i} (\mathbf{x} - \mu_i)^T (\mu_i - \mathbf{y}). \tag{22}$$

Since the total variance is a constant that does not depend on the order of summation, expression (21) is equivalent to maximizing the sum of square deviations between points in different clusters (micro-cluster sum of squares, BCSS [19]).

The simplest version of the algorithm (i.e., naive k -means) uses an iterative refinement method [15–17]. At the initial moment of the algorithm operation, the centers of the clusters are randomly selected. Then, for each element of the set, the distance from the centers is iteratively calculated with the attachment of each element to the cluster with the nearest center. The following two steps are performed sequentially at each iteration t :

- (1) step matching—matching each point with a cluster, the distance to the center of which is the minimum:

$$S_i^{(t)} = \left\{ x_p : \|x_p - \mu_i^{(t)}\|^2 \leq \|x_p - \mu_j^{(t)}\|^2 \forall j, 1 \leq j \leq k \right\}. \tag{23}$$

where each x_p maps to exactly one cluster $S_i^{(t)}$, even if it could have been assigned to two or more

- (2) update step—recalculation of mean values for observations, compared with each cluster:

$$\mu_i^{(t+1)} = \frac{1}{|S_i|} \sum_{x_j \in S_i} x_j. \tag{24}$$

This method has such obvious advantages as simplicity and speed of execution. But there are also significant disadvantages, namely:

- the classification result strongly depends on the random initial positions of the cluster centers;
- the algorithm is sensitive to outliers that can distort the average;
- the number of clusters must be predetermined by the researcher.

An additional disadvantage of the simple method is the violation of the connectivity condition for the elements of one cluster. Therefore, various modifications of the method are being developed, as well as its fuzzy counterparts (fuzzy k -means methods). In such analogs of the method, at the first stage of the algorithm, one element of the set may belong to several clusters (with different degrees of membership) [20].

2.4 Zoning Operators

These operators \widehat{Z}_l simply define the appropriate sets of binary masks in geographic coordinates $\begin{pmatrix} \varphi \\ \theta \\ h \end{pmatrix}$. These operators are used to define areas of interest, highlight administrative areas, and more.

So, considering all of the above, the improved model for clarification of geospatial information in operator form is given as:

$$M(\varphi, \theta, h) = \left\{ \bigcup_k \widehat{C}_k \left\{ I \left(\widehat{T}^{-1} \{x, y\} \right) \right\} \bigcup_i \widehat{Z}_l \{ \varphi, \theta, h \} \right\}, \tag{25}$$

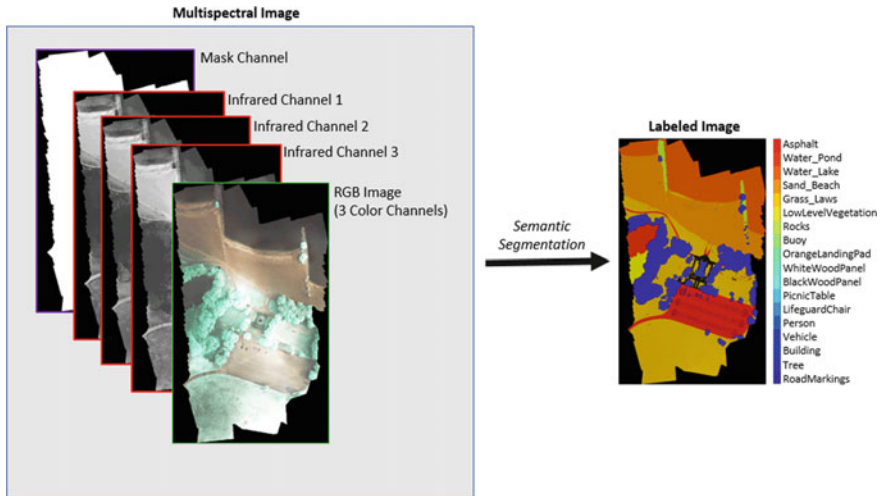


Fig. 1 Example of clarification of the original multispectral imagery using an improved model

where the operator \hat{T}^{-1} is specified by expressions (25), operators \hat{C}_k are proposed to be implemented in the form of a semantic segmentation method, and operators \hat{Z}_l specify geographic zoning.

An example of clarification of the original multispectral imagery using the improved model is shown in Fig. 1. The images in Fig. 1 are the images from WordView-3 [21].

Based on the results of clarification of the original multispectral imagery, the percentage of vegetation cover in the region was calculated (Fig. 1).

Thus, the article formulates in a general form a mathematical model of the formation of imagery for a section of the earth’s surface. A mathematical model of the formation of imagery is a result of an operator that transforms coordinates and operators that determine the brightness of the corresponding image element for a given element of the earth’s surface in the spectral channel.

A mathematical model for clarification of imagery has been formulated in general form, which can be presented as a result of the action of an operator that transforms coordinates, operators that conduct image clustering, and operators that carry out zoning according to some criterion. The coordinate transformation operator, clustering operators, zoning operators, and their explicit form are considered. A model for clarification of imagery in operator form is obtained.

References

1. Z. A-Xing, Z. Fang-He, L. Peng, Q. Cheng-Zhi, Next generation of GIS: must be easy. *Annals GIS* 27, 71–86 (2021). <https://doi.org/10.1080/19475683.2020.1766563>

2. I. Ruban, O. Makoveichuk, V. Khudov, I. Khizhnyak, H. Khudov, I. Yuzova, Y. Drob, The method for selecting the Urban infrastructure objects contours, in *6th International Scientific Practical Conference Problems of Infocommunications, Science and Technology* (Kiev, 2019), pp. 689–693
3. H. Khudov, S. Glukhov, V. Podlipaiev, V. Pavlii, I. Khizhnyak, I. Yuzova, The multiscale image processing method from on-board earth remote sensing systems based on the artificial bee colony algorithm. *IJATCSE* **9**(3), 2557–2562 (2020)
4. W. Fu, J. Ma, P. Chen, F. Chen, Remote sensing satellites for digital earth, ed. by H. Guo, M.F. Goodchild, A. Annoni. *Manual of Digital Earth* (Springer, Singapore, 2020), pp. 55–123. https://doi.org/10.1007/978-981-32-9915-3_3
5. H. Khudov, O. Makoveychuk, I. Khizhnyak, A. Yuzova, A. Irkha, V. Khudov, The mosaic sustainable marker model for augmented reality systems. *IJATCSE* **9**(1), 637–642 (2020)
6. I. Ruban, H. Khudov, O. Makoveichuk, I. Khizhnyak, V. Khudov et al., Segmentation of optoelectronic images from on-board systems of remote sensing of the Earth by the artificial bee colony method. *Eastern Eur. J. Enterp. Technol.* **2**(9–98), 37–45 (2019)
7. V. Patel, D. Kapadia, D. Ghevariya, S. Pappu, All India grievance redressal app. *J. Inf. Technol. Digital World* **2**(2), 91–99 (2020). <https://doi.org/10.36548/jitdw.2020.2.002>
8. D. Sivaganesan, Novel influence maximization algorithm for social network behavior management. *J. ISMAC* **3**(01), 60–68 (2021)
9. S.R. Mugunthan, T. Vijayakumar, Design of improved version of sigmoidal function with biases for classification task in ELM domain. *J. Soft Comput. Paradigm (JSCP)* **3**(02), 70–82 (2021)
10. R. Bastola, P. Campus, Developing domain ontology for issuing certificate of citizenship of Nepal. *J. Inf. Technol.* **2**(02), 73–90 (2020)
11. J. Schott, *Remote Sensing: The Image Chain Approach*, 2nd edn. (Oxford University Press, Oxford, 2007)
12. P.A. Burrough, R.A. McDonnell, *Principles of Geographical Information Systems* (Oxford, Oxford University Press, 1998), p. 333
13. R. Hartley, A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd edn. (Cambridge University Press, UK, 2003)
14. J. Zhang, X. Zhang, Strict geometric model based on affine transformation for remote sensing image with high resolution. *Int. Arch. Photogramm. Remote Sens.* **34**(B3), 309–312 (2003)
15. Y. Li, H. Wu, A clustering method based on k-means algorithm. *Phys. Proc.* 1104–1109 (2012). <https://doi.org/10.1016/J.PHPRO.2012.03.206>
16. L. Morissette, S. Chartier, The k-means technique: general considerations and implementation in Mathematica. *Tutorials Quant. Methods Psychol.* **9**(1), 15–24 (2013). <https://doi.org/10.20982/tqmp.09.1.p015>
17. R. Gonzalez, R.E. Woods, *Digital Image Processing*, 2nd edn. (Prentice Hall, Upper Saddle River, 2005)
18. G. Hamerly, C. Elkan, Alternatives to the k-means algorithm that find better clustering's, in *11th International Conference on Information and Knowledge Management* (McLean, Virginia, 2002), pp. 600–607
19. Y. Yong, Image segmentation based on fuzzy clustering with neighborhood information. *Optica Applicata* **XXXIX** (2009)
20. R. Shumway, D. Stoffer, *Time Series Analysis and its Applications: with R Examples*, 3rd edn. (Springer, 2010)
21. <https://innoter.com/sputniki/worldview-3/>

Face Recognition in Different Light Conditions



Waseem Rana, Ravikant Pandey, and Jaspreet Kaur

Abstract Facial biometrics continues to be the preferred biometric bench-mark even though other human body signatures are also used. But there are many problems with face recognition. A slight change in the image, which may be due to illumination, or to the environment, can drastically affect the results and accuracy. These changes may not even be noticeable to the human eye, but can affect the face recognition process. There are some advantages, such as contactless, easy to use and, keeping in mind the current situation, reduces the risk of getting infected by touching buttons or by simply waiting in the queue for a long time. In this paper, an effort is made to review different kinds of face recognition methods comprehensively. PCA, LDA, SVM, and various hybrid techniques are used for the face recognition approach. This review examines all of the aforementioned techniques as well as various parameters that pose challenges to face recognition, such as illumination, pose variations, and facial expression.

Keywords Face recognition · LDA · SVM · PCA · Iris scan · SOM

1 Introduction

Face recognition has become the most popular biometric authentication technique in past the few years, and today most devices come with face recognition implemented in them by the manufacturer. Face recognition is not only reliable but also requires no extra effort from the user by simply looking at the screen and the device recognizing the face [1].

Face recognition can be divided into two parts: detection and identification. Face detection means to detect faces in the image or frame [2]. There can be more than one face in the frame, so the first step is to detect all the faces in the frame [2]. For

W. Rana (✉) · R. Pandey · J. Kaur

Department of Computer Science and Engineering, JSS Academy of Technical Education, Noida, India

J. Kaur

e-mail: jaspreetkaur@jssaten.ac.in

detecting faces, a system uses a Haar cascade classifier, it is machine learning-based approach were lot of negative and positive images as shown in Fig. 1 are fed in to the system. Negative images, as shown in Fig. 2 are those images that do not have a face in them and positive images are those which have faces in them. There are basically three steps to face recognition: acquisition in which faces are detected and descriptions of those face from different sites; normalization is segmentation, arrangement, and consistency of official descriptions; recognition of faces which were acquired will be compared with the facial descriptions of faces in the database that are already present.

Fig. 1 Negative image (without face)



Fig. 2 Positive images (with faces)



Face recognition is a very successful application for image understanding and analysis, and it has received widespread acclaim, particularly since it was implemented on smart phones in recent years. There may be two reasons for this gradient: The first is the wide scope of business and law authorization applications, and the second is the accessibility of doable advancements following 30 years of examination [3]. There are a variety of biometric features that can and are used for recognition purposes, such as iris, fingerprints, and speech, but all of the other methods require human intervention, and active operations will necessitate additional hardware for scanning iris [4]. Despite the fact that current machine acknowledgment frameworks have arrived at a significant degree of maturity, their accuracy is bounded by the conditions imposed by other real conditions [5]. Let's take an example of recognition of face images captured in a lightning or open environment which changes the illumination or pose, which remains a great, baffling challenge. All in all, current systems are still a long way from the capability of human perception.

2 Literature Survey

In paper [6], face recognition is performed on still images or when an image is captured from a frame in a video, which is a 2D image, so we should use 2D image matching and recognition; distinct images may not be available in most commercial and law enforcement applications. To tackle the problems caused by 2D face recognition, the hyperspectral method was introduced by Di and Lei [7], face recognition can also be based on other perception modalities such as infrared and sketching images studied by Buddharaju et al. [1] is also possible. Despite the fact that this is a misrepresentation of the real acknowledgment issue of 3D [8] items dependent on 2D pictures, we have more focus on this problem caused by 2D, and there are some issues that may be caused because of this 3D to 2D transformation, and we will look into them in other sections.

In paper [2], the author presents a review of face recognition. New calculations that must be developed utilizing half-breed strategies for delicate figuring apparatuses, for example, ANN, SVM, and so on, yield better execution. An attempt is made to audit a wide range of techniques utilized for face acknowledgment exhaustively. This includes PCA, the Institute of Computer Accountants, LDA, Gabor wavelet delicate processing instruments like the artificial neural network for recognition, and various strategy cross-breeds [7]. This audit investigates all of the techniques with boundaries that cause problems with face recognition, such as enlightenment, presenting variety, and facial expressions.

In paper [9], the author describes how the new emerging technologies like VR and AR can be used for retail business. The user, instead of buying physical goods can generate a virtual version of that product and the person can even get the virtual experience of the product. But the problem with the technology is that it does not provide security from the exploiters, and the initial cost of implementing VR and AR technology is significantly high that is not bearable for a small shop owner.

To increase the accuracy of the recognition process, we can take multiple biometrics of the user, like iris, fingerprint, palm print, and face [10]. These can be processed with the help of CNN and deep learning and the accuracy of the system will drastically increase up to 94%.

The improvement in recognition and detection technology can also be used by satellite images for viewing SAR images [8]. Even a small change in the image can easily be detected with the help of machine learning and deep learning. This will help people from disasters and nature changes. This will give people more time to prepare in case of any disaster.

After more than 30 years of innovative work, essential 2D face recognition has reached the development level in [11], and numerous business frameworks are available for various applications. Early research into face recognition was fundamentally focused on the practical question of whether machine detection of faces was feasible. Tests were generally done utilizing datasets comprising of as many as pictures. Huge improvements were seen throughout the mid-1990s, with numerous strategies proposed and tried on datasets comprising upwards of 100 people.

In paper [5], the author presents a structured, coordinated approach to dealing with the posture issue in face confirmation, in which neither one orchestrates the face picture nor creates a framework for the face picture. Layout coordination is carried out by utilizing a restless-formed portrayal of face pictures. The restlessness-based portrayal of face pictures is figured out utilizing (1D) [8] preparing pictures. It checks the personality of an individual by utilizing a score acquired from pattern matching.

3 Basic of Face Recognition

The whole process consists of face detection and face recognition. In face detection, a face image is provided as input which is then converted into a machine-understandable form and outputs a binary value of yes or no [6], indicating whether there are any faces present within the image. If the face is present, then it returns a bounding box around it. With the acquired image, facial feature detection—eyes, nose, lips, ears, etc., is performed [10]. It is a difficult task because of variations in age, skin color, and facial expressions. Faces have variable, complex structures that can change as a result of factors such as exercise, airflow, and atmospheric pressure [4]. Face detection should work under any lighting conditions, background objects, noise, pose presence, even if the structural components are absent, facial appearance, obstruction, and image orientation [1].

In face recognition process, the image is taken as input as shown in Fig. 3 and compared with the dataset. Face recognition techniques that are well-known include geometric face (feature based) and photometric (perspective based).

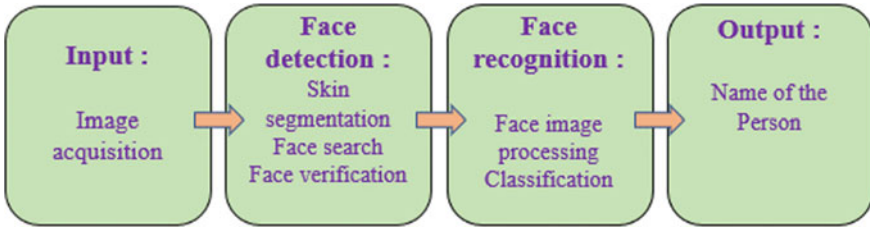


Fig. 3 Face recognition approach

3.1 Geometric (Feature Based)

In the feature-based approach, we take local features like eyes and noses, as shown in Fig. 4 that are segmented based on the geometrical relationship between facial features [8] which are used as input data to ease the function of face recognition. Lin et al. [7] described another method for face recognition by considering individuals and taking 10 images that are stored in the dataset.

The problem with geometric-based features is that they tend to change with time, specifically when talking about young people and children, whose facial geometry may change with time as they grow up. This issue may also arise if two people collide in an accident [11]. With all these problems, the geometric-based approach is considered as all these problems can be overcome by adding a few more checks. If the scan for face recognition is not sufficient, then other specifications of the face may be used.

Fig. 4 Facial regions

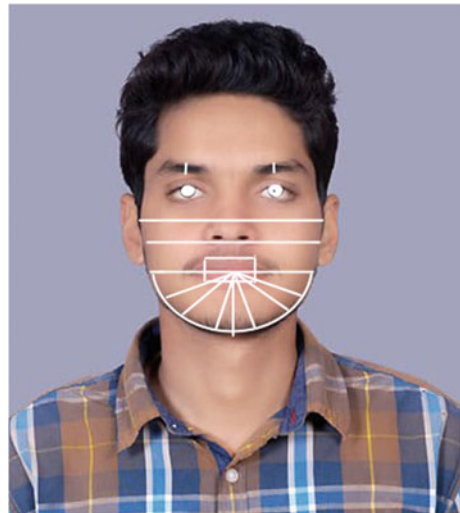


Fig. 5 Facial regions
(high-level template)



3.2 Photometric (View Based)

In the photometric-based approach, we use the entire face as input, as shown in Fig. 5 to the face detection system, which performs face recognition. It is used to regenerate the form of a face (object) from a variety of pictures that were taken under different lighting conditions.

4 Different Techniques for Face Recognition

Popular recognition algorithms include:

4.1 Principal Component Analysis Using Eigenfaces (PCA)

Eigenface is one of the most thoroughly researched approaches to recognizing faces. It was proposed by Cutler [12] by establishing the Eigen method, which was directly developed by Pentland [3]. A face image is approximately reconstructed by projecting the face sample picture onto the Eigen picture, resulting in a minute assembly of weights for each and every face with a typical face image (Eigen picture) [4]. The measures that describe each face are acquired. Eigenface appears to be a fast, very straightforward, and very feasible method. It does not, however, produce a uniform for minor changes in scale or lighting conditions [4]. Using the principal component analysis approach as shown in Fig. 6, it has been demonstrated that the performance

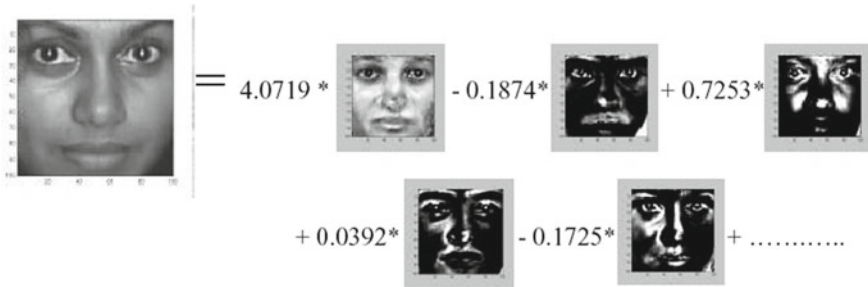


Fig. 6 Face image as weighted sum of eigenfaces

is equivalent to using ear pictures, face matching, or multiple modal recognition, resulting in a statistically significant performance.

4.2 Graph Matching

Graph matching is one of the approaches to face recognition that can be presented using a dynamic link structure and is useful for distortion invariant visual perception because it employs elastic graph matching and searches for the closest stored graph [5]. Different frequency values are assigned to different sub bands [7]. Memorized faces are represented by sparse graphs. The matching process is inefficient, which takes about 25 s to match them with 89 stored images on a parallel machine.

The face bunch graph (FGB) is designed to cover all of the variations in the face. It gathers data from a number of face graphs and combines them. Nodes are labeled and labeled with a set of jets.

The dataset is being used is created by the user as shown in Fig. 7, and images in that dataset are also captured by the user. It is preferred that the user captures the images from the same camera as being used for recognition.

4.3 Support Vector Machine (SVM)

A support vector machine (SVM) determines the hyperplane that is capable of separating the biggest possible section of points of a similar class on an equivalent side, while escalating the gap between classes in the hyper plane [6]. A feature graph is built that maps face images to feature space points [3]. Face images of the same person are mapped in close proximity to each other, with results. As a result, VMS learns the boundaries between different people’s face images.



Fig. 7 Training dataset used in the recognition process

4.4 Artificial Neural Network (ANN)

For face recognition, multiple layer perceptions and convolution neural networks have been applied as shown in Fig. 8. A hybrid neural network is also proposed

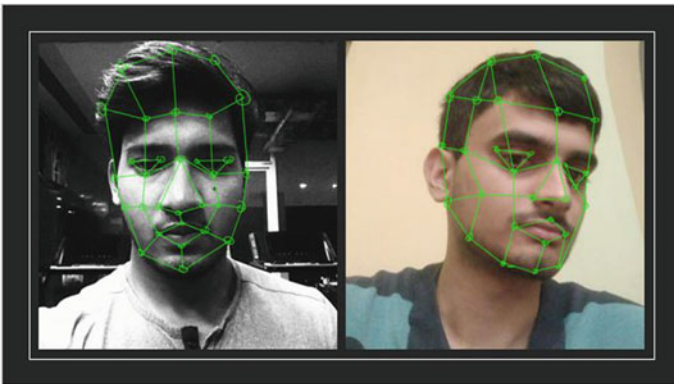


Fig. 8 Graph laid on a face

Fig. 9 Image data mapped onto feature space



which has the capability to combine local image sampling, a self-organizing map (SOM), which is a form of neural network, and a convolution neural network [2]. The SOM gives dimensionality depleting and uniform to compact changes in the pictures identified by quantization of the picture samples into a topological space where inputs that were taken from the original space and where nearby inputs are also nearby in the output space [2]. The convolution network is capable of extracting consecutively sizeable features in a pecking order set of layers and provides partial invariance to translation, rotation, scale, and deformation. It gives an accuracy of about 96.2% correct recognition on ORL database, where there are 400 images of 40 different people. This classification is efficient and take about 0.5 s or less, but training them takes a lot of time, as long as 4 h as shown in Fig. 9.

5 Conclusion

The paper has attempted to check an extensive variety of papers to cover the recent improvements in the area of face recognition. The task of distinguishing a previously recognized face from a known or obscure face is known as face detection. Face detection entails categorizing image windows into two types of ratings. Students are programmed to recognize it. It has been more than 30 years of analysis and evolution. Basic 2D face recognition has reached a mature level in many industrial systems, as shown in Fig. 10, but it can still be improved by using hybrid methods or more than one method to recognize the image, which will also lead to better performance.

The recognition system for identifying people still has a lot of issues. These issues may arise as a result of system flaws in face recognition as shown in Figs. 11 and 12, such as background, camera distortion, storage, inappropriate methods, and so on, as well as network issues caused by environmental conditions.



Fig. 10 Showing edges of the image

```
import cv2
import os

dir = ".\\"
clf = cv2.CascadeClassifier('haarcascade_frontalface_defe

def show_bbx(fname):
    img = cv2.imread(fname)
    bbx = clf.detectMultiScale(img,1.5)
    for (a,b,c,d) in bbx:
        cv2.rectangle(img, (a,b), (a+c,b+d), (255,255,0), 5)

cv2.namedWindow("Image", cv2.WINDOW_NORMAL)
cv2.imshow("Image",img)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

Fig. 11 Detect face using Haar cascade

Fig. 12 Face detection—display bounding box



Acknowledgements I would like to thank to my teachers and Friends who helped me in completing this project. I would like to specially thanks to My Guide (Ms. Jaspreet Kaur) whose efforts, guidance and direction gives me a motivation to complete this manuscript.

References

1. P. Buddharaju, I.T. Pavlidis, M. Bazakos, Physiology-based face recognition in the thermal infrared spectrum, in *IEEE Conference on Advanced Video and Signal Based Surveillance* (2007), pp. 354–359
2. S.A. Robila, Toward hyperspectral face recognition [6812-32], in *Proceedings—SPIE The International Society For Optical Engineering*, vol. 6812 (International Society for Optical Engineering, 1999, 2008), p. 6812
3. M. Turk, A. Pentland, Eigenfaces for recognition. *J. Cogn. Neurosci.* **3**(1), 71–86 (1991)
4. M. Abdullah, M. Wazzan, S. Bo-Saeed, Optimizing face recognition using PCA (2012), arXiv preprint [arXiv:1206.1515](https://arxiv.org/abs/1206.1515)
5. X. Lu, Yagnik, A literature survey on face recognition techniques. *Int. J. Comput. Trends Technol.* **4**(5) (2013)
6. F.J. Prokoski, R.B. Riedel, J.S. Coffin, Identification of individuals by means of facial thermography. *Int. Carnahan Conf. Secur. Technol.* 120–125 (1992)
7. X. Maldague, *Theory and Practice of Infrared Technology for Nondestructive Testing* (Wiley, New York, 2001)
8. R. Dhaya, Hybrid machine learning approach to detect the changes in SAR images for salvation of spectral constriction problem. *J. Innovative Image Process. (JIIP)* **3**(02), 118–130 (2021)
9. S.T. Kumar, Study of retail applications with virtual and augmented reality technologies. *J. Innovative Image Process. (JIIP)* **3**(02), 144–156 (2021)
10. T. Vijayakumar, Synthesis of palm print in feature fusion techniques for multimodal biometric recognition system online signature. *J. Innovative Image Process. (JIIP)* **3**(02), 131–143 (2021)
11. A. Sungheetha, R. Sharma, 3D image processing using machine learning based input processing for man–machine interaction. *J. Innovative Image Process. (JIIP)* **3**(01), 1–6 (2021)

12. G. Ranganathan, Real life human movement realization in multimodal group communication using depth map information and machine learning. *J. Innovative Image Process. (JIIP)* **2**(02), 93–101 (2020)

Music Genre Transfer Using TransGAN



Sandeep Kumar, Jerin Verghese, and Ritesh Dutta

Abstract There is currently no cutting-edge methodology for genre transfer of music as creating music is challenging. CNN-based GANs are used to generate pictures, whereas transformers are used to generate text. Since music does not fall into both categories, none can produce the greatest effects on its own. To address this issue, this research work employs TransGAN, a paradigm that employs transformers (rather than CNN) in the design of GANs. We believe that the proposed approach may be used to provide cutting-edge performance in music generation. A dataset of 64×64 and 128×128 pixel mel spectrogram images was used, and the model was able to transfer genre and detect musical patterns.

Keywords Transformer · GAN · TransGAN · Genre transfer · Mel spectrogram

1 Introduction

Traditionally, GANs or generative adversarial networks have 2 CNN models called generator and discriminator. The generator tries to generate an image similar to the input image, and the discriminator identifies which image is the real image and which image is made by the generator. The output from the discriminator is then used by the generator to generate a more accurate image which the discriminator then compares with the original image. This adversarial architecture is used to generate photorealistic photographs of objects, scenes, and people that even humans cannot tell are fake.

Transformers were originally made for NLP tasks where their multi-head self-attention mechanism enables words in a sentence to be processed parallelly and find long-term correlation between them, which is more suited to modern GPUs. But with the introduction of vision transformer (ViT) images can also be processed by splitting an image into 16×16 “words” that can be processed parallelly by a

S. Kumar · J. Verghese · R. Dutta (✉)

Department of Computer Science, Maharaja Surajmal Institute of Technology, New Delhi, India

S. Kumar

e-mail: sandeep.jaglan@msit.in

traditional transformer. Based on this ability of transformers, we have recreated the TransGAN model and fine-tuned it to be able to understand and differentiate different types of music by their spectrograms and to transfer one genre of music to another.

2 Related Work

1. Perone et al. [1]: This paper deals with word embeddings. It shows how a method that uses bag-of-words with embedding from language models (ELM0) for deep entailment word embeddings proved to get better results in linguistic probing and downstream tasks when compared with sentence encoders trained on natural language relation datasets.
2. Goodfellow et al. [2]: In this paper, they proposed a framework for estimating generative models via an adversarial process, where two CNN models are trained simultaneously: a generative model G that tries to replicate the data and a discriminative model D that tries to distinguish between the actual data and the data generated by G. G that tries to “fool” D while D that tries to identify data generated by G.
3. Chang et al. [3]: The paper proposes a simple GAN architecture that consists of a memory-efficient generator based on transformers that increases feature resolution while decreasing embedding dimension. The model achieves promising performance when compared to modern convolution-based GANs architecture.
4. Vaswani et al. [4]: This paper introduces a simple architecture called transformer, based completely on attention mechanisms, without using convolutions or recurrent model architecture. They show that the transformer also acts as a universal model architecture when trained on both small and large English entailment data.
5. Lu et al. [5]: In this paper, they use a model called FPT or frozen pretrained transformer that uses a pretrained transformer (on natural language) which can be used for various sequence classification tasks with negligible fine-tuning—in particular not fine-tuning its self-attention layer. They study fine-tuning on a variety of tasks like computer vision, statistical computation, and protein fold prediction.
6. Wang et al. [6]: In this paper, they use GAN-based models employing several generators and some form of cycle consistency loss as they were state of the art for image generation and transfer at that time. They use a similar cyclic model for symbolic music and display the practicality of such a model for music genre transfer.
7. Jakob et al. [7]: The paper proposes a method which uses a single 2D convolutional neural network in both sequences. Every layer in the network re-codes input tokens with respect to the output sequence produced. This results in the network having properties similar to transformer. The model has fewer parameters and gives better results than the encoder–decoder format.

8. Mogren [8]: Generative adversarial networks are a way of training deep generative neural networks efficiently. The paper proposes a generative adversarial model which uses continuous sequential data, trained on a collection of classical music. They find that it generates music that sounds better as the model is trained. They also report statistics about the generated music.
9. Chou et al. [9]: This paper proposes using CNNs for generating melody one bar at a time. Other than using a generator, they also use a discriminator for learning the distributions of melodies, thus making it a Generative Adversarial Network (GAN). Moreover, they investigate a novel conditional mechanism to make use of the available prior knowledge

3 TransGAN

TransGAN is a model proposed by Yifan Jiang, Shiyu Chang, and Zhangyang Wang. TransGAN uses a pure transformer network-based architecture to train a GAN for image synthesis. Adding a self-supervised co-training task to a GAN has been found to stabilize the GAN training. Co-training tasks can be rotation prediction or anything. Similarly, in TransGAN training, an auxiliary co-training task of super-resolution is coupled with the GAN loss. The generator loss is added with an auxiliary term $\lambda * L$ (super-resolution), where L (super-resolution) is the MSE of super-resolution image obtained at the end stage and low-resolution image at any of the middle stage and λ is set to 50 empirically. One of the strong motivations of the researchers is to join task pipelines and make them comprehensible, so one general suite of models could be extensively reused by many applications (Fig. 1).

The TransGAN model basically uses sets of transformer encoder blocks as computational engines for both the generator and discriminator. The generator usually takes noise as input so upscales the output to match the discriminator input shape. At the end of the generator, there is a convolution layer to unflatten the output so the output has the desired image shape. The discriminator has a classification head at the end to be able to critique the input into two or more classes. Both the generator and discriminator have an embedding layer as their first layers.

4 Proposed Work

4.1 Data

We used the GTZAN music genre classification dataset from Kaggle, which has directories of dataset of 100 music files each of different genres. The data was then fed into a script which converted it into a mel spectrograms which then were saved as .tiff files. We chose to use the.tiff format because when compared to .png and .jpeg, .tiff files are relatively lossless and also support storing of two-dimensional

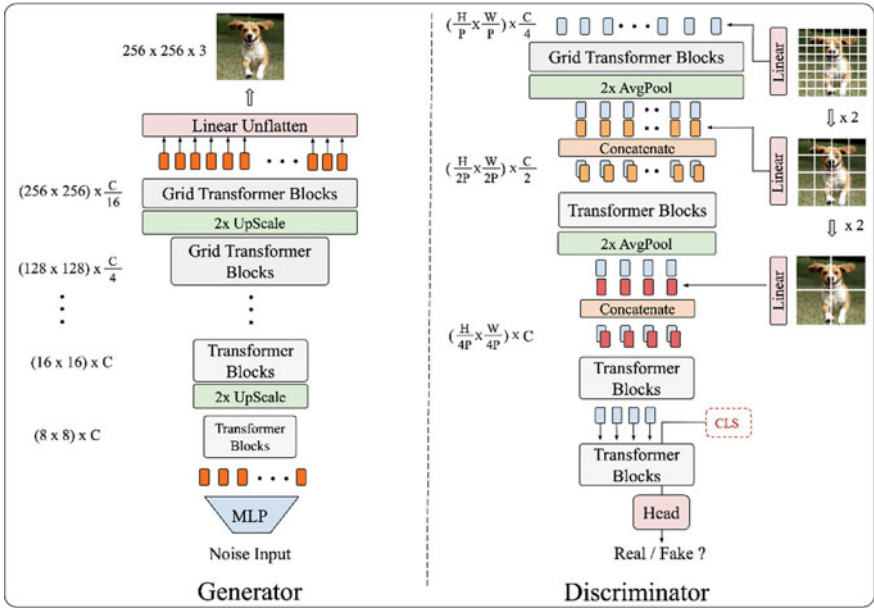
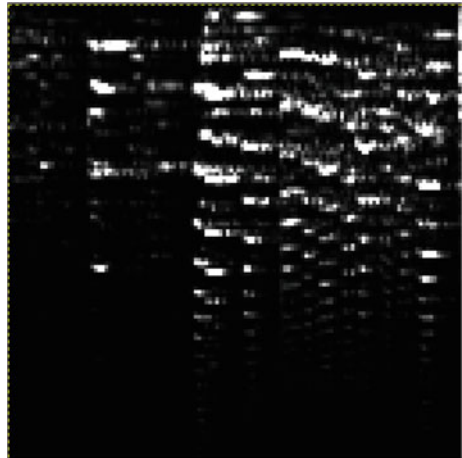


Fig. 1 TransGAN model architecture

data. While creating spectrograms, we chose to divide a music file into several equal samples, and this helped increase the size of data for training (Fig. 2).

Fig. 2 Numpy array of mel spectrogram in .tiff format



4.2 Model

We have built the TransGAN model using PyTorch based on the implementation provided in the TransGAN paper. For both generator and discriminator, we have used the encoder part of the transformer in layers as multiple blocks. We have tried to make the model class as loosely coupled as possible by making a separate class for providing arguments to the model. This has allowed us to experiment with different arguments without making major changes to the model itself.

Our model had a depth of seven encoder layers in the discriminator and eight encoder layers in the generator. In the model, we used the following hyperparameters after experimenting with different values,

- `bottom_width = 32` (Bottom width of the linear layers)
- `gf_dim = 64` (Dimensions of the generator embedding layers)
- `df_dim = 128` (Dimensions of the discriminator embedding layers)
- `patch_size = 3` (This was the dimension of a patch convolutional embedding layer used in the discriminator)
- `g_lr = 0.00001` (Learning rate for generator)
- `d_lr = 0.00001` (Learning rate for discriminator)
- `alpha = 0.7` (Initial value of alpha, used as gradient penalty)
- `wd = 1e - 3` (Value of weight decay, a form of regularization)
- `n_critic = 5` (The number of times discriminator trained for every time generator is trained)
- `fade_in = 0.025` (The value of alpha increases by `fade_in` till it becomes equal to 1)

4.3 Training and Loss

For training the model, we have used the AdamW optimizer along with LR scheduling and weight decay. We also adopted the approach of training the discriminator more than the generator. We trained the model on two sets of data, one with 64×64 dimensional spectrograms and 128×128 dimensional spectrograms to study the importance of frequency and time data in genre recognition and generation. After this, we plotted the training verbose as a graph to interpret the results. We also plotted the resultant output as a spectrogram which allowed us to understand the extent of accuracy.

We used the Wasserstein loss function used in WGANs as the loss function. The Wasserstein loss function for loss is given by the following formulas.

$$\text{Discriminator Loss} = d(x_1) \tag{1}$$

$$\text{Generator Loss} = d(g(x_2)) \tag{2}$$

Here, $d(x)$ represents the forward pass output of the discriminator, $g(x)$ represents the forward pass output of the generator, x_1 is the desired output genre data, and x_2 is the input genre data.

We chose Wasserstein loss because models with Wasserstein tend to get stuck less and also they are more resistant to vanishing and exploding gradients. The whole training was done on a Collab runtime with CUDA-based NVIDIA GPUs for faster computation.

5 Results

We trained our TransGAN model on the generated music spectrograms, first on 64×64 spectrograms and then on 128×128 spectrograms. With 64×64 , we got around 1800 data points to train on, and with 128×128 , we got around 990 data points. We were successfully able to train the model, and we got an output for both the cases. We obtained the following results.

5.1 Output

From the above outputs, we can make out that the output of 128×128 dimension spectrograms is much more defined and has much more intricate patterns when compared to 64×64 dimension spectrograms. This is because 128×128 dimension spectrograms have 128 mels to train on which means it has more frequency points to distinguish a song (Figs. 3 and 4).

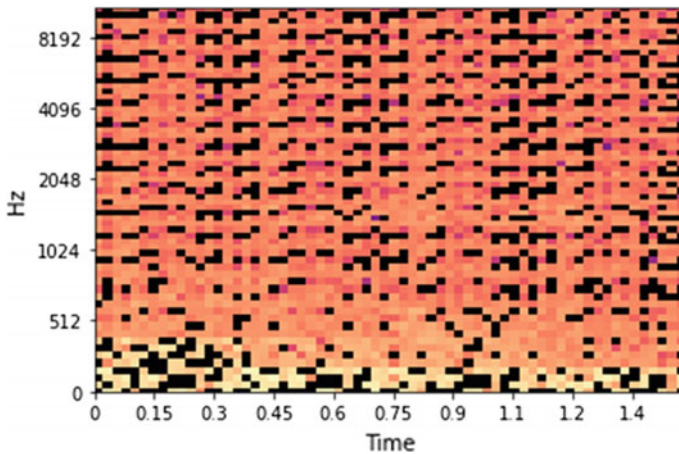


Fig. 3 Output for 64×64 spectrogram

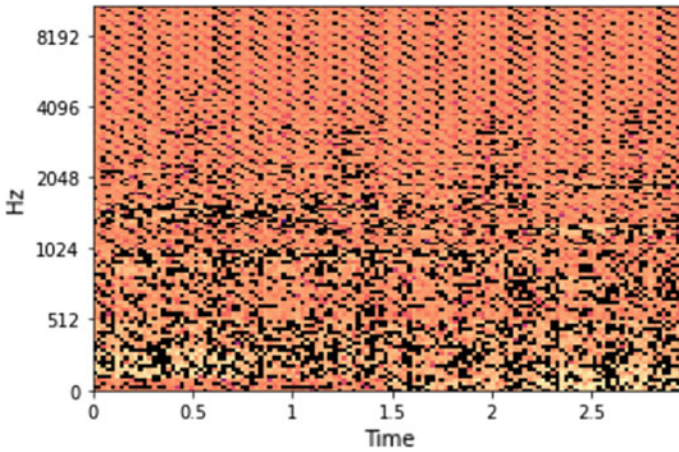


Fig. 4 Output for 128×128 spectrogram

5.2 Deviation

Even though we were able to train a model which was able to detect and filter patterns from a spectrogram, the results we got did not produce a human distinguishable result. The reason for this is even a spectrogram with 128×128 dimension does not have enough frequency and time data to differentiate a music's genre.

The following training graph helps to understand this (Fig. 5).

From the above training verbose and the graph, we can see that after an initial performance gain of both the generator and discriminator, both of them stopped

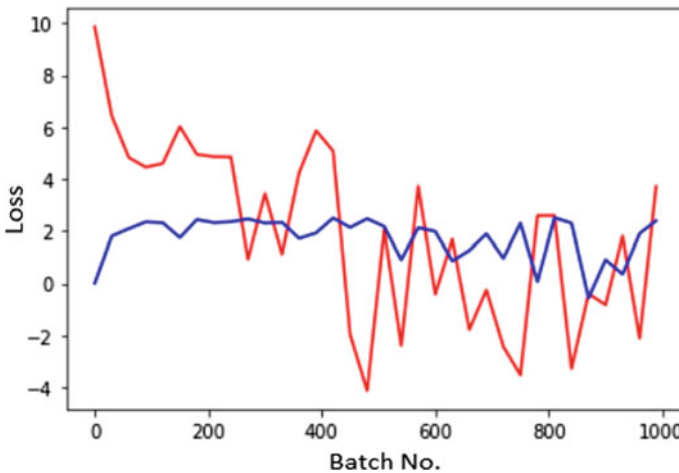


Fig. 5 Training loss (Wasserstein), blue = generator, red = discriminator

showing any trend. This is because the learning at this point has become saturated as the discriminator cannot make out any more difference between the rock and jazz images.

6 Conclusion

From the results obtained, we have come to the conclusion that a genre transfer of music is possible using its spectrograms, but for any human distinguishable results, we need lossless music with high frequency range and high dimension spectrograms. A higher processing power is also needed for a better result. Some optimizations can also be made to the transformer architecture by reducing its time complexity and efficiency by using some models like Linformer and Longformer. This will reduce the hardware required to train the model, and a higher resolution spectrogram can be used.

References

1. C.S. Perone, R. Silveira, T.S. Paula, *Evaluation of Sentence Embeddings in Downstream and Linguistic Probing Tasks*. [arXiv:1806.06259](#) (2018)
2. I.J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, *Generative Adversarial Networks*. [arXiv:1406.2661](#) (2014)
3. Y. Jiang, S. Chang, Z. Wang, *TransGAN: Two Transformers Can Make One Strong GAN*. [arXiv:2102.07074](#) (2021)
4. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need. *Adv. Neural Inf. Process. Syst.* 5998–6008 (2017)
5. K. Lu, A. Grover, P. Abbeel, I. Mordatch, *Pretrained Transformers as Universal Computation Engines*. *arXiv preprint* [arXiv:2103.05247](#) (2021)
6. G. Brunner, Y. Wang, R. Wattenhofer, S. Zhao, Symbolic music genre transfer with cyclegan, in *2018 IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI)* (IEEE, 2018), pp. 786–793
7. Y.S. Huang, Y.H. Yang, Pop music transformer: beat-based modeling and generation of expressive pop piano compositions, in *Proceedings of the 28th ACM International Conference on Multimedia*, pp. 1180–1188, 2020
8. E. Maha, B. Laurent, V. Jakob, *Pervasive Attention: 2D Convolutional Neural Networks for Sequence-to-Sequence Prediction*. [arXiv:1808.03867](#)
9. O. Mogren, *C-RNN-GAN: Continuous Recurrent Neural Networks with Adversarial Training*. [arXiv:1611.09904](#) (2016)

Diskless Booting: A Hybrid Computing Technology



S. Suriya Prasath, Shwetha Ravikumar, and Anindita Khade

Abstract Any IT organization needs an infrastructure consisting mainly of computers with different capacities of hard drives and other components. The operation and maintenance costs of such decentralized networks are high; hence, troubleshooting and security can be difficult. Educational institutions are also forced to cut budgets and provide better solutions for student's learning and development. The main purpose of this paper is to analyze and develop a system based on the concept of diskless booting a hybrid computing, in which the diskless client PC is devoid of the secondary storage device responsible for storing the necessary boot-oriented and operating system files. This paper also provides critical analysis of how inexpensive, energy-efficient, and fully managed diskless clients have proven to be a better approach which enhances speed and reduces cost, keeping in mind the existing infrastructure.

Keywords Dynamic host configuration protocol · Trivial file transfer protocol · Network file system · Firewall · Fail2ban · Open-source antivirus · Network booting · Install OS to the base machine bypassing traditional method · Cockpit · Netdata

1 Introduction

Diskless client server technology plays a critical role in streamlining the administrative operations, guaranteeing a high level of security and simplifying the data administration. This efficient approach is widely embraced by mass educational institutes such as schools and colleges, and other IT organizations including

S. Suriya Prasath (✉) · S. Ravikumar · A. Khade
SIES GST, University of Mumbai, Navi Mumbai, India
e-mail: suriya.prasath17@siesgst.ac.in

S. Ravikumar
e-mail: shwetha.ravi17@siesgst.ac.in

A. Khade
e-mail: anindita.khade@siesgst.ac.in

government offices, businesses, and many other places where multiple computer systems/workstations are connected through a network, due to added benefits such as cheaper cost, simplicity of maintenance, and higher productivity.

The typical method of constructing a PC cluster involves manually configuring each node for the better performance, but this approach is fraught with issues, such as huge number of machine maintenance and installation. Furthermore, these PC clusters consume more energy compared to the energy-efficient systems (diskless technology/hybrid technology), escalating their operational costs. Additionally, the PC models at educational institutes and other IT organizations have a high overall cost of ownership [1]. Software licensing, expenditures for virus protection, support, and upgrades for PC's are pricy; a license must be purchased for each machine, and the application or upgrades must be installed by the IT admin/staff one at a time on each unit.

In contrast to the foregoing, the expenses associated with the hybrid computing technology is relatively cheap. Diskless clients utilize the operating system and the software application by accessing from a nearby server rather than storing on each computer's secondary storage device or a hard drive. Applications stored on the server can be updated for use by all connected computers at once at this central point of control rather than at every computer station. Many of these applications have the added bonus of being open source, which means they are free to use. Further, because the information originates directly from the local storage and are screened by the host, hybrid computing architecture is safer from viruses and bugs than a cloud-based cluster computer network. The data is distributed via a switch employing unshielded twisted pair (UTP) CAT 6 cables as a transmission medium from the host/server workstation [2].

The use of LINUX OS to implement diskless client server technology is greatly recognized. The reason for this is that Linux operating systems are extremely stable and may be used for a variety of high-availability tasks, such as Web and database servers. Linux clusters can also be used to provide high-performance computing resources. While implementing this concept in the college premises, Ubuntu 20.04.2 LTS (Latest LTS release of 2020) was used among several Linux distributions.

One or more server workstations not just provide the bootstrap service but also related network services in a hybrid computing model, and many clients with no secondary storage device or hard disk drive request booting over the network. Thus, before booting, each diskless node can use a floppy disk or a NIC's boot ROM with a simple bootstrap application, or even a NIC's PXE, which sends a broadcast packet to the DHCP server and is then granted with an IP address. After each node has been assigned a valid IP address, it uses the TCP/IP protocol to send a request to the TFTP server for the appropriate boot image and kernel image (with arguments supplied) and commences the booting process [3]. The supplied arguments help the kernel self-configure and mount the supplied directory path through the configured NFS. All of the remaining necessary system files are delivered over the network during the booting procedure. The client workstation is available to use, once the remote file system has been mounted as the root file system and the system start-up has been completed [3]. Each time the client boots, it mounts most of the OS from

the NFS server as read-only and other directory as read–write. Each client has its own read–write directory so that one client cannot affect the other. Also, because the server station will be running on top of highly effective hardware, there will be no performance drop from the server side.

In this paper, LTS version of Ubuntu (Version-20.04.2) is used to develop and test the hybrid computing concept. Proper firewall rules and open-source antivirus were configured appropriately to provide comprehensive security from the external threats. GUI-based client monitoring tools were incorporated to make the administrative tasks easier and to track the resource usage by the client machines. Alongside, supporting the hybrid computing concept, the proposed architecture was capable of installing a fresh OS via the network bypassing the traditional installation method.

2 Terminologies/Backgrounds

2.1 Shell

It is a crucial component of a Linux system that allows users to run commands, then sends signals to the kernel to initiate the system calls, and finally accepts the output from the kernel and passes it on to the user [4]. Shells come in a variety of kinds, and they can be allotted to the users by default, by defining a default shell of login in the `/etc/passwd` file of the system, which is the Linux system’s local user database. There are many shells, but `/bin/bash` is the one which mostly gets used.

2.2 File Attributes and Permissions

These are special permissions given to the files and the directories that control what can be done with them. The most common file permissions are read, write, and execute [4]. Permissions can be assigned to the file or the directory owner, the group, and others, which covers everyone except the assigned owner and group.

2.3 Open-Source Software

Open-source software shares similarities with free software and is part of the broader term **free and open-source software**. Open-source software (OSS) is computer software that is released under a license in which the copyright holder grants users the rights to use, study, change, and distribute the software and its source code to anyone and for any purpose. Open-source software may be developed in a collaborative public manner. When it comes to free software, Richard Stallman wrote the definition,

which is still in use today. It states that any software is free, if the people who obtain a copy of the software have the four freedoms listed below:

- **Freedom 0:** The freedom to run the program for any purpose.
- **Freedom 1:** The freedom to study how the program works and change it to make it do what you wish.
- **Freedom 2:** The freedom to redistribute and make copies so you can help your neighbor.
- **Freedom 3:** The freedom to improve the program, and release your improvements (and modified versions in general) to the public, so that the whole community benefits.

2.4 PXE Protocol

PXE is a network booting protocol that allows network cards to load boot code via the network. This is done using TFTP, so the appropriate files are needed to be included in the tftp directory [5]. The binary file pxelinux.0 from the PxeLinux distribution enables the computers to boot Linux.

2.5 DHCP

When a client node's Netboot process initiates, it is devoid of any information. An IP address and a host name are the initial pieces of information a node needs, which is fulfilled by the dynamic host configuration protocol (DHCP) by acknowledging the nodes with a fixed IP address. This technique eliminates the need to manually configure the network devices on a per-device basis. Furthermore, the DHCP services are available for both IPv4 and IPv6 networks.

2.6 TFTP

Trivial file transfer protocol (TFTP) is a simple file transfer protocol that allows a client to receive or upload files to a remote host. One of its most common use is during the initial stages of node booting from a local area network. Because TFTP is so simple to use, it was chosen for this application. It is a basic file transfer protocol built on top of the UDP/IP protocols and uses the well-known port number 69. Furthermore, this file transfer protocol solely reads and writes files from or to a remote server.

2.7 NFS

Network file system (NFS) is a distributed file system protocol that allows users on client workstations to access files via a computer network in the same way they access local storage. Using the `/etc/exports` configuration file and the `exportfs` command, the server administrator selects what to make available by exporting the names and parameters of the folders. Within the limits allowed, users on the client machine can view and interact with mounted filesystems on the server.

3 Objective/Motivation

The following bullets list the significant factors that influenced the development of this system:

- Considerable cost reductions in the initial capital and implementation.
- The benefits of both thin and fat client technologies are combined in this hybrid computing paradigm.
- Simplifies the troubleshooting process while also fine-tuning administrative chores without jeopardizing security or productivity.
- Because the client machines lack a secondary storage device, scalability is simple to achieve.
- This technology has a high level of user reliance, software uniformity, and one-click program installation.
- Minimizes the power consumption and cooling requirements.

4 Literature Survey

Server-based computing models have been existed since the first mainframe computers were developed. Moreover ‘Dumb’ or ‘Text Based Terminals’ were integrated with these, facilitating the display and the data entry. Gradually, most mainframe computers were replaced with mini-servers, connected to the PC with secondary storage devices, hosting their own operating systems and providing a graphical user interface.

In the early 90’s, Windows 3.x operating system was the first OS to support the network boot (also referred to as NetBoot). This enabled the computer to boot via the network rather than a secondary storage device or a local drive. This booting method was used by centrally managed computers (thin computing machines). Moreover, in the late 1980s/early 1990s, network boot technology was utilized to cut-down the expenses of a disk drive.

In addition to the above, with the development of Linux, the network boot protocol was adopted for the open-source platform and boot manager program such as PxeLinux became available in the mid-90s.

In the late 1990s, the fat client merged with the first terminal computer. Due to its huge dependence on the server and limited computing power, it was also called a thin client. As the demand for graphics applications continued to grow, traditional thin clients were facing huge challenges due to low processing power, lack of graphics support, and complete reliance on the network [2]. The below table indicates the work done by different researchers. By referring to these works, we have identified few research gaps. These observations have led us to our problem definition.

S. No.	Techniques	Author and year of publication	Observation
1	Compatibility of Linux architecture for diskless technology system	Aryanti Aryanti, Ade Silvia Handayani, Ibnu ziad, Ikhthison Mekongga, Farid Jatri Abiyyu. February 2021	WINE is used as an alternative compatibility layer which can support cross-platforms. Gives good quality of service
2	User control on diskless client server platform	B. S. Sonawane, R. R. Deshmukh, S. D Waghmare and Pushpendra Chavan. April 2018	For good fault tolerance and read–write operations diskless client with the RAID 1 mirroring works well
3	Performance of virtual machines using diskfull and diskless compute nodes	Michael Galloway*, Gabriel Loewen, Jeffrey Robinson and Susan Vrbsky. September 2018	Diskless compute nodes using PXE were benchmarked as compared to traditional diskfull compute nodes. Networking performance suffered most on diskless compute nodes hosting stationary and migrating virtual machines
4	Linux-based diskless system using RSYNC algorithm	August Anthony N. Balute, Dennis B. Gonzales, Mateo D. Macalaguing, Caroline J. Aga-ab. December 2016	Utilized LTSP to give cost effective solution for virtualization technology. Features like good interface, fast image syncing, and full restoration has been added
5	Performance comparison of the diskless technology	Kulthida Phanpikhor, Suchart Khummanee, Panida Songram, Chatklaw Jareanpon. May 2013	The structure with link aggression shows good load balance, but when number of clients are increased from 9 to 22, the average time rises with heavy traffic

(continued)

(continued)

S. No.	Techniques	Author and year of publication	Observation
6	Desktop energy consumption. A comparison of thin clients and PCs	Steve Greenberg, Christa Anderson, Wyse Technology Inc.	Study showed that compared to personal computers or thick clients, thin clients are more energy efficient
7	Diskless HPC cluster for parallel and grid computing on fedora	Toro. Victoria, and A. V. Nestor Waldyd. September 2009	Diskless computing clusters were created to run high-performance application using fedora core 7. Parallel computing and grid computing was used to increase power and efficiency
8	A jobs allocation strategy for multiple DRBL diskless linux clusters with condor schedulers	Chao-Tung Yang, Ping-I Chen, Sung-Yi Chen, Hao-Yu Tung. October 2006	This paper is attempted to find out the best cluster environment in computer classroom at school where HPC is used to demonstrate Cluster performance
9	Remote boot of a diskless Linux client for operating system integrity	Allen, Bruce. July 2002	It provides instructions and steps on how to build a Linux-based diskless client with dedicated system storage
10	A measurement study of diskless workstation traffic on an ethernet	Riccardo Gusella. September 1990	The paper shows a way to provide control of client in the network and manage the resources from server end. System having control calling provision over the network and put the call through RPC and complete the call lock mechanism which help to provide control to the specified system

In the past few decades, various methods have been researched and developed to solve problems with thin clients. Therefore, the introduction of Linux-based diskless workstations emerged which may not be new, but the implementation and deployment are new, and it provides a cost-effective, sustainable, energy-efficient, GUI-compatible, and easy-to-maintain hybrid solution [1].

5 Proposed System

5.1 Architecture

The goal of this study is to develop a system that can replace the old system of totally independent individual workstations. The technology is based on the notion of hybrid computing technology, which allows the client stations to operate without the need for secondary storage device (refer Fig. 1). The system so generated will be indistinguishable from a standard setup, and the fact that diskless booting is used will be hidden from the end-user as well. To complete the booting procedure, server workstations are used. Authentication of users, file storage and access, user quota management, security and backup procedures, and so on are all handled by distinct server modules. Depending on the organization’s available resources, the server modules can be on the same station or on distinct stations.

The system administrator will have access to a GUI-based Web application (Cockpit and Netdata), as stated in further section, to make the process of administering the infrastructure easier. This will enable the administrator to accomplish

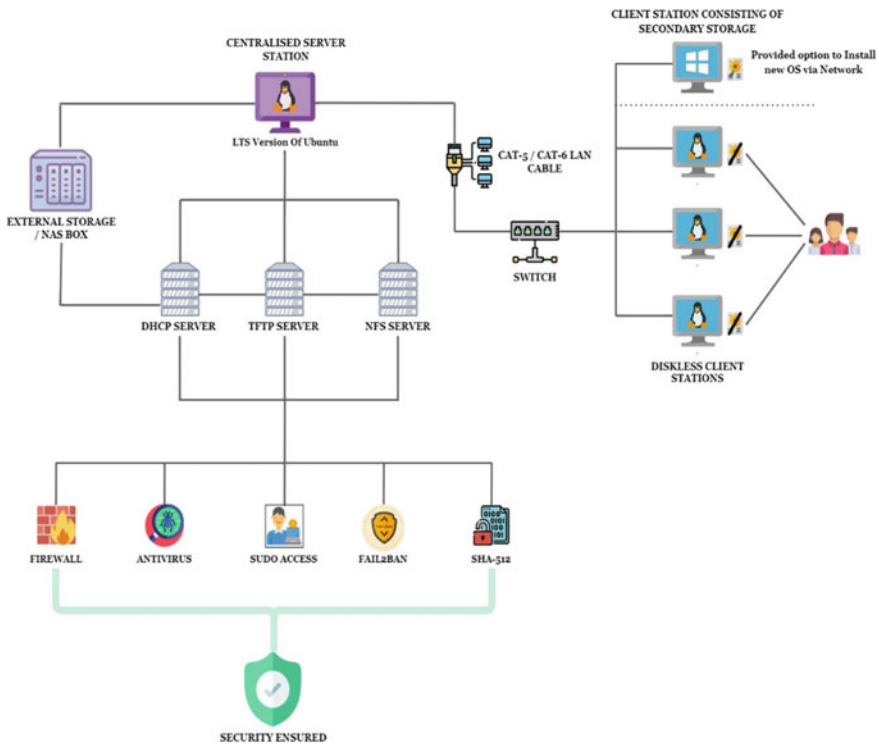


Fig. 1 Architecture

activities with little to no command line experience. Additionally, the system does not put the users under any additional financial strain. Rather, by selectively eliminating the storage medium, infrastructure costs can be substantially decreased while management, security, and disaster recovery can all be improved. In addition, the following measures are applied on the server workstation to protect the entire system from external threats:

- LTS Version of the Ubuntu OS is used, i.e., Ubuntu 20.04.2-LTS.
- Firewall configured with relevant rules. This can differ from one organization to another.
- To protect the system from external attacks, open-source antivirus is utilized.
- Fail2Ban service is used to prevent the brute force attack.
- Sudo access is granted only to specific users.

5.2 Activity Diagram

The flow of the process from initializing the boot to make the diskless systems available for the user's work is depicted in Fig. 2. To identify the type of boot to attempt, the BIOS first analyses the priority of the boot choices. This must be configured so that the network boot is at top of the priority list. When the diskless client PC boots over the network, it asks for an IP address, which is issued dynamically by the DHCP server on the server side. Once the IP address has been confirmed, the diskless boot menu screen appears with the following options:

- Boot the OS via the Network
- Install a fresh Ubuntu OS to the Base Machine (Workstation with secondary storage device) via the network.

If the user chooses the first option, the required boot-oriented files will be supplied via the server's TFTP module. As a result, the actual splash screen appears, and the OS is eventually started. The particular user will now be prompted to input their login credentials. The home screen looks identical to the standard booting process after successful authentication. In the background, all essential folders that the user is authorized to access will be mounted. The NFS server manages this process. Eventually, the diskless client PCs will be ready to use.

However, if the user chooses the second option, the relevant OS installation files will be transferred via the TFTP module, resulting in a successful installation of the fresh OS on the base machine (workstation with secondary storage device), bypassing the time consuming and inefficient traditional method (refer Figs. 3 and 4).

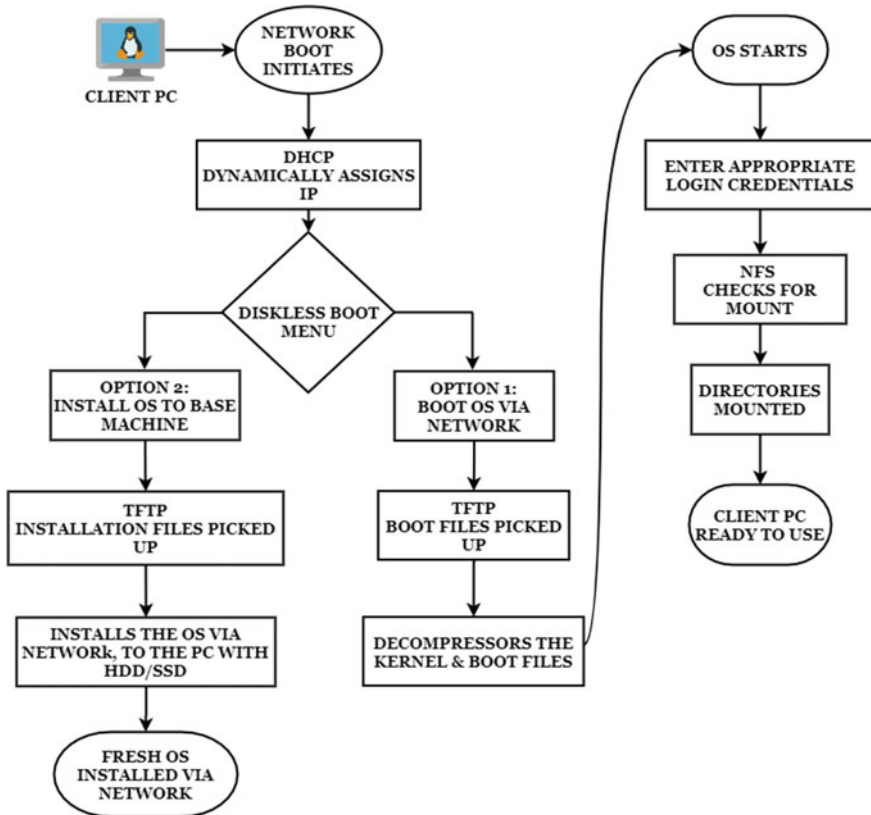


Fig. 2 Activity diagram

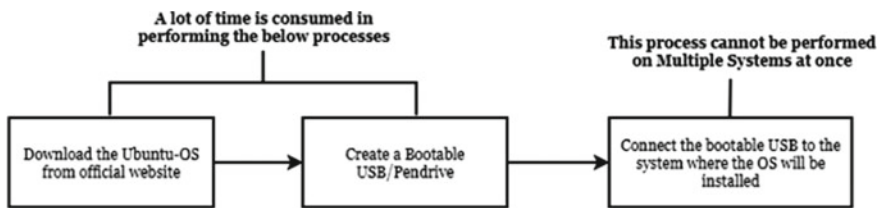


Fig. 3 Traditional installation process

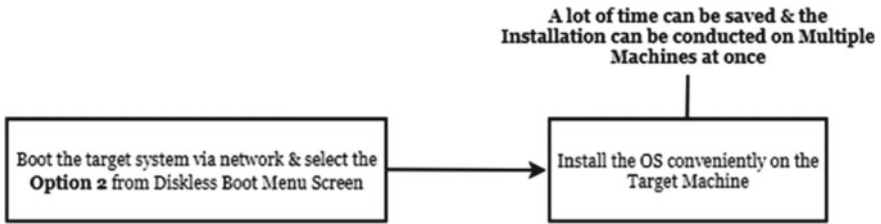


Fig. 4 Bypassing the traditional installation process

6 Implementation

6.1 Operating System Used

Ubuntu OS (20.04.2-LTS) has been utilized, which is the most recent LTS release. Ubuntu is built on Debian's architecture and infrastructure, and comprises Linux Server and Desktop operating system versions. Ubuntu releases new versions every six months, and each one comes with nine months of free support, including security updates, high-impact bug patches, and conservative, significantly helpful low-risk bug fixes. The LTS releases have a five-year support period and are delivered every two years. Long-term support includes updates for new hardware, security patches, and updates to the operating system. The built-in Ubuntu Software, as well as any other APT-based package management tools, provides access to a large number of extra software packages.

By default, Ubuntu OS aspires to be secure. User applications have limited privileges and are unable to harm the operating system or other user-specific files. Moreover, most of the network ports are closed by default to prevent hacking.

6.2 Implemented Diskless Technology Property Devices

This notion of diskless client technology was put into practice in our campus lab, which consisted a total of 48 PCs, one of which functioned as the server and the remaining 47 as clients. The following are the details for the server, client and background network:

- (a) **Server:** For server, Ubuntu 20.04.2-LTS operating system was selected to use since it serves the following:
- It is open source and free.
 - Completely customizable and secure.
 - Better suited for developmental environment.
 - Utilizes less storage space compared to other operating systems.
 - Expert and Naïve users can get conveniently adapted to this OS.

- Can be conveniently updated without restarting, most of the times.

The server system uses Toshiba DT01ACA100 hard disk drive (HDD), having a storage capacity of 1TB and a low-profile form-factor of 3.5-in. In addition to the aforementioned, this HDD features a third-generation SATA interface (SATA-III), that runs at 6.0 Gb/s, engineered with a spin rate of 7200 RPM (revolution per minute). Moreover, this system consisted of 8 GB DDR4 RAM having a clock rate of 800–1600 MHz, with Intel i5-7th Generation Processor and Mesa Intel HD Graphics 630 GPU.

- (b) **Client:** The remaining 47 PCs were used as diskless client PC consisting of Intel i5-7th Generation CPU, 8 GB DDR4 RAM and Mesa Intel HD Graphics 630 GPU along with sound and the network card.
- (c) **Backbone Network:** All the 48 PCs were connected via two 24 port gigabit switches, with a CAT 6 RJ45 LAN Cable, having a network speed of 100 Mbps.

6.3 Admin Monitoring Tools

The diskless client systems were linked with the following open-source tools to enhance admin convenience while performing monitoring activities remotely on any client machines:

- (a) **Cockpit:** It is an open-source framework with a user-friendly Web interface that allows for remote management of the diskless client workstations. Because it is a Web console, it is also simple to use on mobile devices. For system administration, the cockpit package includes a sophisticated and extendable Web console. The cockpit framework can be used to perform the following subset of tasks:
 - Browse and search system logs
 - Upgrade software
 - Manage user accounts
 - Use a terminal on a remote server
 - Configuring firewall as per the needs
 - Writing your own custom modules
- (b) **Netdata:** It is an open-source application that gathers real-time metrics like CPU utilization, disk activity, bandwidth usage, and so on, and then displays them in real-time easy-to-understand graphs. This application is meant to depict any action in the most detailed way possible, allowing the administrator to get a complete picture of what is going on at each client station. In addition to the foregoing, Netdata is a lightweight tool which consumes minimal resources; i.e. about 2% on a single CPU system.

6.4 Results

See Figs. 5, 6, 7, 8 and 9.

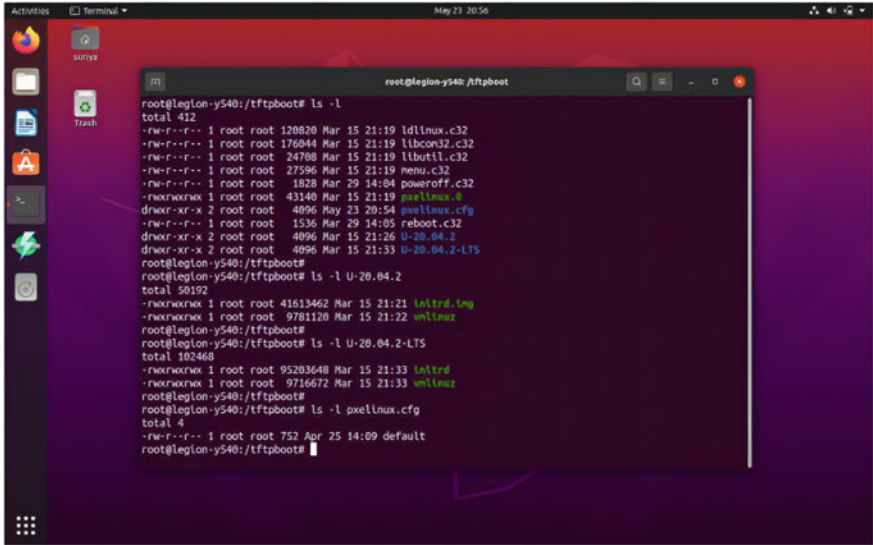


Fig. 5 Tftpboot directory

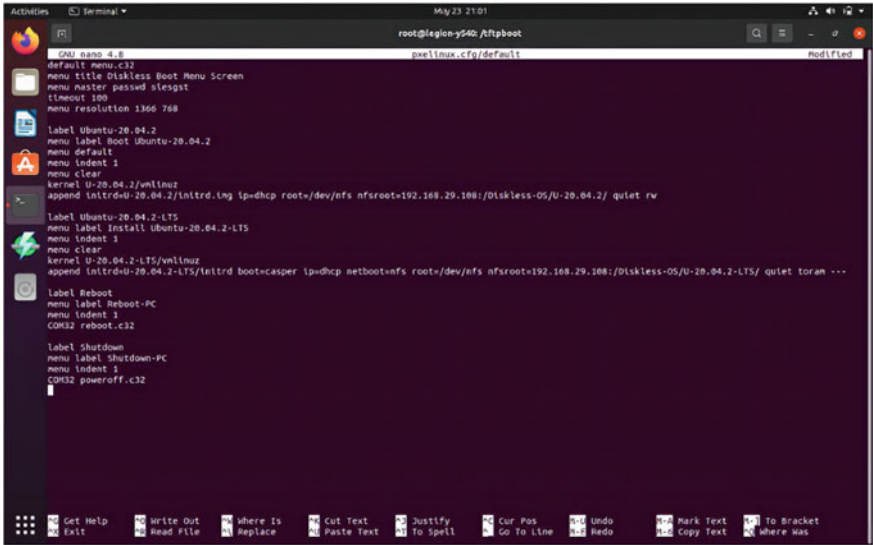


Fig. 6 Default file configuration

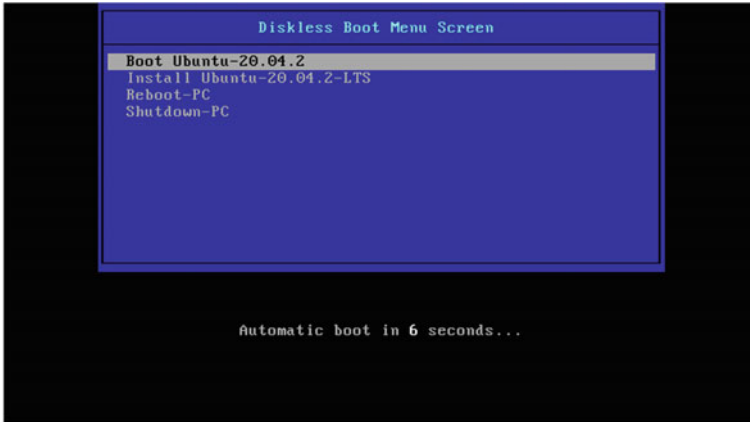


Fig. 7 Diskless boot menu screen

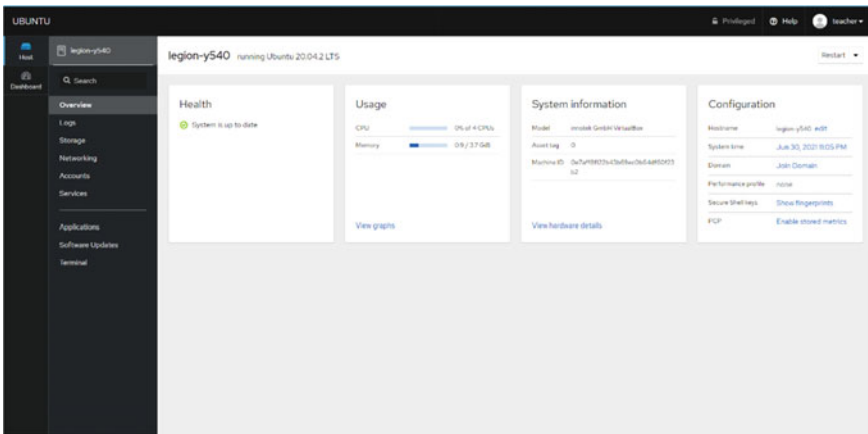


Fig. 8 Cockpit interface

7 Comparative Study of Different Computing Models

Following are three types of computational models which come under network booting:

- (a) **Thick Clients:** A thick client is essentially a networked computer that has most of the resources, including the operating system and many locally installed applications. Generally, they have sufficient RAM, powerful processors, GPU graphics resources, audio functions, hard drives, CD-ROM or DVD drives, and network card [6]. The main disadvantages of thick clients are higher energy costs, higher support and services, and increased total cost of ownership.



Fig. 9 Netdata interface

- (b) **Thin Clients:** A thin client in a narrow sense is an inexpensive computing device in a client/server architecture whose only function is to process keyboard commands and provide output on the screen. Thin clients often lack processing power, persistent storage, or I/O support [1, 6]. Although thin clients have proven their value in improving software performance, surfing the Internet, and other less-dependent applications with poor graphics performance, frustrations result when attempting to use thin client technology for multimedia, gaming and other graphic intense applications.
- (c) **Diskless Clients:** A diskless workstation or PC on a local area network (LAN) is a computer system that does not have its own disk. Instead, it stores files on a network file server. Diskless client is in fact a hybrid model, which can offer an acceptable alternative between thin client and fat client [7]. Table 1 shows the functional difference between three types of computing models included in centralized network booting:

Table 1 Comparative analysis of computing models

Parameters/metrics	Thin client	Diskless client	Fat client
Local HDD	No	No	Yes
Local processing	No	Yes	Yes
RAM capacity	Low	Moderate	High
Accelerated graphics	No	Yes	Yes
Terminal services	Yes	Optional	No
Energy consumption	Low	Low	High
Support costs	Low (centralized)	Low (centralized)	High (PC-based)

The unit cost and support cost offered by diskless clients are low and also give accelerated graphics for users who use high-end applications and ease administration tasks. The previous comparison of computing models begs the question: What can diskless clients offer to the organization which neither thick nor thin clients can provide? Diskless clients provide the best compromises for large computing environments where the total cost of ownership, energy saving, flexibility, and functionality are necessary. Thin clients cannot subvert the popularity and functionality of thick clients, and without a significant increase in funding and support, thick clients cannot be supported because there is no model that provides a framework technology that meets all the goals, and hence, requirements provided by diskless clients give an opportunity to support current and future educational needs [8].

8 Experimental Analysis

The main purpose of the Linux diskless client technology is: the user's PC becomes the secondary of the actual business process and allows the user to perform the required tasks without worrying about viruses, failures, access, and support. This creates an educational environment with a lower total cost of ownership, lower maintenance and support costs, while drastically reducing power consumption. While experimenting, Linux-based diskless clients were used for computing which can provide teachers and students with a safe, stable, and inexpensive computer information system. The following analysis is done based on the practical implementation conducted in the collage lab and the observed results.

- (a) **Boot Speed Analysis:** In order to improve the boot speed of the client workstation, all the start-up services were analyzed which were started during the boot point. After analysis, all unnecessary services were removed, disabled, and blocked leading to the improvement in the boot speed of the diskless client PCs. Table 2 shows the time required for 1 PC and 47 PCs to fully boot over the network:

The average boot time a single Linux-based diskless client takes is around 10–15 s, and for 47 PCs, it takes approximately 4 min and 30 s at 100 Mbps network speed, which is fairly good. As the number of diskless clients to be booted increases to 100 or more, the time required to boot increases, depending on the speed of the network. In addition to the boot process, many other improvements are made to the diskless client image. When starting up to 300 clients from the Ubuntu server, it is assumed that the boot time will be fast

Table 2 Boot speed analysis

No. of PC	Boot time (in s)
1	10–15 s
47	4 min 30 s

Table 3 Storage analysis

OS used	Storage consumed by OS (GB)	Storage after new user creation (MB)
Windows 10	17.1	736
Ubuntu 20.04.2 LTS	7.1	393.2

or slow, depending on the number of active clients and the network backbone. For a large number of diskless clients, there are many factors that affect system optimization. This includes ensuring optimized network infrastructure, diskless client functions, etc. Since the diskless client is a hybrid model that combines the advantages of the thin client and the thick client, it can integrate functions to make the performance of the diskless client much better than the real thin client.

- (b) **Storage Analysis:** The diskless client does not have any disk, and all its software and storage requirements depend on the server. The diskless client remotely mounts its root (/), /usr, and /home file systems from the server. The main server having 1 TB storage space was used as the centralized storage for all the 47 clients.

Table 3 shows the storage difference between different OS, i.e., Windows 10 and Ubuntu 20.04.2 LTS. After the fresh install of both OS in the server PC, the storage consumed by Windows 10 is more compared to Ubuntu. In addition to the foregoing, for creation of new users, Windows takes more storage space. In both cases, Ubuntu saves a total space of 10–13 GB approximately, and this saved storage space can be used for other purposes like creating more users, backup, and other applications.

- (c) **Cost Analysis:** The total cost of ownership of the PC classroom model is high. Software licensing, support, and updates on PCs are expensive; licenses must be purchased for each computer, and IT personnel must install applications and updates separately on each device. It also requires high costs for virus protection, service packs, and patches, as well as the hardware and support personnel required to maintain each device. Hardware used in schools or colleges is typically only 1–4 years old and expensive to replace when it dies and creates e-waste. Outside of normal working hours, any computers that have not been turned off by users or service personnel at the end of each day will continue to use electricity at night and on weekends [8]. The costs related to the clients utilized in a diskless client model are comparatively low. Diskless clients utilize operational systems and software system applications that may be accessed from a close-by server.

Figure 10 shows the comparison of cost of ownership between thick PCs and diskless client PCs per unit. Our campus consists a total of approximately 700 PCs. The cost of these 700 PCs is approximately Rs. 38,500,000, including hardware, software, licensing, virus protection, installation, and other costs. Hence, if the cost for single unit is calculated, it comes around Rs. 50,000–Rs. 60,000. Whereas, if

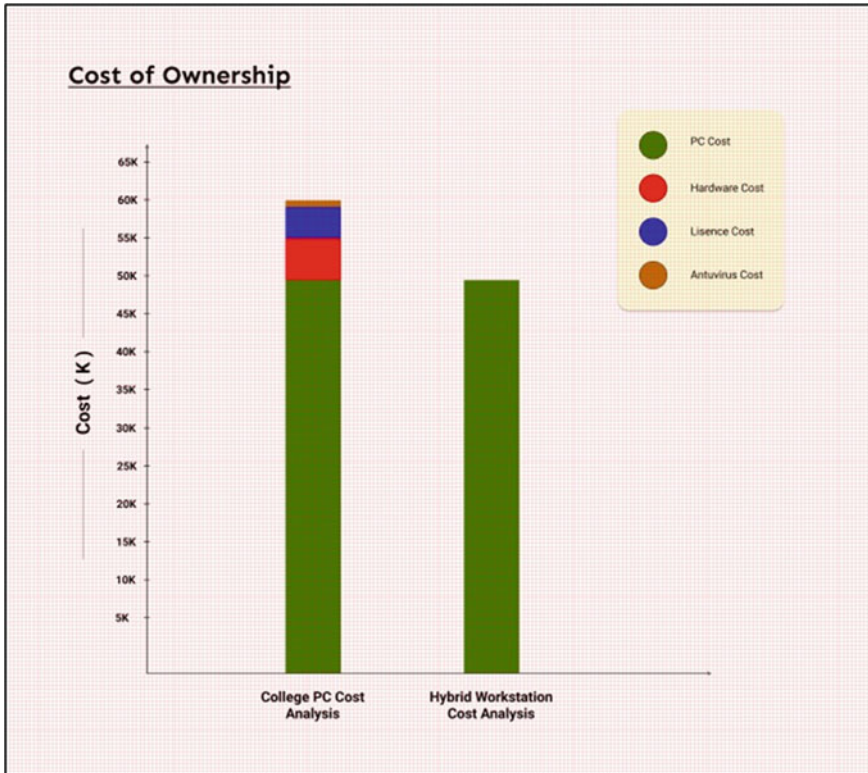


Fig. 10 Cost analysis graph

these 700 PCs use diskless computing model, then the total cost would eliminate the cost of HDD, software license (since Ubuntu is open-source), and antivirus cost. Now, these 700 Linux-based diskless workstations will cost around Rs, 18,412,100 and per unit it will cost approximately Rs. 26,303. Hence, implementing diskless clients will reduce the overall budget.

9 Conclusion and Future-Scope

Diskless client technology is not a new technology; however, the way it is deployed is new and allows for accelerated video and audio transmission that greatly improves the user experience. Because they can scale very cheaply, diskless clients do not focus on the user’s PC but on the use of the software and applications. The diskless clients that have been implemented and continue to implement enable GUI support, ubiquitous access not only over the network but also securely using software such as Cockpit and Netdata. In addition, extraordinarily little and significant support, maintenance,

and energy consumption can be seen. This system can be further enhanced with following possible additions:

- Adding more module support to the cockpit framework to deliver any necessary functionality.
- An easy-to-install script can be developed, allowing the system to be setup from the ground up.

Compared with desktop solutions with centralized support, it provides software standardization, better development, and ubiquitous accessibility, and also costs less than half of desktop solutions.

Limitations

- If the computer server fails, all of the computer clients are rendered ineffective. Clients are unable to boot.
- If the centralized LAN (Ethernet) cable is broken, all computer clients will fail.
- It is necessary to learn some technical aspects of utilizing diskless boot software, as well as a plethora of troubleshooting guides for software- and hardware-related issues in a diskless setup.

Acknowledgements We are incredibly grateful to the Computer Engineering Department of SIES Graduate School of Technology, Navi Mumbai, Maharashtra to let us implement and demonstrate our project in their Advanced Computing Lab.

References

1. C.-T. Yang, W.-F. Hsieh, H.-Y. Chan, Implementation of a diskless cluster computing environment in a computer classroom, in *IEEE Asia-Pacific Services Computing Conference* (2008)
2. A. Aryanti, A.S. Handayani, I. Ziad, I. Mekongga, F.J. Abiyuu, *Compatibility of Linux Architecture for Diskless Technology System* (2021)
3. C.-T. Yang, P.-I. Chen, S.-Y. Chen, H.-Y. Tung, *A Jobs Allocation Strategy for Multiple DRBL Diskless Linux Clusters with Condor Schedulers* (IEEE, 2006)
4. B.S. Sonawane, R.R. Deshmukh, User control on diskless client server platform. *Int. J. Innov. Eng. Technol.* (2018)
5. S. Bohringer, *Building a Diskless Linux Cluster for High Performance Computations from a Standard Linux Distribution* (Apr 2003)
6. A. Daga, M. Ghumnani, P. Pawar, *Survey of Diskless Workstation* (2015)
7. A.A.N. Balute, D.B. Gonzales, M.D. Macalaguing, C.J. Agaab, *Linux Based Diskless System using RSYNC Algorithm* (2016)
8. G. Ferrie, *The Benefits of Managed Diskless Client Technologies in an Educational Environment* (July 2011)

Analysis of Deep Learning Models for Early Action Prediction Using LSTM



D. Manju, M. Seetha, and P. Sammulal

Abstract Video surveillance is being increasingly adopted for ensuring safety and security both in public and private places. Automated prediction of abnormal events like theft, robbery, murder etc., from continuous observation of surveillance videos is a multidisciplinary study involving computer vision, deep learning and artificial intelligence. Deep learning-based video analysis and categorization is a most researched topic. Many deep learning models based on long short-term memory (LSTM) are proposed for automated prediction of abnormal events. This work does a comparative analysis of six LSTM-based deep learning models for abnormal event prediction from surveillance videos. Deep learning models of ResNet, VGG16, VGG19, 3DCNN, Inception V2 and Inception-ResNet-V2 are combined with LSTM for prediction of abnormal event from past observation of events in the video stream. These six models are run against different benchmarked abnormal event detection datasets and performance is compared in terms of accuracy, loss and execution time. It is observed that Inception-ResNet with LSTM provides training accuracy of 80% and test accuracy of 80% compared to other models.

Keywords 3DCNN · Inception · Inception-ResNet · VGG16 · VGG19 · ResNet

1 Introduction

In video surveillance, systems are being increasingly deployed in many places like roads, stations, airport, mall etc., for public safety. Detecting abnormal events from video surveillance systems is very important for security applications. Detecting

D. Manju (✉) · M. Seetha
G. Narayanamma Institute of Technology and Science, Hyderabad, India
e-mail: s.r.manju@gnits.ac.in

M. Seetha
e-mail: maddala.seetha@gnits.ac.in

P. Sammulal
JNTUH CEJ, Jagtial, India
e-mail: sam@jntuh.ac.in

abnormal activities can provide better security to the individuals. People and their interactions must be constantly monitored for longer duration and any abnormal activity must be predicted. It is difficult for trained personnel to reliably monitor videos for longer duration and predict abnormal events. With the need to automate this activity with high accuracy, many autonomous abnormal activity detection systems are proposed. The goal of any autonomous anomaly recognition system is to detect/predict any offensive or disruptive activities in the surveillance video in real time. The conventional systems extract various features of appearance, dynamic relationship and interactions between the entities in the video and classify them to detect any abnormal activity. The accuracy is limited in this approach due to insufficiency of handcrafted features to detect the abnormal activity. As abnormality is context dependent, identification of features which represent the activity in the relevant context is challenging. Recently, deep learning algorithms are being used for many computer vision problems. Deep learning algorithms learn features automatically and provide better accuracy. Deep learning uses discriminative feature representations of both appearance and motion patterns to model the event patterns.

In this work, a comparative analysis of six deep learning LSTM-based models for abnormal event prediction is presented. With capability of learning long-term dependencies and ability to extrapolate temporarily sequential data, long short-term memory (LSTM) is best suited for abnormal event prediction. LSTM combined with deep learning event classification can provide a better accuracy of abnormal event prediction. This work explores six different deep learning models of ResNet [1], VGG16 [2], VGG19 [2], 3DCNN (3D deep convolutional neural network) [3], Inception [4] and Inception-ResNet [5] in combination with LSTM for abnormal event prediction. The performance of these four models is compared against benchmarking datasets in terms of accuracy, loss and execution time.

2 Deep Learning-Based LSTM Models

The six deep learning-based LSTM models used for abnormal activity prediction are detailed in this section. LSTM is combined with ResNet, VGG16, VGG19, 3DCNN, Inception V2 and Inception-ResNet-V2. LSTM is an adapted version of recurrent neural networks to solve the problem of vanishing gradient. LSTM has a memory unit. This memory unit encodes the knowledge learnt. It learns when to forget and update hidden states when new information is provided as input. Memory unit functionality is controlled by three gates: input gate (i), forget gate (f) and output gate (o). The update and output functions are defined as below

$$i_t = \sigma(W_{ix}x_t + W_{im}m_{t-1} + b_i) \quad (1)$$

$$f_t = \sigma(W_{fx}x_t + W_{fm}m_{t-1} + b_f) \quad (2)$$

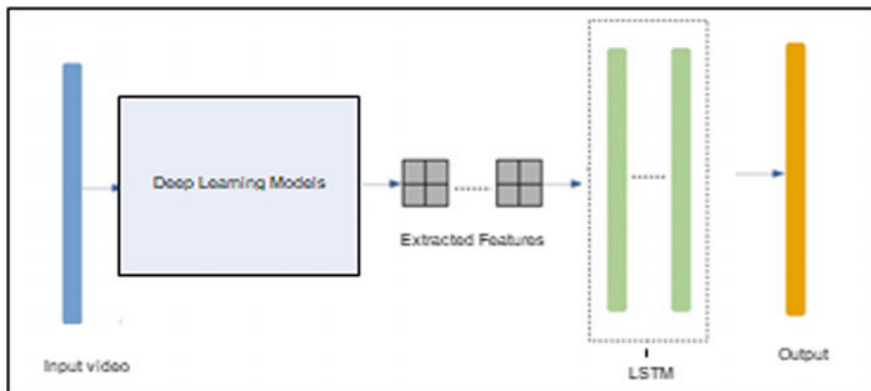


Fig. 1 Deep learning LSTM model

$$g_t = \sigma(W_{cx}x_t + W_{Cm}m_{t-1} + b_c) \tag{3}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \quad h_t = o_t \odot c_t \tag{4}$$

$$\sigma(x) = (1 + e^{-x})^{-1} \tag{5}$$

where $\sigma(x)$ is input mapping sigmoid, \odot is the matrix representing the parameters of the gates. W represents product operation with values of gate. LSTM control multiple gates to mitigate vanishing gradient problem and capture temporal dependencies. The overall architecture of the proposed comparison model is given in Fig. 1.

A deep learning models extract features and provide to LSTM for prediction of abnormality. Four different deep learning models of ResNet, VGG16, VGG19 and 3DCNN are used for extracting features.

2.1 ResNet with LSTM

ResNet or residual network was proposed by Microsoft researchers in 2015 as a solution to the problem of vanishing/exploding gradient with the increase in number of layers in deep convolutional neural network. Skip connection strategy is adopted in the ResNet network to enable feature learning ability. Due to this, the depth of the network is expanded, leading to improved learning ability. Gradient vanishing problem is solved effectively in ResNet due to passage of useful information to next layer with skin connection model.

ResNet with LSTM abnormality prediction model uses ResNet-50 for spatial feature extraction and LSTM for prediction using temporal feature extraction. The ResNet-50 model used for spatial feature extraction is of following configuration.

The ResNet-50 is trained using transfer learning strategy to boost the training performance. The advantage of transfer learning is that it is able to reduce the number of parameters to be trained and achieved convergence faster.

2.2 VGG16 with LSTM

VGG16 is a simple deep convolutional neural network. It is formed by stacking deep CNN followed by two fully connected layers. The fully connected layers have 4096 neurons on each of them. In VGG16 with LSTM model, VGG16 network [6] prepares the input feature vector for LSTM which predicts abnormality.

Input frames from the video are sized to $224 * 224$. The features are extracted using VGG16 and provided to LSTM in a time sequence to provide abnormal or normal class as output.

The dimension of the extracted features are (7, 70, 512). The feature vector is sliced into a set of ten samples with each of size $\{7, 7 * 512\}$. This feature vector is given as input to LSTM with ten time steps. The output of the last cell in LSTM is a binary output which gives as 1 for abnormal and 0 for normal.

2.3 VGG19 with LSTM

VGG19 is similar to VGG16, but it is deeper than VGG16. Due to this depth, VGG19 is able to extract more low-level features from a frame than VGG16. VGG19 has 16 layers for convolution, 3 layers fully connected, 5 Max pool layer and 1 Softmax layer.

Feature extraction part in this architecture is from input layer to the last max pooling layer ($7 \times 7 \times 512$). The feature extracted from VGG19 model is passed to LSTM to learn the long-term dependencies between the video frames and predict abnormal activity. The pipelining of VGG19 with LSTM is same as that used for VGG16.

2.4 3DCNN with LSTM

The architecture of 3DCNN with LSTM is preferred in video processing as it takes care of spatio temporal feature extraction from raw input video. There are 20 layers in this network. Convolutional layers are 12, followed by 5 pooling layers and one layer for FC, LSTM and output. The convolution block with paired with two or three

2D CNN and a pooling layer. It is followed by a dropout layer. The dropout layer has a dropout rate of 25%. Features are extracted using convolutional layer with $3 * 3$ kernel. ReLU activation function is used in the convolutional layer. The input dimension of image is reduced using max pooling layer with 2×2 kernels.

LSTM is at last to extract time information. The output shape after convolution is (none, 7, 7, 512). The input size of LSTM layer becomes (49, 512) due to reshape method.

After analyzing the time characteristics, the video frames are passed through fully connected layer to predict the category of output: Normal/Abnormal.

2.5 Inception

The Inception V1 also called GoogLeNet produced lowest error for ImageNet classification dataset. But use of $5 * 5$ convolutions because decrease of input dimension and resulted in reduced accuracy. To improve it, Inception V2 architecture replaced $5 * 5$ convolution with two $3 * 3$ convolution. In addition to increased accuracy, this change also reduced the computation time by 2.78 times compared to Inception V1.

The feature extracted from Inception model is passed to LSTM to learn the long-term dependencies between the video frames and predict abnormal activity. The pipelining of Inception V2 with LSTM is same as that used for VGG16.

2.6 Inception-ResNet

Inception-ResNet-v2 is a convolutional neural architecture that builds on the Inception family of architectures but incorporates residual connections (replacing the filter concatenation stage of the Inception architecture). The architecture of Inception-ResNet-v2 is a combination of two blocks (inception block and residual network).

The Softmax layer is removed, and the features are passed to LSTM to predict the abnormal activity.

Summary of all models

See Table 1.

3 Results

The performance comparison of the six deep learning-based LSTM models was done using following setup (Table 2).

To start the work, we used front end as keras and back end as TensorFlow. Keras is an open-source library which provides artificial neural networks with a Python interface. Keras serves as a TensorFlow library GUI. Keras uses TensorFlow 2.0's high-level API, an open, highly efficient framework to solve machine learning issues with a focus on modern deep learning. For learning the environment, LSTM model is used, which in turn helps for predicting the next frames.

Table 1 Comparison of all models in terms of number of layers, parameters it supports, advantages and disadvantages by using the model

Model	Advantages	Disadvantages	No. of layers	No. of parameters
VGG16-Visual Geometry Group 16	<ol style="list-style-type: none"> 1. It can be trained up to 22,000 categories 2. The filter size is uniform throughout all the layers 	<ol style="list-style-type: none"> 1. It is slow to train 2. The network architecture weights are quite large 	13 convolutional layers and 3 fully connected layers	138 million parameters
VGG19-Visual Geometry Group 19	<ol style="list-style-type: none"> 1. The pretrained network can classify images into 1000 object categories 2. The filter size is uniform throughout all the layers 	<ol style="list-style-type: none"> 1. It can take large amount of resources like computational time 	16 convolutional layers and 3 fully connected layers	140 million parameters
ResNet called Residual Network	<ol style="list-style-type: none"> 1. ResNet takes less memory 2. Faster Inference Time 3. Allows deeper networks to be trained 4. Supports skip connections which help to increase performance 	<ol style="list-style-type: none"> 1. Increased complexity of architecture 2. Implementation of Batch Normalization is needed 3. Adding skip connections for which you have taken into account the dimensionality between the different layers which can become a headache to manage 	50 layers	23 million trainable parameters

(continued)

Table 1 (continued)

Model	Advantages	Disadvantages	No. of layers	No. of parameters
Inception	1. It has 22 layers 2. It has less parameters than VGG16 3. Inception Net learns more complex features Computationally efficient 4. Auxiliary classifier is used	Number of 5×5 convolutions can be prohibitively expensive on top of a convolutional layer with a large number of filters. This problem becomes even more pronounced once pooling units are added to the mix: their number of output filters equals to the number of filters in the previous stage	22 layers	26 million trainable parameters
Inception-ResNet	Training the network is much is faster and has better final accuracy	Increased network to manage	164 layers	56 million trainable parameters

Table 2 Performance configuration

PC configuration	Intel i7, 8 GB RAM, Nvidia MX350
Software tools	Python 3.7, Keras, tensor flow
Dataset	UCSD anomaly detection dataset
Number of training samples	34 videos
Number of test samples	26 videos

After the feature extraction is done, the objects in each frame are classified into respective classes like cat, dog, car, fire etc. The frames are also classified into anomalous and non-anomalous frames. So, in a video streaming, if there is any occurrence of unusual activity or any anomaly, it immediately reports because of the classification.

The ResNet with LSTM and VGG16 with LSTM model are able to achieve an accuracy of 80% at epoch of 4 s. VGG19 with LSTM is able to achieve 80% accuracy only at epoch of 8 s. Inception with LSTM is able to achieve the highest accuracy of 80% at epoch of 7 s. Out of all methods, Inception-ResNet with LSTM is able

to achieve 80% accuracy from start. There is not much difference in accuracy in terms of value as almost all the models are able to achieve 80%, but Inception-ResNet with LSTM is able to achieve it from it compared to more epochs taken in other solution. Use of residual layer and better protection against local gradient has helped Inception-ResNet with LSTM to achieve the highest accuracy in less epochs compared to others.

The loss achieves is minimal value of 56% at epoch of 4 s in ResNet with LSTM. VGG16 with LSTM is able achieve loss of 44% at 9 s. VGG19 with LSTM is able to achieve loss of 56% at 9 s. Loss is only 6% in Inception with LSTM at epoch of 6 s. Inception-ResNet with LSTM is able to achieve a loss of 10% at epoch of 1 s. The loss is lowest in inception with LSTM due to use of convolutions of different sizes to capture details at varied scales. But combining inception with ResNet increased the error rate.

The comparison of training and test accuracy and loss across the six models are given Table 3.

The result shows that two models of ResNet, VGG16 with LSTM are able to achieve same accuracy of 70%. The accuracy of 3DCNN is 76% which is lower compared to VGG19 model, but the loss is lower in 3DCNN. The loss is very low in Inception-ResNet with LSTM at 22%.

The average execution time is compared against all the six different models and the result is given Table 4.

The execution time is lower in Inception-ResNet with LSTM compared to other models due to residual connections. 3DCNN feature extraction takes almost same

Table 3 Comparison of loss and accuracy

Model	Training		Test	
	Loss	Accuracy	Loss	Accuracy
ResNet with LSTM	0.49	0.80	0.60	0.73
VGG16 with LSTM	0.64	0.70	0.60	0.70
VGG19 with LSTM	0.64	0.60	0.68	0.72
3DCNN with LSTM	0.60	0.68	0.54	0.66
Inception with LSTM	0.66	0.80	0.58	0.75
Inception-ResNet with LSTM	0.22	0.80	0.17	0.80

Table 4 Comparison of execution

Model	Average execution time (s)
ResNet with LSTM	75
VGG16 with LSTM	50
VGG19 with LSTM	60
3DCNN with LSTM	72
Inception with LSTM	70
Inception-ResNet with LSTM	21

Table 5 Precision across models

Model	Precision
ResNet with LSTM	70.4
VGG16 with LSTM	71.2
VGG19 with LSTM	72.4
3DCNN with LSTM	61.4
Inception with LSTM	73.1
Inception-ResNet with LSTM	74.1

Table 6 Recall across models

Model	Recall
ResNet with LSTM	68.4
VGG16 with LSTM	62.12
VGG19 with LSTM	66.55
3DCNN with LSTM	60.12
Inception with LSTM	69.45
Inception-ResNet with LSTM	70.51

execution time as that of ResNet with LSTM even though layers are less due to 3D feature complexity.

Precision

The precision for detection of abnormal events in each of the models is shown Table 5.

Recall

The recall for detection of abnormal events in each of the models is given Table 6.

4 Conclusion

A comparative analysis of six deep learning-based LSTM models for abnormal activity prediction is presented in this work. ResNet, VGG16 and Inception when combined with LSTM are able to provide an accuracy of 70%. Inception ResNet with LSTM is able to provide training accuracy of 80%, a lowest loss of 22% with a lower execution time of 24.66 s compared to other deep learning with LSTM models. It is observed that there is no much difference in test accuracy between the ResNet and Inception model. The work can be extended further to more classes of anomalies.

References

1. K. He, X. Zhang, S. Ren, J. Sun, *Deep Residual Learning for Image Recognition*. [arXiv:1512.03385](https://arxiv.org/abs/1512.03385) (2015)
2. Decentralization of Control Loop for Self-Adaptive Software through Reinforcement Learning, in *24th Asia-Pacific Software Engineering Conference Workshops (APSECW)* (Nanjing, China, 2017)
3. S. Ji, W. Xu, M. Yang, K. Yu, 3D convolutional neural networks for human action recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(1), 221–231 (2013)
4. H.N. Ho, E. Lee, Model-based reinforcement learning approach for planning in self-adaptive software system, in *Proceedings of the 9th International Conference on Ubiquitous Information Management and Communication*, 2015
5. D. Kim, S. Park, Reinforcement learning-based dynamic adaptation planning method for architecture-based self-managed software, in *SEAMS'09. ICSE Workshop on Software Engineering for Adaptive and Self-Managing Systems*, 2009
6. H. Qassim, A. Verma, D. Feinzimer, Compressed residual-VGG16 CNN model for big data places image recognition, in *Proceedings of the 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)* (Las Vegas, NV, USA, 8–10 Jan 2018)
7. Y. Liu, Ke., Xu, J. Xu, Periodic surface defect detection in steel plates based on deep learning. *Appl. Sci.* **9**, 3127 (2019). <https://doi.org/10.3390/app9153127>

Identification of Masked Face Using Deep Learning Techniques



M. Sheshikala, P. Praveen, and B. Swathi

Abstract Face detection is an artificial intelligence [AI]-based methodology, where a cascade function is prepared with a bunch of information and input data. It is prepared from a great deal of positive and negative pictures. It is then used to identify objects in different pictures. The proposed model's classifier essentially increases the precision and strength of robustness search on faces with demeanor variation and vague forms. In the element extraction stage, this research work leverages a philosophy for increasing efficiency through the connection of two strategies: mathematical (geometrical) element-based approach and independent component analysis strategy. To implement face coordination step, this research work proposes a model that combines several neural networks to coordinate with the mathematical highlights of the human face. Since the proposed model connects several neural networks, it is termed as multi-artificial neural network. MIT + CMU information base is utilized for assessing the proposed strategies for face identification and arrangement. At last, the experimented results on Caltech dataset show the possibility of implementing the proposed model.

Keywords Face · CNN · Neural networks · ANN

1 Introduction

Acknowledgements is the process of identifying a person, object, or individual from information. Furthermore, we may understand that face recognition is a deep learning computation that is used to detect individuals from images, videos, or webcams. Profound learning and artificial intelligence [AI] techniques are the recent advancements that are making their particular imprints in individual fields such as self-driving vehicles, item suggestions, security [1], promoting [2]. We are relying upon these advances since they diminish the manual human work and builds the productivity. This paper is about the task that perceives the human faces and gives the precision of human parts like ears, nose, decorations, garments, and so on as the yield.

M. Sheshikala (✉) · P. Praveen · B. Swathi
School of CS and AI, Department of CS and AI, SR University, Warangal, India

1.1 Neural Network

The combination of many neural networks is used for checking similarity in the geometric features of human face. Deep learning techniques are classified as machine learning and artificial intelligence approaches because they learn data representation and abstraction. The data might take the form of a picture, a sound, or text. Deep learning algorithms produce outcomes that mimic people, including how humans think and proceed. Deep learning is commonly associated with the multi-layered structure of algorithms known as neural networks. This multi-layered structure mimics the human intelligence in analyzing the data. Neural network is represented as a collection of connected nodes. The above nodes are represented as neurons. These artificial neurons slightly represent the neurons of brain. The connection between nodes is considered as the axon in the human brain. The role of an axon is to transmit information to different neurons in the brain [3].

By training the data, numerical values of these connected neurons get changed.

Input layer

Input is given to the very first layer, which is called as an input layer. Here, the input is an image. Input has elements same as the number of neurons in the input layer. Here, the input layer has three neurons so that the input should also contain three elements.

Hidden Layer

Here, mathematical computation is done by using the dot product. Input layer takes the input. Weights are represented in the form of a matrix. Dot product is performed on inputs and weight matrix.

The output of this dot product is called as an activation function. This output is given as input to the next hidden layer and similarly, it can be calculated by using the weight matrix of hidden layer 2; this process is repeated until we reach the output layer.

Output layer

The final output is obtained from the hidden layer.

1.2 Face Detection

Face detection is an AI-based innovation that can distinguish and find the presence of human appearances in advanced photographs and recordings. It may be viewed as an exceptional instance of item class identification, where the assignment aims to discover the areas and determine the spans of the relative multitude of articles that have a place with a given class—for this situation, faces—inside a particular picture or pictures.

Due to the recent advancements in face recognition technologies, it is now feasible to recognize faces in an image or video, paying less attention to the presence, lighting conditions, and skin tone. Face recognizable proof applications use computations that choose if pictures are positive pictures (for instance pictures with a face) or negative pictures (for instance pictures without a face). To have the alternative to do this correctly, the estimations ought to be set up on gigantic datasets containing incalculable face pictures and non-face pictures. At the point when arranged, the computations can react to two requests due to commitment to the kind of an image.

- Are there any countenances in this picture?
- If indeed, where right?

If a face or faces are accessible in an image, the computations will react to these requests by setting a skipping box around the perceived face. Beforehand, these computations were AI based, and were seriously affected by parts. For instance, incredible head presents (where the head is turned far aside or moved far up or far down, for example) and fluctuating lighting conditions [4, 5]. Today, regardless, significant learning procedures can be used to finish exact face area in a wide extent of circumstances.

2 Literature Review

A. S. Tolba, A. H. El-Baz, and A. A. El-Harby, this paper provides an up-to-date review of literature review them recent face recognition techniques present is. Description major human face recognition research. We first present another re perspective on face acknowledgment and its applications. Then, at that point, restrictions of face information bases which are utilized to test the exhibition of these face acknowledgment calculations have given the presentation of these face acknowledgment calculations. A short outline of the face acknowledgment seller test (FRVT) 2002, a huge scope assessment of programmed face acknowledgment innovation, and ends are likewise given. At last, we give a synopsis of their indexed lists.

To make a start to finish face cooperation of the board structure, in the year 2013 Chintalapati et al. [6] proposed a procedure to do the cooperation structure using the Viola Johns technique [7] for face revelation, followed by histogram balance for feature extraction, and afterward SVM classifier has been utilized for face affirmation. Later in 2017, Rathod et al. [8] proposed a beginning to end face support structure by using same technique for face recognizable proof and request for the face affirmation. In any case, these procedures relied upon customary AI-based computation. In 2017, Arsenovic et al. [9] proposed the top-tier methodology called FaceTime using CNN course for face revelation and CNN for making face embeddings and afterward they are used for face affirmation.

3 Existing System

Different algorithms are developed by many developers to deal with the face detection and give accurate results. Some of them claim that their algorithm is the best, but many algorithms fail at a point where they cannot handle face recognition task. Some of the algorithms or methods were used in many cases and applications. Years pass by and people are still developing various methods and algorithms to deal with the face recognition and create a better algorithm, which give more accurate result than the existing algorithms.

3.1 *Skin Color-Based Algorithm*

Human skin tone is likewise an exceptionally simple one to separate one from other; once in a while, we can even recognize an individual's country just by his/her skin tone. This research work utilizes the facial element technique by including the skin tone, which is additionally a basic calculation used for recognizing the skin pixels, which is utilized for skin shading calculation. In this, every single pixel is characterized into two kinds, skin tone and non-skin tone. The grouping depends on the shading segment. For some random information present in the picture, the technique has used the shading space for the skin locale as the classifier. Then, at that point, it is applied to the face as cover and afterward bound box is attracted to separate the face from the given information picture [10, 11].

There are three more popular color spaces they are:

- a. RGB.
- b. YCbCr (Luminance chrominance).
- c. HIS (Hue Saturation Intensity).

3.2 *Wavelet-Based Algorithm*

In this calculation for a given face picture, it will be depicted by a subset of a band separated pictures containing wavelet coefficient. It offers a probability of giving a vigorous multi-scale path examination of the given picture. This strategy is entirely adaptable as there exist different bases and we need to pick the appropriate reason for an application. Gabor wavelet technique is the most utilized strategy and for the most part utilized in picture surface investigation.

3.3 *Gabor Wavelet*

The Gabor technique utilizes spatial recurrence design and direction connection. This methodology works by recognizing short lines like end lines, arches, and so on. The bends are relating admirable with conspicuous highlights of human countenances as eyes, mouth, eyebrows, and so on.

3.4 *Artificial Neural Networks-Based Algorithm*

The ANN calculation is the most realized strategy in different applications. This calculation is mostly used for acknowledgement purposes. Once a face is differentiated for distinguishing and perceiving, the cycle begins. The ANN collects face data to determine the individual's identity. The ANN is like a human cerebrum; it comprises of a neuron network and different layers. Initially, a picture is given as an information, then the primary layer will attempt to dissect it, and then it will duplicate into a huge number and send it to the following layer, and at the last layer, all the weight esteems will be added. The ANN is isolated into different types, they are feed forward neural organization, backpropagation neural organization, and outspread premise work. These are the three ordinarily utilized types in ANN [10].

4 **Implementation**

This is predominantly carried out in three stages, such as making the dataset and afterward train them, work on perceiving the human appearances lastly create the exactness in the recognition of human facial parts and different trimmings. The dataset is made by gathering the photographs of the understudies for almost 15 to 30 for every understudy and make the different organizers for individual understudies with the separate move quantities of the understudies. We can make the live enlightening assortment of understudies by taking 5–10 s video by pressing “s” to save the photos of the understudies while taking the video and press “q” to exit and save them in the specific manner, which is referred by the customer. Then, the video is stacked when it is required and subsequently the video ought to be segmented into the housings. At the point, when this division association is finished, treatment of the edges will start.

Edge preparation includes workouts such as padding on each side with the dull and RGB tone, and it is then enlarged with appropriate estimates. In the following stage, the dimensionally reduced image is sent via the MTCNN, which transmits the face with the skipping boxes. Then, the restricted pictures with the bearings are used for managing the faces from housings, and it will be subsequently sent for the increment; this joins the factors like darkening, level moving, adding Gaussian disturbance,

flipping, salt and pepper ruckus, vertical moving, developing and decreasing the wonder. The increased faces are then diminished to 160×160 and subsequently saved in the display envelope. The entire cycle is reiterated for various housings present in the picture. Then, the dataset creation is finished. Following the completion of the dataset, the subsequent phase is to put them up. The prepared programming then authorizes the client to take the particular rate for planning, and the leftover rate is typically utilized for testing the already organized classifier. Resulting to preparing, it is saved moreover, later utilized for the assertion task. So as in the current structures, there is no description of precision and it do not show the results of human facial parts, so our system helps with portraying the accuracy, and it shows the level of precision of human facial parts and enhancement.

4.1 Algorithm

Recognition of a person’s face can be done by using Eigen’s faces, eyes, mouth, nose, mask and the relative distance between them are detected by using Eigensface. Eigen faces are collection of Eigenvectors used for face recognition.

Principal component analysis (PCA) is a mathematical tool that is used to extract the facial features from the given image. PCA is a dimension reduction method that reduces the dimension of large data sets.

Eigenvectors and Eigen values.

For a square matrix A, the Eigenvector and Eigenvalues are such that

$$AX = \lambda X \tag{1}$$

Here,

A is the vector function which is a square matrix.

X is the Eigenvector which is non-zero.

λ is a scalar called as Eigenvalue.

Linear transformation will scale the Eigenvector

Only the magnitude changes when the Eigenvector is multiplied by a matrix; only its magnitude changes, but not the direction.

Now,

$$AX = \lambda IX \tag{2}$$

This is equal to multiplying things by 1 and does not change the value of anything.

$$AX - \lambda IX = 0(A - \lambda I) = 0 \tag{3}$$

where I is $n \times n$ matrix.

For a system to have non-trivial solutions, $\det(A - \lambda I) = 0$.

Where $\det()$ denotes determinant. When it is calculated, we get a polynomial of order n , with n number of roots which are the eigenvalues.

Steps involved in face recognition: Initialization of process [3]

- i. Group the initial set of images called as training set.
- ii. Calculate Eigenfaces from training set. These M images define face space.
- iii. Calculate corresponding distribution in M -dimensional weight space.

The next steps after initialization are.

- (a) The weights of the given input image are calculated.
- (b) Evaluate M Eigenfaces by projecting the image which is an input, onto each of the Eigenfaces
- (c) Check the image in a face to verify whether the picture is close to a “free space.”
- (d) If it is distinguished as the face, then at that point, arrange the weight design as a referred to individual or as an obscure individual.
- (e) Known face or unknown is loaded. On the off chance that the non-indistinguishable individual’s face is seen a few times, ascertain the trademark weight design and join it into known countenances.

The last advance is needed by couple of frameworks and the other few do not need it is a discretionary prerequisite.

5 Results

The proposed project displays the accuracy details of the facial parts and ornaments of human as shown in Figs. 1, 2 and 3. Our project supports both group images and single images facial recognition.

This paper identifies each and every part of human facial parts and other ornaments that are placed on human face.

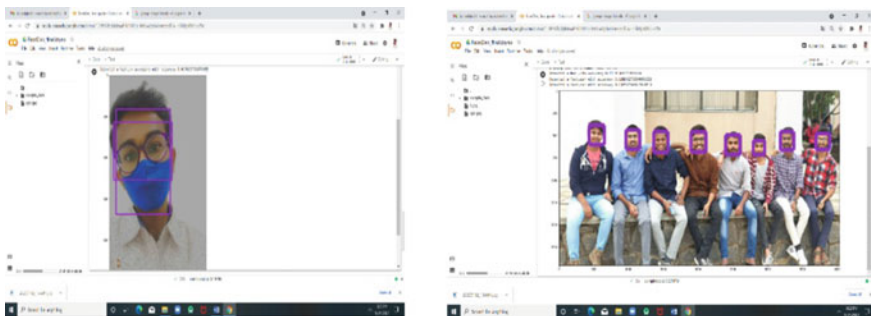


Fig. 1 Identification of human face which is masked

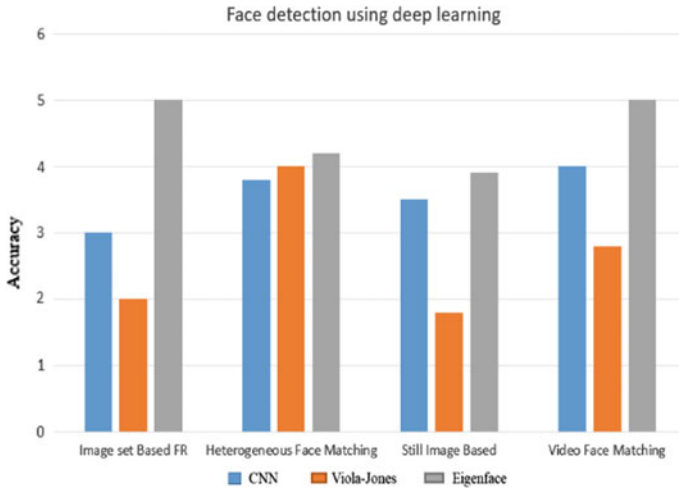


Fig. 2 Analysis of face detection versus different algorithms

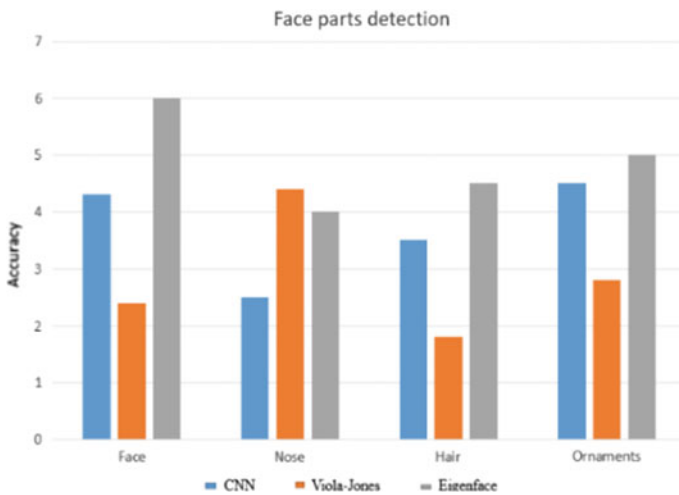


Fig. 3 Analysis of face parts detection versus different algorithms

6 Conclusion

This paper has successfully recommended the Eigenface facial recognition strategy. Also, an improved strategy has been proposed for face discovery for identifying different difficulties, for example, unique face points, different look, complex foundation, light, and so on. This research work has worked and executed an improved philosophy for face location based on various difficulties like distinctive face points,

different face appearance, troublesome foundation, light, and so on. We have utilized Eigenface algorithm for calculation and precision of this improved calculation of all the standard image database that is superior to the previously appeared in the outcomes segment. A pre-preparing step counting picture improvement and clamor evacuation has been proposed. Finally, we recognize the human face accurately. It is not yet reasonable for face acknowledgment in video reconnaissance; it is hard to be sure. Given the reality how helpful and practical this innovation is, this forecast is not fantastic. On the off chance that this expectation turns into a reality, any organization that executed the innovation today may acquire a serious advantage later on. We feel that recognizing associated faces was the hardest piece of the project. A significant amount of time was invested to come up with a format coordinated with data that adapts effectively to related faces, particularly those that are only partially visible.

References

1. C. Soutar, D. Roberge, A. Stoianov, R. Gilroy, B.V. Kumar, Biometric encryption using image processing, in *Optical Security and Counterfeit Deterrence Techniques II*, vol. 3314 (International Society for Optics and Photonics, 1998), pp. 178–188
2. M. Synder, K.G. DeBono, Appeals to image and claims about quality: understanding the psychology of advertising. *J. Pers. Soc. Psychol.* **49**(3), 586 (1985)
3. T.S. Kumar, Construction of hybrid deep learning model for predicting children behavior based on their emotional reaction. *J. Inf. Technol.* **3**(01), 29–43 (2021)
4. A. Sungheetha, R. Sharma, 3D image processing using machine learning based input processing for man-machine interaction. *J. Innov. Image Process. (JIIP)* **3**(01), 1–6 (2021)
5. G. Ranganathan, A study to find facts behind preprocessing on deep learning algorithms. *J. Innov. Image Process. (JIIP)* **3**(01), 66–74 (2021)
6. S. Chintalapati, Ragunadh, Automated attendance management system based on face recognition algorithms, in *Year 2013 IEEE International Conference held on Computational Intelligence and Computing, Research* (IEEE, 2013), pp. 1–5
7. P. Viola, M.J. Jones, Robust real-time face detection. *Int. J. Comput. Vision* **57**(2), 137–154 (2004)
8. H. Rathod, Y. Ware, S. Sane, S. Raulo, V. Pakhare, I.A. Rizvi, Automated attendance system using machine learning approach, in *2017 International Conference on Nascent Technologies in Engineering (ICNTE)* (IEEE, 2017), pp. 1–5
9. S.A. Arsenovic, D. Stefanovic, FaceTime-deep learning based face recognition attendance system, in *2017 IEEE 15th International Symposium on Intelligent systems and Informatics (SISY)* (IEEE, 2017), pp. 000053–000058
10. H. Rowley, S. Baluja, T. Kanade, Neural network-based face detection. *Proc. IEEE Trans. Pattern Anal. Mach. Intell.* **1**(20), 23–28 (1998)
11. C.V. Joe, J.S. Raj, Location-based, orientation context dependent recommender system for users. *J. Trends Comput. Sci. Smart Technol. (TCSST)* **3**(01), 14–23 (2021)
12. M. Sallauddin, D. Ramesh, A. Harshavardhan, S.N. Pasha, Shabana, A comprehensive study on traditional AI and ANN architecture. *Int. J. Adv. Sci. Technol.* **28**(17), 479 (2019). <http://sersc.org/journals/index.php/IJAST/article/view/2297>
13. M.A. Shaik, P. Praveen, R. Vijaya Prakash, Novel classification scheme for multi agents. *Asian J. Comput. Sci. Inf. Technol.* **8**(S3), 54–58 (2019). https://www.researchgate.net/publication/334030333_Novel_Classification_Scheme_for_Multi_Agents
14. A. Harshavardhan, M.D.S. Mohammad, D. Ramesh, K. RaviChythanya, Design methods for detecting sensor node failure and node scheduling scheme for WSN. *Int. J. Eng. Adv.*

- Technol. (IJEAT) ISSN: 2249–8958, V-9(-1). <https://www.ijeat.org/wp-content/uploads/papers/v9i1/A3081109119.pdf>
15. M. Sheshikala, Natural language processing and machine learning classifier used for detecting the author of the sentence. Int. J. Recent Technol. Eng. (IJRTE) (2019).<https://doi.org/10.35940/ijrte.C4098.098319>
 16. M. Sheshikala Sallauddin, S. Mohmmad, Survey on multi level security for IoT network in cloud and data centers. J. Adv. Res. Dyn. Control Syst. **10**(10), 134–146 (2018). <https://www.jardcs.org/backissues/abstract.php?archiveid=4556>

An Intrusion Detection Approach for Small-Sized Networks



Phong Cao Nguyen, Van The Ho, Dong Hai Duong, Thinh Truong Nguyen, Luan Anh Luong, Huong Hoang Luong, and Hai Thanh Nguyen

Abstract In any network system, the intrusion is undesirable, and organizations are constantly searching for solutions that could effectively detect intrusion and, consequently, help them to react appropriately. However, packaged enterprise solutions provided by industry-leading companies usually leave little room for optimization and control in the hands of the users and sometimes incur costs that small- and medium-sized organizations want to curtail, especially if the solutions are *smart*. This work demonstrates how such organizations may build their home-grown Deep Learning-based Intrusion Detection Systems (DL-IDSs) and integrate them into their existing network. We have implemented an Intrusion Detection System for Small networks using deep learning architectures. The proposed system evaluated on UNSW-NB15 dataset including more than 250,000 network packets and has obtained an accuracy of 89% in discriminating between abnormal and normal packets and 74% for various nine network attack types classification.

Keywords Intrusion detection systems · Small systems · Deep learning · Network attack types

1 Introduction

Network intrusion can be defined as any unauthorized access to a system. By gaining such access, attackers can proceed to later plans, causing direct damage to the organization. For instance, they may perform privilege escalation (e.g., having system rights that they should not have) and retrieve sensitive data from system databases or leave a backdoor for later exploitation. For example, IBM's cost of a Data Breach Report [1] pointed out that the average cost per lost or stolen record in 2021 is US\$ 161. This

P. C. Nguyen · V. T. Ho · D. H. Duong · T. T. Nguyen · L. A. Luong · H. H. Luong
FPT University, Can Tho, Vietnam

H. T. Nguyen (✉)
Can Tho University, Can Tho, Vietnam
e-mail: nthai.cit@ctu.edu.vn

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022
S. Smys et al. (eds.), *Inventive Computation and Information Technologies*, Lecture Notes
in Networks and Systems 336, https://doi.org/10.1007/978-981-16-6723-7_67

899

cost, multiplied by the number of records an organization usually holds, plus the loss of reputation, is simply unbearable to small organizations. Alternatively, large-scale unauthorized access may consume much of the bandwidth and resources, depriving legitimate users of the system, resulting in a Denial-of-Service (DoS) attack. In all of these cases, early intrusion detection is crucial to successfully protect a system by enabling preventative or remedial actions to be taken promptly—directing malicious traffic to a honeynet, for example.

Unfortunately, small organizations are particularly vulnerable to such attacks because of their limited budget for cybersecurity. Traditional firewalls or IDSs may be effective against common types of attack but do not have the ability to *learn*, hence having a tough time detecting unconventional forms of attacks. “Smart” IDSs, with an element of AI, are the solutions to this. However, most proposed models are mainly concerned with theoretical results and lack of implementation details in practical circumstances. This fact hinders small organizations from actively joining the game by developing, customizing, and deploying their solution.

These observations are the motivation for this paper. We propose a lightweight model that is suitable to the real-timeliness nature of the task and develops a practical implementation scenario. In Sect. 2, we review some previous works in the field of DL-IDS. Section 3 contains details of the system design and implementation. The results are found in Sect. 4, and the paper ends with a brief conclusion. By carrying out this research, we aim to make the following contributions:

- A not-too-complex model, trained on the UNSW-NB15 dataset with feature selection applied to make computation time shorter
- An implementation of the algorithms in [2] that extracts the additional features in the dataset (those that cannot be collected)
- We propose a system with different easy-to-use free tools utilized to help the system work smoothly from start to end, thus proposing a promising method of solving the stated problem. The research is expected to be what makes this research stand out from other works in the field—putting a model into actual use.

2 Overview Related Works

Only a few datasets are used across most of the studies in the field, indicating a lack of open-source, authentic, expert-validated datasets pertinent to the problem of general network intrusion. Some of them are DARPA (1998, 1999), KDD99 (and its refined version NSL-KDD), ISCX-IDS 2012, UNSW_NB15, and CSE-CIC-IDS (with annual updates).

KDD99 is one of the most widely used datasets in the field. Studies with remarkable results include [3] (97.85% accuracy using Stacked Non-symmetric Deep Auto-encoder) or [4] (99.91% accuracy using Restricted Boltzmann Machine Procedure). Although KDD99 is very popular among IDS studies, it is hard to assert whether these models work well in the real world. The experiment showed how the dataset was generated: mainly in an experimental environment [13].

Because of the significant amount of redundant records in KDD99, thus affecting the *real* accuracy of models, M. Tavallae et al. improved the dataset and came up with NSL-KDD [5]. Regarding this dataset, Chiba et al. [6] achieved an impressive 99.86% accuracy on this dataset with their DNN-based ML-IDS. They also trained this model on CSE-CIC-IDS 2017 and obtained similarly good results. The focus of this study is on cloud environments. However, although there were improvements compared to KDD99, it is worth noting that the data is still largely outdated as its core is still from KDD99 and does not reflect current trends in network attacks.

Other algorithms have also been proposed in [7] (genetic algorithm), [8] (gradient boosting tree), or [9] (decision tree), just to name a few. However, one of the first works to advocate the use of CNN in developing an IDS is that of Liu et al. [10]. In a later work, Wang et al. developed HAST-IDS and IDS with an automatic feature learning capability using CNN and LSTM [11]. Although these works were not tested against state-of-the-art datasets (for [10], KDD99 and for [11], DARPA 1998 and ISCX-IDS 2012), they laid some of the most important grounds of implementing the seemingly irrelevant CNNs into building DL-IDS.

In contrast to these studies, which focus on datasets extracted from metadata and packet flow, a whole new approach has also been proposed. Instead of looking only at extracted features in metadata, these IDSs instead also navigate the raw contents of the packets. This type of IDS is capable of detecting types of attack in which the main component of the attack is the payload; for example, SQL injection (in which attackers utilize the SQL language to execute unauthorized commands on the database) or XSS (in which attackers cleverly use Javascript instead of legal input). For example, M. Soltani et al. employed RNN and LSTM for their so-called *deep intrusion detection* (DID) [12] and reached a 0.992 precision against the quite-comprehensive dataset ISCX IDS 2017, which also included the content-based attack Heartbleed. In another research, A. Kim et al. built a CNN-LSTM model utilizing real-time web traffic and obtained encouraging results against three datasets (91.54% accuracy against CSIC-2010, 93% CSE-CIC-IDS 2017 HTTP 98.07% against their generated web traffic dataset) [13]. Although the research only achieved initial good results (the authors claimed that the IDS needs to be re-validated due to its high false-positive rates [13]—partly because benign and malicious packets have quite similar raw contents), these works have successfully introduced researchers to an alternative path.

3 Methodology

3.1 Background

3.1.1 CNN

Although most usually used to process two- and three-dimensional inputs like image or video, Convolutional Neural Networks (CNNs) are also suitable for any data where a spatial ordering exists. Time is one such ordering, and network traffic data, which,

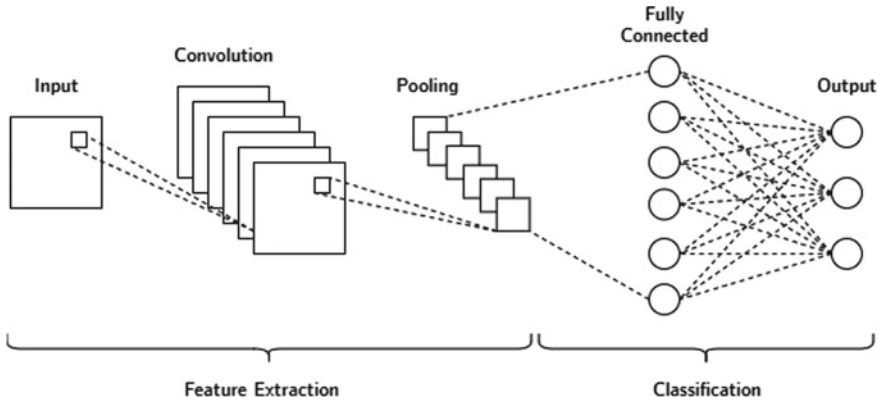


Fig. 1 The components of a convolutional neural network

of course, includes timestamps, is one type of one-dimensional data on which CNNs can be applied. Figure 1 illustrates the components of a CNN, using the type of data it is most widely applied on—2D data. It contains an input layer, a hidden layer (which comprises other layers), and an output layer. First, a filter (of the same size as the small square in the input layer) slides through the input matrix, left to right, top to bottom, and is dot-multiplied with the equal-sized part of the input matrix to discover features within the input. Then, the pooling layer is applied to reduce the size of the feature map but still retains its most important features. Our model uses a max-pooling layer (see Sect. 3.4), which means we keep the highest value out of a group of cells in the feature map. Then, the pooled feature map is flattened into a vector used to compute the final output.

3.1.2 LSTM

First proposed by S. Hochreiter and J. Schmidhuber in 1997 [14], Long Short-term Memory (LSTM) has since become one of the most widely used recurrent neural network architecture for time series data, where the processing of single data points is not sufficient and making predictions require looking at sequences of data. Figure 2 illustrates the principle of LSTM cells. There is a solid connection between the previous cell and the current cell in an LSTM layer (two values from the last cell are fed to the current cell) with four gates (functions in rectangular boxes, *sigma* denoting the sigmoid function) in each cell. It determines what part of memory to keep and discard, allowing it to use that memory to compute the output (with addition and dot multiplication) effectively.

Because of the nature of the problem and of the characteristics discussed above, the authors have decided to use CNN and LSTM layers for the model. The approach is far from the novel; previous works have published similar models with near-perfect accuracy in experiments. However, many such works are concerned with theoretical accuracy and lack validation against another dataset obtained from a network.

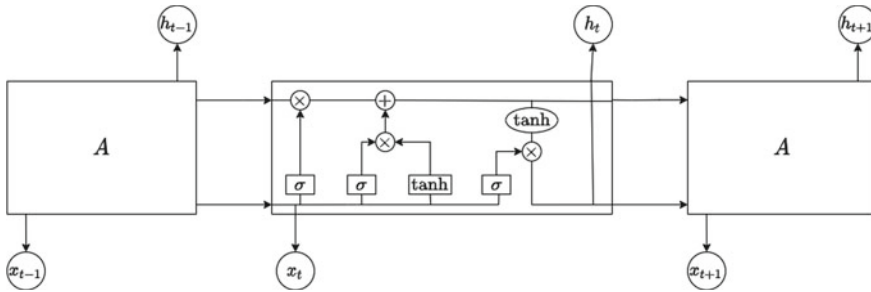


Fig. 2 Diagram of LSTM cells

3.1.3 Feature Selection

To make the dataset lighter and at the same time eliminate features that have the same effect on the final result, feature selection using Pearson correlation coefficients is usually applied. In this approach, after the correlation matrix has been calculated (with the coefficient between two features x and y computed as $r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}}$), one feature from a pair with a high correlation coefficient (above a predetermined threshold) is eliminated (because they supposedly have the same effect on the final prediction). In this study, the threshold is set to 0.8.

3.2 Network Design

In this research, a network was set up to simulate the Windows domain-based systems that target organizations typically use (see Fig. 3). A server acts as the domain controller for the whole domain, which also includes user machines. A soft firewall, which is an Ubuntu computer, stands between this intranet and the Internet. The IDS built in this research resides on this firewall, together with the open-source tools Zeek [15] and Argus [16], whose main capabilities are network monitoring, packet capturing and data gathering. On the other side of the firewall, a honeynet is set up using T-Pot [17] so that once a potential attack has been detected, all future traffic from that source will be directed to this subnetwork.

3.3 Workflow

In this section, we describe one iteration in the workflow of the system. More details are provided in Sect. 3.4.

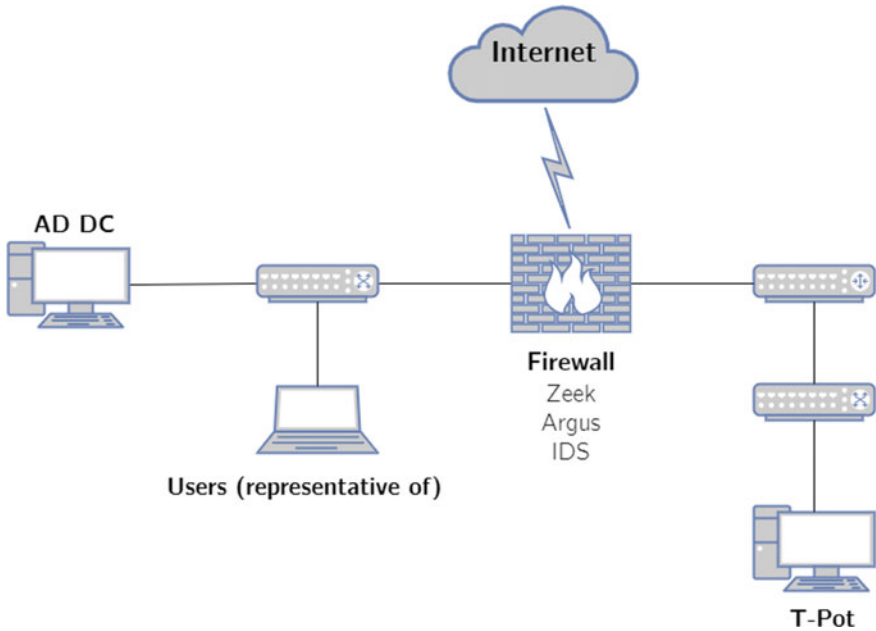
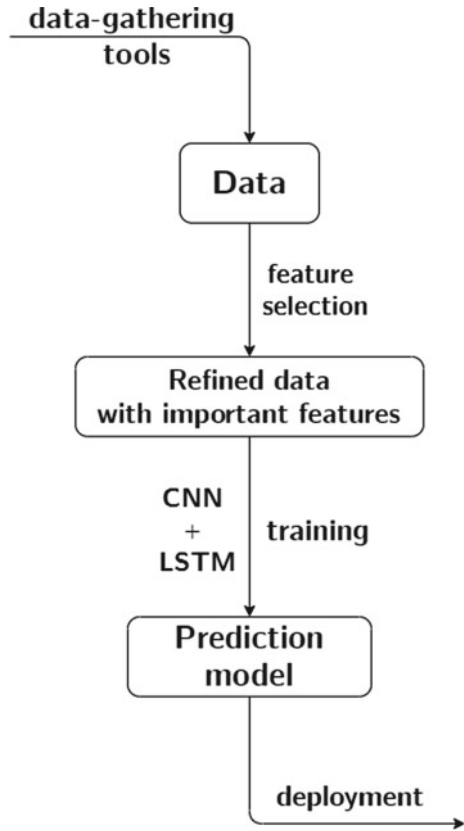


Fig. 3 Diagram of the network

1. On the firewall, Zeek and Argus monitor the traffic and collect data about the packets. Every 3 s, a script we have additionally written will take the output files of these tools, parse the data and upload it to a MySQL database. We chose the 3-second interval not to overload the processing capabilities of the IDS and not to affect the alerts' real-timeliness negatively.
2. The script will export the data table from MySQL as a .csv file and process it so that the file format matches the format used during training. The script then sends this file to the prediction script.
3. The prediction script accepts the input processes the data just as done with the training dataset, and makes predictions. There are two modes of prediction—binary and multi-class (default). Based on the predictions, the script will decide on whether to raise alerts. To compensate for the less-than-impressive accuracy of the model, we do not raise an alert right after the model detects a possible attack. Instead, we will wait for three packets from the same IP address within 1 s to be flagged before deciding.
4. If it is decided that an alert is raised, the script will print the alert to the terminal, store it in a log file, take note of the IP address of the packet that raised the alert, and use `iptables` on the Ubuntu-based firewall to forward all incoming packets from that IP address to the honeynet.

Fig. 4 Workflow for the model training module



3.4 Model Training for the IDS

Figure 4 describes the overall model training process with the steps as follow.

3.4.1 Dataset and Data Pre-processing

The dataset used for training the IDS is the UNSW-NB15.¹ The training and testing datasets contain 175,341 and 82,332 rows, respectively. For each row (corresponding to one packet), there are 45 features, two of which are the labels to be predicted—one being a Yes/No answer to the question “Is this packet from an attack?,” the other indicating the specific type of attack among nine common categories: Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, and Worms (plus the “Normal” category). The remaining 42 features are mainly packet-, traffic-

¹ <https://research.unsw.edu.au/projects/unsw-nb15-dataset>, accessed on 01 July 2021.

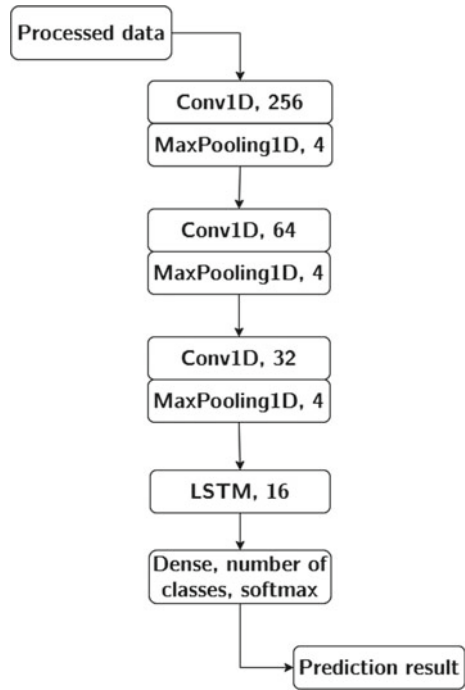
and connection-related. However, the dataset does not include source IP addresses, which makes forwarding using `iptables` impossible. Therefore, during our implementation in the system, we add one column to the end to store source IP addresses. This column is not used during training, testing, or making predictions. For more details about the dataset, readers are referred to [18–22].

However, we also performed some pre-processing to the data to increase accuracy and reduce training and prediction time. First, we ordinal-encoded nominal columns `proto`, `service` and `state` to make them all numerical. Because there is no natural ordering among the different values in each column, we made observations based on the popularity of the protocols/services/states in real-world and in the datasets (and thus the likelihood of them being utilized for attacks) and encoded them accordingly (see Table 1). Features are then normalized into the [0, 1] range to avoid biases toward any of them, especially those with inherently large values. Then, we carried out feature selection with several techniques and found the best results with Pearson correlation when we eliminated one feature in any pair with a correlation coefficient exceeding 0.8, leaving us with 27 features.

Table 1 Encoded values for `proto`, `service` and `state` columns

Raw value	% of Attacks in training	Encoded value	% of attacks in testing
proto			
udp	41.36	5	47.03
tcp	34.21	4	33.63
unas	10.13	3	7.75
ospf	2.12	2	1.41
sctp	0.96	1	0.71
Others	≤ 0.252 Each	0	≤ 0.212 Each
service			
-	48.31	5	43.63
dns	33.35	4	40.37
http	11.21	3	9.43
smtp	2.92	2	2.68
ftp	1.85	2	1.75
ftp-data	1.21	1	0.99
pop3	0.92	1	0.93
Others	≤ 0.079 Each	0	≤ 0.067 Each
state			
INT	64.15	5	65.47
FIN	34.06	5	33.46
REQ	0.89	2	0.30
CON	0.88	2	0.77
RST	0.01	1	0
Others	0	0	≤ 0.005 Each

Fig. 5 The proposed model used for training the IDS



3.4.2 Model Selection

As discussed in Sect. 2, multiple models, most complex, can achieve very high results against this dataset. However, when such a model is used to make real-time predictions in a network, several considerations must be made. The most notable among them is the trade-off between accuracy and performance. Generally, more complex networks with more layers are likely to produce a better result but slower than a simpler one. Therefore, as we were going through the trial-and-error process, we decided on a moderately simple model (see Fig. 5) with reasonable accuracy (see Sect. 4). However, because of this lower accuracy than typically seen in other models, we also had to keep certain reservations upon receiving prediction results from the model (see Sect. 3.3).

4 Results

Figure 6 is the Pearson correlation matrix among the original 47 features. Again, we can see some areas with bold blue cells—these features statistically have the same effect on the final prediction, so one from each pair is eliminated to simplify the computations. The correlation matrix among the remaining 27 features

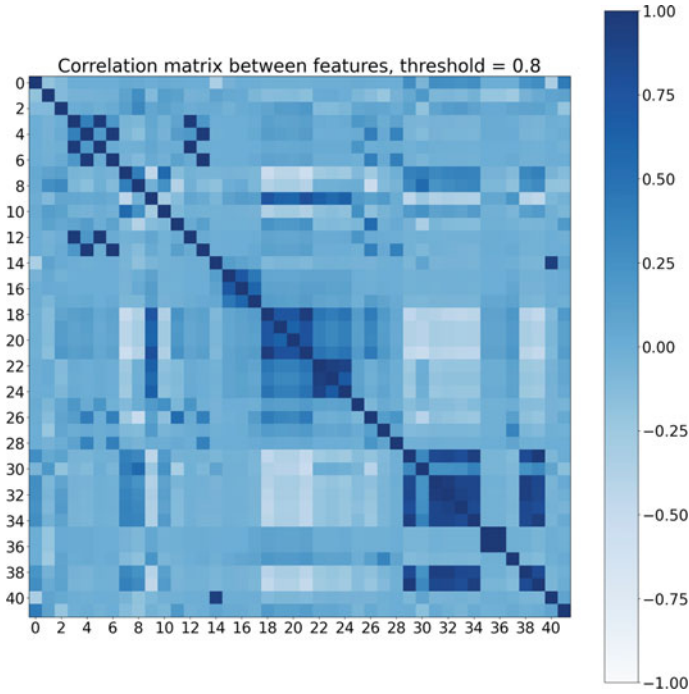


Fig. 6 Correlation matrix among original 47 features

is shown in Fig. 7. Here is the list of the 27 extracted features: dur, proto, service, state, spkts, dbytes, rate, sttl, dttl, sload, dloss, sinpkt, dinpkt, sjit, djit, swin, stcpb, dwin, ackdat, smean, dmean, trans_depth, response_body_len, ct_srv_src, ct_dst_src_ltm, ct_ftp_cmd, is_sm_ips_ports.

Our best accuracies for binary (Fig. 8) and multi-class (Fig. 10), given the specific requirements of this problem, are shown below. The blue lines are accurate against the training dataset, while the orange ones are against the testing dataset. We observe that the accuracy against the testing set becomes stable or fluctuates within a small range approximately after 120–150 epochs. For binary classification, the accuracy against the training set easily exceeded 0.96, while the figure for the testing set is around 0.89. For circumstances where the correct type of attack is not of primary importance (our case is one example; the first course of action when a possible attack is detected is to forward traffic coming from the source IP address to the honeynet regardless of the attack type), this along with our measures discussed in Sect. 3.3, should be enough for the IDS to be reliable most of the time (Fig. 9).

The confusion matrix for binary classification also looks good, with 0.97 and 0.86 true negative and true positive, respectively (see Fig. 9). The accuracy for multi-class classification is also acceptable: 0.87 against the training set and 0.74 against the

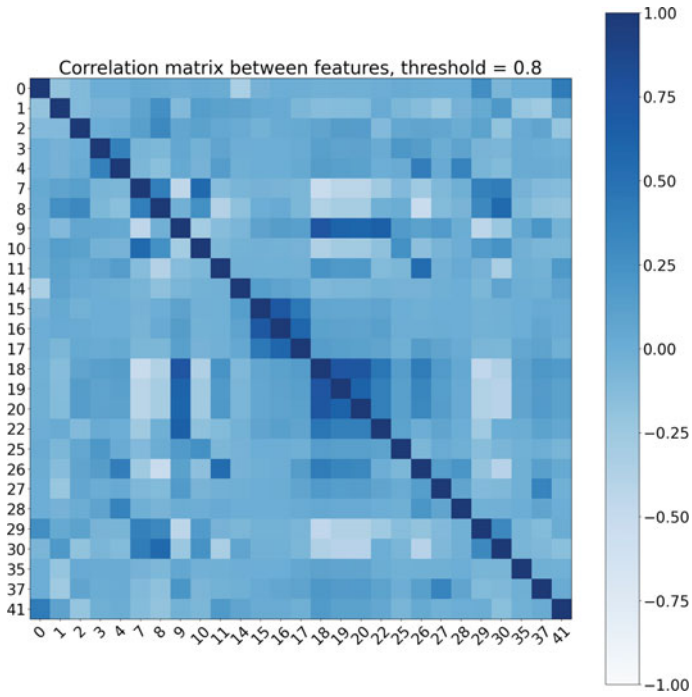
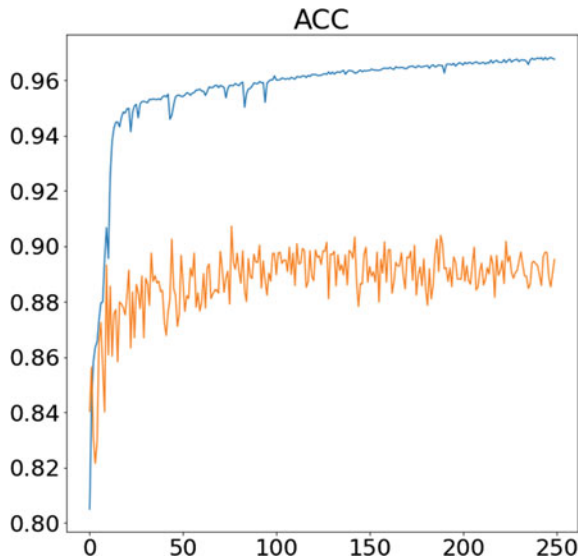


Fig. 7 Correlation matrix among remaining 27 features

Fig. 8 Accuracy for binary classification



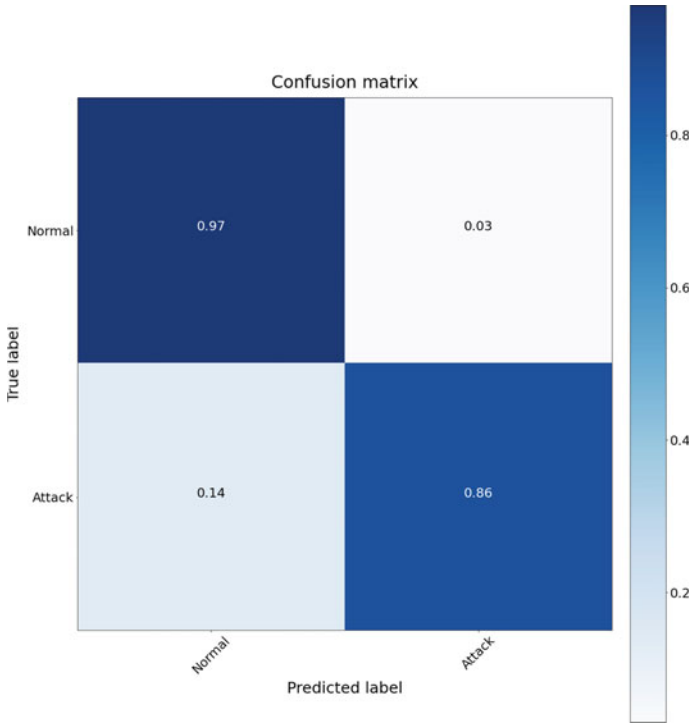
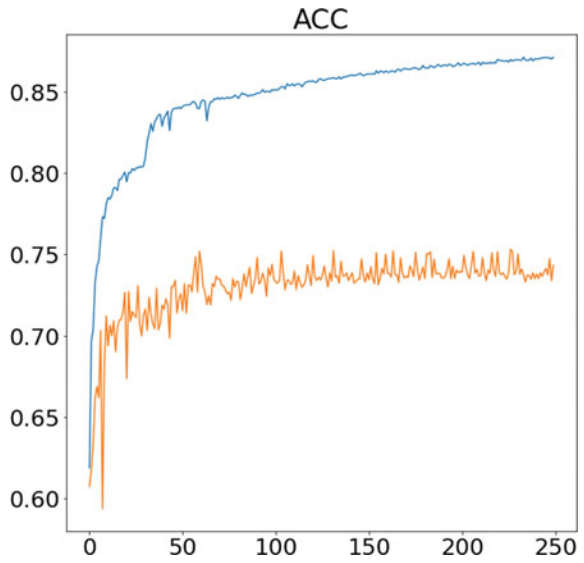


Fig. 9 Confusion matrix for binary classification

Fig. 10 Accuracy for multi-class classification



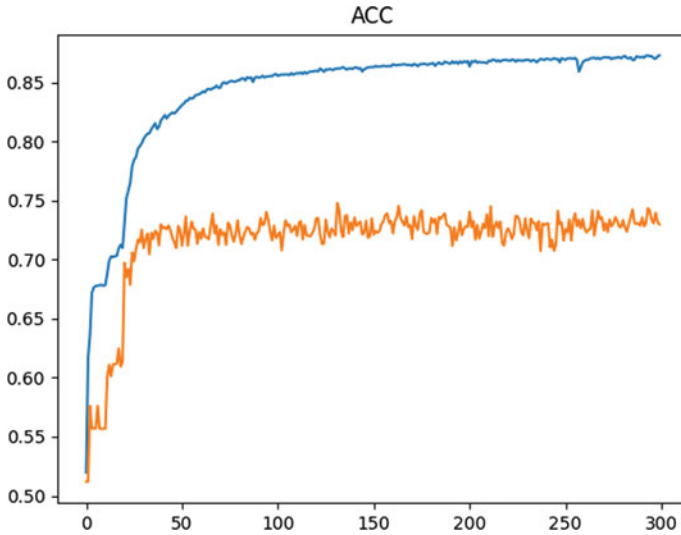


Fig. 11 Accuracy for multi-class classification on full feature set, same model

```
2021-Aug-04 12:04:05 ALERT!!! Possible attack. Possible type: Exploits. Traffic from 192.168.141.192 redirected!  
2021-Aug-04 12:04:05 ALERT!!! Possible attack. Possible type: Reconnaissance. Traffic from 192.168.141.192 redirected!  
2021-Aug-04 12:04:05 ALERT!!! Possible attack. Possible type: Fuzzers. Traffic from 192.168.141.192 redirected!  
2021-Aug-04 12:04:06 ALERT!!! Possible attack. Possible type: DoS. Traffic from 192.168.141.192 redirected!  
2021-Aug-04 12:04:06 ALERT!!! Possible attack. Possible type: Generic. Traffic from 192.168.141.192 redirected!
```

Fig. 12 Alerts on the terminal

testing set, given that packet header and flow statistics are not so clear an indicator of the possible type of attack. These results are comparable with the DNN approach on the same dataset performed by R. Vinayakumar et al. (0.65–0.75) [23]. We also find these figures compare with those obtained, while using the complete set of features (without feature selection) (see Fig. 11), which means our feature selection method is successful in reducing the computational cost, while maintaining the same level of accuracy.

When an alert is raised, the administrator will see things like Fig. 12 in the terminal (this sample is obtained after we conducted multiple attacks at the machine), after all, actions listed in Sect. 3.3 have been performed.

5 Conclusions and Future Works

For a DL-IDS to work in a production environment, we must perform processing to the data to make it lighter, select a model that strikes the right balance between accuracy and performance, and compensate for the compromises. Although this research has successfully demonstrated a working solution to this problem, there

are still some limitations: The dataset, which dates back to 2015, is not among the newest. Therefore, it may miss out on some current attack types or contain features that later researchers found necessary. Because of limited computing resources, the authors have not set up a more extensive network or perform more complex types of attacks from outside the network for testing purposes. Finally, the system sometimes raises a false positive alert because of the accuracy level, although not too often.

In future works, we would hope to try other deep learning algorithms to improve the model's accuracy, while still allowing it to run fast enough. In addition, the authors would train new models with more current datasets, particularly CSE-CIC-IDS, to see if the model proposed is ready for use in production. Should conditions permit, a more extensive system will be set up to simulate systems of medium-sized organizations, and more varied and complicated tests should also be performed on the model and the system. Finally, we would also incorporate the novel content-based approach to our IDS module to detect more sophisticated types of attacks.

References

1. Cost of a Data Breach Report 2021, IBM, USA, July 2021. Accessed: 5 Aug 2021 [Online]. Available: <https://www.ibm.com/security/data-breach>
2. N. Moustafa, Designing an online and reliable statistical anomaly detection framework for dealing with large high-speed network traffic. M.S. thesis, School of Engineering and IT, UNSW, Canberra, Australia (2017)
3. N. Shone, T.N. Ngoc, V.D. Phai, Q. Shi, A deep learning approach to network intrusion detection. *IEEE Trans. Emerg. Top. Comput. Intell.* **2**(1), 41–50 (2018). <https://doi.org/10.1109/TETCI.2017.2772792>
4. S. Otoum, B. Kantarci, H.T. Mouftah, On the feasibility of deep learning in sensor network intrusion detection. *IEEE Networking Lett.* **1**(2), 68–71 (2019). <https://doi.org/10.1109/LNET.2019.2901792>
5. M. Tavallaei, E. Bagheri, W. Lu, A.A. Ghorbani, A detailed analysis of the KDD CUP 99 data set, in *IEEE Symposium on Computational Intelligence for Security and Defense Applications*, vol. 2009 (2009), pp. 1–6. <https://doi.org/10.1109/CISDA.2009.5356528>
6. Z. Chiba, N. Abghour, K. Moussaid, A. Elomr, M. Rida, Intelligent approach to build a deep neural network-based IDS for cloud environment using a combination of machine learning algorithms, in *Computers & Security*, vol. 86, Sept 2019, pp. 291–317. <https://doi.org/10.1016/j.cose.2019.06.013>
7. S.A. Azwari, H. Turabieh, Intrusion detection using deep learning long short-term memory with wrapper feature selection method. *Int. J. Adv. Comput. Sci. Appl. (IJACSA)* **12**(3) (2021). <https://doi.org/10.14569/IJACSA.2021.0120366>
8. O. Faker, E. Dogdu, Intrusion detection using big data and deep learning techniques, in *Proceedings of the 2019 ACM Southeast Conference*, Apr 2019, pp. 86–93. <https://doi.org/10.1145/3299815.3314439>
9. M. Aloqaily, S. Otoum, I.A. Ridhawi, Y. Jararweh, An intrusion detection system for connected vehicles in smart cities. *Ad Hoc Networks* **90** (2019). <https://doi.org/10.1016/j.adhoc.2019.02.001>
10. Y. Liu, S. Liu, X. Zhao, Intrusion detection algorithm based on convolutional neural network. *Beijing Ligong Daxue Xuebao/Trans. Beijing Inst. Technol.* **37**(12), 1271–1275 (2017). <https://doi.org/10.15918/j.tbit1001-0645.2017.12.011>

11. W. Wang et al., HAST-IDS: learning hierarchical spatial-temporal features using deep neural networks to improve intrusion detection. *IEEE Access* **6**, 1792–1806 (2018). <https://doi.org/10.1109/ACCESS.2017.2780250>
12. M. Soltani, M.J. Siavoshani, A.H. Jahangir, A Content-Based Deep Intrusion Detection System. arXiv preprint [arXiv:2001.05009](https://arxiv.org/abs/2001.05009)
13. A. Kim, M. Park, D.H. Lee, AI-IDS: application of deep learning to real-time web intrusion detection. *IEEE Access* **8**, 70245–70261 (2020). <https://doi.org/10.1109/ACCESS.2020.2986882>
14. S. Hochreiter, J. Schmidhuber, Long short-term memory. *Neural Comput.* **9**(8), 1735–1780 (1997). <https://doi.org/10.1162/neco.1997.9.8.1735>
15. Zeek, Version 4.0.3. The Zeek Project. Accessed: July 2021 [Online]. Available: <https://zeek.org/get-zeek/>
16. Argus, Version 3.0.8.2. QoSient, LLC. Accessed: July 2021 [Online]. Available: <https://openargus.org/>
17. T-Pot, Version 20.06. Accessed: July 2021 [Online]. Available: <https://github.com/telekom-security/tpotce>
18. N. Moustafa, J. Slay, UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set), in *2015 Military Communications and Information Systems Conference (MilCIS)*, Nov 2015, pp. 1–6. <https://doi.org/10.1109/MilCIS.2015.7348942>
19. N. Moustafa, J. Slay, The evaluation of network anomaly detection systems: statistical analysis of the UNSW-NB15 dataset and the comparison with the KDD99 dataset. *Inf. Secur. J. Glob. Perspect.* **25**(1–3), 18–31 (2016). <https://doi.org/10.1080/19393555.2015.1125974>
20. N. Moustafa et al., Novel geometric area analysis technique for anomaly detection using trapezoidal area estimation on large-scale networks. *IEEE Trans. Big Data* **5**(4), 481–494 (2017), Dec 2019. <https://doi.org/10.1109/TBDDATA.2017.2715166>
21. N. Moustafa et al., Big data analytics for intrusion detection system: statistical decision-making using finite Dirichlet mixture models, in *Data Analytics and Decision Support for Cybersecurity* (Springer, Cham, Switzerland, 2017), pp. 127–156. https://doi.org/10.1007/978-3-319-59439-2_5
22. S. Mohanad, S. Layeghy, N. Moustafa, M. Portmann, NetFlow datasets for machine learning-based network intrusion detection systems, in *Big Data Technologies and Applications: 10th EAI International Conference, BDTA 2020, and 13th EAI International Conference on Wireless Internet, WiCON 2020, Virtual Event, Proceedings*, 11 Dec 2020 (Springer Nature, Berlin), p. 117. https://doi.org/10.1007/978-3-030-72802-1_9
23. R. Vinayakumar, M. Alazab, K.P. Soman, P. Poornachandran, A. Al-Nemrat, S. Venkatraman, Deep learning approach for intelligent intrusion detection system. *IEEE Access* **7**, 41525–41550 (2019). <https://doi.org/10.1109/ACCESS.2019.2895334>

Author Index

A

Abdul Rahiman, M., 139
Adarsh, S., 107
Aditya, A., 771
Agnihotri, Ankita, 169
Anand, Adithya, 373
Anjali, T., 1, 107, 267
Anjum, Mohd, 687
Anusree, L., 139
Anzar, N. S., 771
Asandi, Deepika, 89
Aswini, J., 181
Avi, Shuva Dasgupta, 509

B

Baby Shalini, V., 481
Balajee, R. M., 51, 61, 73
Balamurugan, K. S., 227
Bhairavi, R., 241
Bharati, Vishal, 781
Bhatnagar, Priyadarshini, 463
Bhavathankar, Prasenjit, 321

C

Chairma Lakshmi, K. R., 631, 747
Chanalya, Nikita, 1
Chandran, Vishwaak, 107
Chavan, Sudhanshu, 321

D

Darji, Miss Dhara N., 521
Dave, Jaivik, 421
Debnath, Narayan C., 349

Deepika, K., 655
Deshpande, Kiran, 607
Devan, Swetha V., 641
Devisetty, Vijay Kumar, 89
Dharan, Nidhin S., 203
Dineshan, Aathira, 267
Diwakar, Manoj Kumar, 169
Dudyala, Anil Kumar, 721
Duong, Dong Hai, 899
Dutta, Ritesh, 851

E

Endurthi, Anjaneyulu, 701

F

Fatima Ansari, Sana, 321

G

Ganesh Kumar, S., 595
Ganesh, Jishnu, 267
Ganesh, Siddarth, 373
Garg, Pramika, 281
Gedela, Vamsy Vivek, 89
Geetha Lekshmy, V., 39
Govindarajan, J., 449
Gowtham, R., 153, 203
Gupta, Kalpesh, 267
Gupta, Manu, 403

H

Hameed, Shahul T. A., 809
Harikrishnan, J., 267

Harikrishnan, P. S., 39
 Harshini Poojaa, K., 595
 Hatti, Daneshwari I., 433
 Hemanandhini, I. G., 711
 Hennadii, Khudov, 827
 Hiremath, Sujatha, 739
 Ho, Van The, 899
 Hridya Krishna, R., 1
 Hussain, Abir, 721

I

Igor, Butko, 827
 Irina, Khizhnyak, 827
 Iryna, Yuzova, 827
 Ismail, Muhammed, 771

J

Jagiri, Gayathri, 701
 Jajodia, Babita, 333
 Jakhodia, Simran, 333
 Jayanthi Kannan, M. K., 51, 61, 73
 Jayaraman, Poonthugilan, 575
 Jha, Avish, 281
 Jhansi Sri Latha, A., 227
 Johnson, Shanoop, 153
 Jose, Nithil, 449

K

Kala, L., 809
 Kanavalli, Anita, 305
 Kannimoola, Jinesh M., 29
 Karthigha, M., 759
 Kaur, Jaspreet, 839
 Khade, Anindita, 859
 Khan, Nayeem Ahmad, 217
 Khare, Yash, 107
 Koppad, Deepali, 739
 Kotaprolu, Sai Smaran, 89
 Krishna, Aki Vamsi, 15
 Kumar, Arun, 531, 781, 793
 Kumar, Sandeep, 851
 Kumar, Sanjay, 395

L

Lakshmi Devi, P., 297
 Lakshmi, K. S., 641
 Lavenya, K., 673
 Luong, Huong Hoang, 899
 Luong, Luan Anh, 899

M

Maheswari, B., 181
 Mahto, Sanchit, 395
 Malipatil, Somashekhar, 297
 Manivannan, R., 561
 Manju, D., 879
 Megalingam, Rajesh Kannan, 89
 Meghadev, C., 1
 Monzur Rahaman, K. M., 509
 Muralidharan, V., 119
 Murali Mohan, V., 51, 61, 73

N

NagaSai Manojna, Ch., 227
 Nair, Amrita, 267
 Nair, Anand R., 153
 Nair, Varun, 29
 Najeeb, T., 771
 Nguyen, Hai Thanh, 899
 Nguyen, Phong Cao, 899
 Nguyen, Thinh Truong, 899

O

Oleksandr, Makoveichuk, 827

P

Padma Ashalesha, Ch. N. L., 227
 Padmavathi, S., 421
 Padmavathy, C., 711, 759
 Pahareeya, Jankisharan, 721
 Pandey, Ravikant, 839
 Pandey, Sudhakar, 395
 Parikh, Satyen M., 521
 Patel, Archana, 349
 Patel, Hiral R., 521
 Pavithra, R., 759
 Pillai, Smriti, 373
 Podder, Prajoy, 493
 Poorna, B. R., 771
 Prabha, D., 181
 Prabhu, E., 15
 Pramod, V. R., 809
 Praveena, B., 631, 747
 Praveen, K., 361
 Praveen, P., 889
 Pravija, Danda, 395
 Premkumar, Aswathi, 1
 Priya, Gadhiraaju Hari, 403
 Pyarapu, Aparna, 701

R

Rajaguru, Harikumar, 257
 Rajarajeswari, S., 305
 Ramesh, Abhijit, 107
 Ramya, G. R., 463
 Rana, Waseem, 839
 Rao, Madhuri, 607
 Rashedul Arefin, Md., 509
 Ravikumar, Shwetha, 859
 Rawshan Habib, Md., 509
 Reddy, Nandikonda Archana, 403
 Renuka Prasad, B., 655
 Rizwan, M., 541
 Rouf, Mohammad Abdur, 493
 Rubayet Hossain, A. M., 509
 Rudravaram, Gaurav, 89

S

Sahoo, Abinash, 169
 Sai Jishnu, M., 373
 Sammulal, P., 879
 Sanjana, A., 403
 Sannasi Chakravarthy, S. R., 257
 Sarosh Umar, M., 687
 Sathya, R., 561
 Seetha, M., 879
 Shahab, Sana, 687
 Shahariar Parvez, A. H. M., 493
 Shahnewaz Tanvir, Md, 509
 Sheshikala, M., 889
 Shobitha, M., 383
 Shreyashree, S., 305
 Shukla, Kumar A., 281
 Siddiqui, Atique, 321
 Sidharth, Prakash R., 383
 Siji Rani, S., 1
 Singh, Divyanshu, 333
 Singh, Pranjal, 107
 Singh, Upma, 541
 Sirajul Islam, Md., 493
 Siva Subramanian, R., 181
 Sonkar, Nidhi, 395

Sreesruthi, P. K., 383
 Sriram, Padmamala, 267
 Sudha, Gnanou Florence, 241
 Suhan, Ahmed Yousuf, 509
 Sunagar, Pramod, 305
 Surekha, P. S., 297
 Suriya Prasath, S., 859
 Sutagundar, Ashok V., 433
 Swaminathan, Jayaraman, 383
 Swathi, B., 889

T

Taraka Rama Mokshagna Teja, M., 361
 Thara, S., 373

U

Umavathi, M., 673
 Usha, A., 575

V

Vadher, Abhishek, 509
 Vaidehi, K., 561
 Varma, Utkrist Arvind, 1
 Varun Raj, P., 383
 Veerasamy, Sevagen, 107
 Vennam, SaiSuma, 701
 Verghese, Jerin, 851
 Vijayalakshmi, S., 631, 747
 Vijayalakshmi, V., 119
 Vijay Anand, K., 631, 747
 Vinutha, K., 575
 Vishnu, P. A., 39
 Vladyslav, Khudov, 827
 Vohra, Rajan, 721

Y

Yadav, Abhishek Kumar, 793
 Yadav, Ashok, 531
 Yuriy, Solomonenko, 827