

Ravindra Hegadi
Gaurav Gupta
KC Santosh (Eds.)

Communications in Computer and Information Science

2621

Applied Artificial Intelligence

First International Conference, 2AI 2024
Solan, India, July 2–4, 2024
Proceedings



Springer

Communications in Computer and Information Science

2621

Series Editors

Gang Li , *School of Information Technology, Deakin University, Burwood, VIC, Australia*

Joaquim Filipe , *Polytechnic Institute of Setúbal, Setúbal, Portugal*

Zhiwei Xu, *Chinese Academy of Sciences, Beijing, China*

Rationale

The CCIS series is devoted to the publication of proceedings of computer science conferences. Its aim is to efficiently disseminate original research results in informatics in printed and electronic form. While the focus is on publication of peer-reviewed full papers presenting mature work, inclusion of reviewed short papers reporting on work in progress is welcome, too. Besides globally relevant meetings with internationally representative program committees guaranteeing a strict peer-reviewing and paper selection process, conferences run by societies or of high regional or national relevance are also considered for publication.

Topics

The topical scope of CCIS spans the entire spectrum of informatics ranging from foundational topics in the theory of computing to information and communications science and technology and a broad variety of interdisciplinary application fields.

Information for Volume Editors and Authors

Publication in CCIS is free of charge. No royalties are paid, however, we offer registered conference participants temporary free access to the online version of the conference proceedings on SpringerLink (<http://link.springer.com>) by means of an http referrer from the conference website and/or a number of complimentary printed copies, as specified in the official acceptance email of the event.

CCIS proceedings can be published in time for distribution at conferences or as post-proceedings, and delivered in the form of printed books and/or electronically as USBs and/or e-content licenses for accessing proceedings at SpringerLink. Furthermore, CCIS proceedings are included in the CCIS electronic book series hosted in the SpringerLink digital library at <http://link.springer.com/bookseries/7899>. Conferences publishing in CCIS are allowed to use our online conference service (Meteor) for managing the whole proceedings lifecycle (from submission and reviewing to preparing for publication) free of charge.

Publication process

The language of publication is exclusively English. Authors publishing in CCIS have to sign the Springer CCIS copyright transfer form, however, they are free to use their material published in CCIS for substantially changed, more elaborate subsequent publications elsewhere. For the preparation of the camera-ready papers/files, authors have to strictly adhere to the Springer CCIS Authors' Instructions and are strongly encouraged to use the CCIS LaTeX style files or templates.

Abstracting/Indexing

CCIS is abstracted/indexed in DBLP, Google Scholar, EI-Compendex, Mathematical Reviews, SCImago, Scopus. CCIS volumes are also submitted for the inclusion in ISI Proceedings.

How to start

To start the evaluation of your proposal for inclusion in the CCIS series, please send an e-mail to ccis@springer.com

Ravindra Hegadi · Gaurav Gupta · KC Santosh
Editors

Applied Artificial Intelligence

First International Conference, 2AI 2024
Solan, India, July 2–4, 2024
Proceedings

Editors

Ravindra Hegadi 
Central University of Karnataka
Kalaburagi, Karnataka, India

Gaurav Gupta 
Yogananda School of AI
Shoolini University
Solan, Himachal Pradesh, India

KC Santosh 
AI Lab
University of South Dakota
Vermillion, SD, USA

ISSN 1865-0929 ISSN 1865-0937 (electronic)
Communications in Computer and Information Science
ISBN 978-3-032-00792-6 ISBN 978-3-032-00793-3 (eBook)
<https://doi.org/10.1007/978-3-032-00793-3>

© The Editor(s) (if applicable) and The Author(s), under exclusive license
to Springer Nature Switzerland AG 2026

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 Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

If disposing of this product, please recycle the paper.

Preface

We are delighted to present this compilation of research papers in Springer's Communications in Computer and Information Science (CCIS) series from the first International Conference on Applied Artificial Intelligence (2AI 2024) (URL: <https://applied-ai-conference.org/>). 2AI 2024 was hosted in-person at Shoolini University, Solan, India, from July 02–04, 2024, in collaboration with the Applied AI Research Lab (2AI) at the University of South Dakota, USA, and Central University of Karnataka, India.

As highlighted in the call for papers, 2AI 2024 attracted a wealth of current and recent research on applications of artificial intelligence in education, healthcare, agriculture, business and finance, energy, defence and information security and other related sectors. With 151 submissions, we carefully selected 30 papers for presentation during the conference. A plagiarism check was applied to all the articles, and articles with plagiarised content were rejected. We diligently reviewed submissions, and authors were asked to incorporate the recommendations of the reviewers to arrive at final decisions, resulting in an acceptance rate of 19.87%. Each selected/accepted paper underwent a rigorous evaluation process, with a minimum of three single-blind reviews per selection.

The conference provided a valuable platform that brought together research scientists, academics, and industry practitioners. The event encapsulated a wealth of innovative ideas. We thank all the Keynote speakers and other resource persons who served in different capacities and enriched the participants with their knowledge on the conference theme. During the conference, KC Santosh spearheaded an introductory speech on Sustainable AI solutions (for all) and Green Computing.

Our heartfelt thanks go out to all contributors who made 2AI 2024 a success.

July 2024

Ravindra Hegadi
Gaurav Gupta
KC Santosh

Organization

Chief Patron

Prem Kumar Khosla

Shoolini University, India

Patrons

Atul Khosla

Shoolini University, India

Vishal Anand

Shoolini University, India

Ashish Khosla

Shoolini University, India

Honorary Chairs

KC Santosh

University of South Dakota, USA

Virender Rihani

Shoolini University, India

General Chairs

Ravindra Hegadi

Central University of Karnataka, India

Gaurav Gupta

Shoolini University, India

Pankaj Vaidya

Shoolini University, India

Ashutosh K. Singh

IIIT Bhopal, India

Program Chairs

Manju Khari

Jawaharlal Nehru University, India

M. Tanveer

IIT Indore, India

Mufti Mahmud

Nottingham Trent University, UK

Surbhi Bhatia

University of Salford, UK

Rodrigue Rizk

University of South Dakota, USA

Deepak Kumar

Shoolini University, India

Harish Garg

Thapar Institute, India

Theme Chairs

AI for Healthcare

Vinaytosh Mishra
Pethuru Raj Chelliah
Nibaran Das

Gulf Medical University, UAE
Reliance Jio Platform, India
Jadavpur University, India

AI for Agriculture

Shakir Khan
K. K. Patel

IMIS Islamic University, Saudi Arabia
Charotar University of Science and Technology,
India

AI for Business and Finance

Rajeev Singh

Banasthali Vidyapith, India

AI for Energy

S. S. Chandel
Yashwant Singh

Shoolini University, India
Central University of Jammu, India

AI for Defence and Information Security

Sonia
Ashish Ghosh

Shoolini University, India
Indian Statistical Institute, India

AI for Education

Janmenjoy Nayak
Anshuman Shastri
Shankar Mali

MSCB University, India
Vanasthali Vidhyapith, India
MIT World Peace University, India

National Advisory Committee

Rajeev Agrawal
Baijnath Kaushik
Rahul Katarya
Rashmi Gupta

LIET, India
Shri Mata Vaishno Devi University, India
Delhi Technological University, India
NSUT, India

Pritee Khanna
Devesh Pratap Singh

IIITDM Jabalpur, India
Graphic Era Deemed University, India

International Advisory Committee

Chandan Pal Singh	SKPU, Kazakastan
Chitresh Sharma	L'Oréal, France
D. S. Guru	University of Mysore, India
Longwei Wang	University of South Dakota, USA
Abdelkrim Haqiq	Hassan I University, Morocco
Anas Abou el Kalam	Cadi Ayyad University, Morocco
Shashi Kant Gupta	Eudoxia Research University, USA
Arvind Bhardwaj	Capgemini, USA

Technical Program Committee

Dhananjay Deshpande	MBAESG, India
Dhananjay Upasani	MIT School of Engineering, India
Diego Liberati	Politecnico di Milano, Italy
Digvijay Singh	Uttaranchal University, India
Dileep M. R.	Nitte Meenakshi Institute of Technology, India
Dipak Sah	IITISM Dhanbad, India
Abdullah Kaleem	MPGI SOE, India
Prema Akkasaligar	KLE Technological University, India
V. Thakare	SGB Amaravati University, India
Ganesh Janvale	College of Agricultural Biotechnology, Loni, India
Ganga Holi	K S Institute of Technology, India
Gaurav Gupta	Shoolini University, India
Gireesh C. N.	BMS Institute of Technology and Management, India
Girijalaxmi G.	Central University of Karnataka, India
Girish Chowdhary	SRTM University, India
Gnaneswara Nitta	DRK Institute of Science and Technology, India
Harshavardhana D.	S.J.C. Institute of Technology, India
Harsimran Kaur	Chitkara University, India
Hema Patel	CHARUSAT, India
Himadri Mukherjee	West Bengal State University, India
Hoang-Anh Ngo	Woolcock Institute of Medical Research, Vietnam
Imad Ali Shah	University of Galway, Ireland

Ishani Sharma	Shoolini University, India
Jatinder Kumar	National Institute of Technology Kurukshetra, India
Jay Kumar Pandey	Shri Ramswaroop Memorial University, India
Jeetendra Kumar	Atal Bihari Vajpayee Vishwavidyalaya, India
Joana Sousa	University of Lisbon, Portugal
Kanubhai Patel	Charotar University of Science and Technology, India
Kapil Mehrotra	C-DAC, India
Kapil Sethi	Bahra University, India
Kaushik Roy	WBSU, India
Kavita Houde	Central University of Karnataka, India
Kiran Phalke	Solapur University, India
Kishu Gupta	National Sun Yat-sen University, Taiwan
Kritika Rana	Shoolini University, India
L. S. Sankaranarayanan	Google, USA
Lorenzo Putzu	University of Cagliari, Italy
M. Ravi	GFGC Raichur, India
Madhu Arora	Sri Balaji University, India
Mahadev Patil	Bharati Vidyapeeth Solapur, India
Maheswaran S	Kongu Engineering College, India
Mainak Sen	Techno India University, India
Mallikarjun Holi	University BDT College of Engineering, India
Mallikarjunaswamy M. S.	Sri Jayachamarajendra College of Engineering, India
Manjunath T. N.	BMS Institute of Technology and Management, India
Manjunatha Hiremath	Christ (Deemed to be University), India
Manohar Madgi	KLE Technological University, India
Manoj Patil	KBC North Maharashtra University, India
Masud Rana Rashel	University of Évora, Portugal
Maya Thapa	Shoolini University, India
Meena Tiwari	Shri Ram Institute of Science & Technology, India
Meenakshi Bhardwaj	Amity University, India
Mian Ibad Ali Shah	University of Galway, Ireland
Mickael Coustaty	L3i laboratory, France
Mohammod N. Islam	University of Sheffield, UK
Mohan Rawat	Chameli Devi Group of Institutions, India
Mukesh Khandelwal	Govt. Mahila Engineering College Ajmer, India
Nagaraj Cholli	RV College of Engineering, India
Nagaratna P. Hegde	Vasavi College of Engineering, India
Nagsen Bansod	JSPM University, India

Nahid Sami	University of Derby, UK
Narottam Patel	VIT Bhopal, India
Navneet Kaur	Guru Nanak Dev Engineering College, India
Navneet Sharma	IIS Deemed to be University, India
Neha Singh	Illinois Institute of Technology, USA
Nikhil Thakur	Amity University, India
Nilay Vaidya	CMPICA, India
Parag Kaveri	Symbiosis Institute of Computer Studies & Research, India
Parag Tamhankar	Georgia State University, USA
Parbhat Gupta	SRM Institute of Science and Technology, India
Parminder Kaur	Guru Nanak Dev University, India
Pavithra M.	Vellore Institute of Technology, India
Peter Dolan	University of Minnesota, USA
Piyush Sewal	Shoolini University, India
Pooja Janse	Dr. Babasaheb Ambedkar Marathwada University, India
Pooja Nagpal	International School of Mgt. Excellence, India
Pooja S.	Galgotias University, India
Prabhakar C. J.	Kuvempu University, India
Pradeep Chintale	SEI Investment Company, USA
Prakash H. Unki	BLDE College, India
Prakash Hiremath	KLE Technological University, India
Pranav Kumar Singh	Central Institute of Technology Kokrajhar, India
Prashant Kumar	Chandigarh University, India
Prashant Patavardhan	KLS Gogte Institute of Technology, India
Pratima Manhas	MRIIRS, India
Rahul Kumar	BIT Meerut, India
Rajendra Hegadi	IIIT Dharwad, India
Rajesh Dhumal	Symbiosis Institute of Geoinformatics, India
Rajiv Avacharmal	New York Life Insurance Company, USA
Raksithaa S.	Christ (Deemed to be University), India
Ranit Kishore	Tula's Institute, India
Rashmi Soni	Dayananda Sagar Academy, India
Ravindra Hegadi	Central University of Karnataka, India
Ravindra Sharma	Swami Rama Himalayan University, India
Renjith V. Ravi	M.E.A. Engineering College, India
Rhytheema Dulloo	Hindustan Institute of Technology and Science, India
Rodrigo Nava	AstraZeneca, UK
Rodrigue Rizk	University of South Dakota, USA
Rohini Bhusnurmath	KSAW University, India

S. Patil	University of Agricultural Science, India
S. Pamulaparthivenkata	VillageMD, USA
Samson Anosh Babu	Chaitanya Bharathi Institute of Technology, India
Sangita Kakarwal	ICEEM, India
Sanjay Jain	SOET, India
Santosh Chowhan	Symbiosis Institute (Deemed University), India
Saraswathi S.	SS University, India
Satender Kumar	National Institute of Technology Kurukshetra, India
Satish Kolhe	North Maharashtra University, India
Shakeel Ahmed	CCSIT, India
Shallu Sehgal	Shoolini Institute, India
Shanmugapriya P.	Saranathan College of Engineering, India
Sharad Saxena	Thapar Institute of Engineering and Technology, India
Shashi Kant Gupta	Eudoxia Research University, India
Shashikant Patil	ATLAS SkillTech University, India
Shaveta Thakral	KJ College of Engineering & Mgt. Research, India
Shilpa Sthavarikmath	Central University of Karnataka, India
Shiva Murthy G.	Visvesvaraya Technological University, India
Shivanand Gornale	Rani Channamma University, India
Shrikant Mapari	SICSR, India
Shriram Raut	P A H Solapur University, India
Shubham Mahajan	Ajeenkya DY Patil University, India
Shweta Pandey	Uttaranchal University, India
Sivabalan Settu	Vignan University, India
Slimane Larabi	USTHB, Algeria
Smriti Bhandari	LF, India
Sopan Talekar	MVPS KAB Thakare College, India
Sumanth S.	Government College for Women Bangalore, India
Sumit Sharma	National Institute of Technology Kurukshetra, India
Sunil Nimbhore	Dr. Babasaheb Ambedkar Marathwada University, India
Swathi S.	Central University of Karnataka, India
Szilárd Vajda	Central Washington University, USA
Tejpal Sharma	Chandigarh Group of Colleges, India
T. Kartheeswaran	University of Vavuniya, Sri Lanka
Thungamani M.	University of Horticultural Sciences, Bagalkot, India
Umar Sada	Jamia Millia Islamia, India

Urmila Pol	Shivaji University, India
V. Malemath	KLE Dr. M S Sheshgiri College, India
V. Palanisamy	Sabaragamuwa University of Sri Lanka, Sri Lanka
Vijaya Kittu Manda	PBMEIT, India
Vinay T. R.	Ramaiah Institute of Technology, India
Vinaytosh Mishra	College of Healthcare Mgt. and Economics, UAE
Vineet Vishnoi	Shobhit University, India
V. S. Hallur	Angadi Institute of Technology and Mgt., India
Vivek Singh	Sharda University, India
Yousef Sharafi	Toosi University of Technology, Iran
Zakaria Rguibi	FST Settat, Morocco

Contents

AI for Health Care

Analysis of Feature Fusion in Deep Learning for Knee Osteoarthritis Severity Grade Classification	3
<i>Sushma V. Chaugule, V. S. Malemath, and Shivanand Patil</i>	
Lung Cancer Detection and Analysis Using Machine Learning	15
<i>Sanskriti S. Joshi, Janhavi Bhagat, Ambarish Kulkarni, Shambhawi Sharma, and S. M. Mali</i>	
Machine Learning Based Depression Prediction Using Gradient Boosting Algorithm	21
<i>Mohith Manoharan Satheesh, R. Nagavibha, Mishbah Pinto Chandy, and N. Jayapandian</i>	
Patient Sickness Predictive System Using Chatbot	33
<i>Dappili Bharat Kumar Reddy, Vineet Dasari, C. Madhu Kumar, Mmihiraansh, Naga Venkata Yaswanth Lankadasu, Devendra Babu Pesarlanka, Ajay Sharma, and Shamneesh Sharma</i>	
EPC-CCM Based Saliency Equivalent Mechanism for Multimodal Medical Image	45
<i>Nalini Manogaran, Resmi R. Nair, Suresh Annamalai, Balamurugan Balusamy, Parimala Veluvali, and Sushama Tanwar</i>	
Healthcare Effective Diabetes Disease Prediction Using AI Based Techniques	62
<i>Gopal Kumar Thakur and Madan Mohan Tito Ayyalasomayajula</i>	
Manual and Automated Web-Based Diagnosis and Interpretation of Mammograms of Breast Cancer and Robust Analysis	73
<i>Nalini Manogaran, Suresh Annamalai, Koteeswaran Seerangan, Balamurugan Balusamy, Parimala Veluvali, and Sushama Tanwar</i>	
Social Media and Stress Detection: A Machine Learning Approach	87
<i>Rutuja Kamble, Piyush Borhade, Urjita Garud, and S. M. Mali</i>	

AI for Business and Finance

Computational Optimization of Image Feature Extraction Algorithms
on PySpark and Hadoop 101
Arshi Jamal and K. Ramesh

Sentiment Analysis of Political Party Unveiling Insights from Big Data 114
Mayuri Bapat, Shreya Karade, and Aniket S. Nagane

AI for Defense and Information Security

Securing Web Applications with Improved Multifactor Authentication
Using Browser-Based Geolocation 127
A. Yadav, K. Kumar, P. Pandey, M. Khari, and P. P. Singh

Human Activity Recognition Using Machine Learning 140
Meena Tiwari and Prateek Gupta

Predictive Analytics of Injection Attacks in Web Applications 154
U. Farjana and Gowrineni Harshitha Sai

Comparative Study of Vision-based Deep Learning Techniques for Human
Action Recognition in Videos 164
R. Divya Rani and C. J. Prabhakar

Person Identification Using Improved Dilated Convolutional Network
for Radar Signal Based Hand Gestures 177
*Helen Victoria, Deepa Natesan, Balamurugan Balusamy,
Shilpa Bade-Gite, Sushama, and Alka Agnihotri*

Multimodal Iris, Face and Fingerprint Biometric Fusion for Enhanced
Person Identification Using Hybrid Deep Learning 195
Sharad B. Jadhav, N. K. Deshmukh, K. A. Hambarde, and M. S. Darak

Adaptive Inertial Weight with Artificial Rabbit Optimization for Intrusion
Detection System in Cyber Security 207
*Pradeep Chintale, Anirudh Khanna, Madhavi Najana,
and Hrushikesh Deshmukh*

Deepfake Video Detection: A CSP-Dencenet and LSTM Based Approach 221
Ayush Bajpai and Ritu Garg

Neural Networks in Cybersecurity: Applications for Predictive Threat Modeling	234
<i>Ayisha Tabbassum, Arun Pandiyan Perumal, Anirudh Khanna, Pradeep Chintale, and Madhavi Najana</i>	
Weighted Extreme Gradient Boosting Based Cybersecurity Risk Assessment in Investment Banking and Financial Sector	249
<i>Pradeep Chintale, Hrushikesh Deshmukh, Anirudh Khanna, Ankur Mahida, and Madhavi Najana</i>	
Swarm Learning Based Federated Learning for Data Privacy in Cloud Environment	261
<i>Madhavi Najana, Arun Pandiyan Perumal, Pradeep Chintale, Laxminarayana Korada, Ankur Mahida, and Sri Harsha Vardhan Sanne</i>	
AI for Agriculture	
Detection of Tomato Leaf Disease Using Convolutional Neural Network	275
<i>Manohar Madgi, Vivek Verma, Prabhat K. Rai, Tsewang Norbu, and Vrashabh Kambayal</i>	
Pbstam: A Proactive Borewell Safety Using Teachable Arduino Model	286
<i>Sri Silpa Padmanabhuni, D. Durga Prasad, Perumalla Prasanna Siri, Repalli Vineetha, and Vejandla Pavithra Prasad</i>	
Waste Management Using Geo Special Processing Data	300
<i>D. Saisanthiya, Deepa Natesan, Balamurugan Balusamy, Deepali Rani Sahoo, Sushama, and Alka Agnihotri</i>	
Tomato Leaf Disease Detection Using Convolutional Neural Network	315
<i>Singh Trisha, Jaswanth Singh Kumar Lankadas, Satti Satya Reddy, Veera Venkata Varshith Nagubandi, Ajay Sharma, and Shamneesh Sharma</i>	
AI for Education	
Recognition of Characters on English Language Number Plate Using CNN	331
<i>Ravindra S. Hegadi, Ashutosh Mishra, and Kavita V. Houde</i>	
Artificial Intelligence Based Automatic Question Paper Generation Using Natural Language Processing	343
<i>Joyal Antony, George Bobby, Jowin Joshi, and N. Jayapandian</i>	

Exploring Information-Theoretic Measures for Feature Extraction
in Image Processing: A Comprehensive Analysis and Future Directions 356
Romisha and Shruti Vashist

Enhancing Text Classification in Natural Language Processing:
A Comparative Study of Transformer Models and the Potential
of Few-Shot Learning 366
Syed Hur Abbas, Gitanjali Wadhwa, and Sandra Hummel

A Modified NSGA-II Algorithm Using New Crowding Distance
and Chaotic OBL Techniques Applied to Multi-Object Engineering
Design Problems 381
Nastaran Ahmadi Ramezanloo, Yousef Sharafi, and Yashar Jebraeily

Author Index 395

AI for Health Care



Analysis of Feature Fusion in Deep Learning for Knee Osteoarthritis Severity Grade Classification

Sushma V. Chaugule¹ , V. S. Malemath¹ , and Shivanand Patil² 

¹ KLE Dr. M. S. Sheshgiri College of Engineering and Technology, Belagavi, Karnataka, India
vidhu_sun@yahoo.co.in

² SG Balekundri Institute of Technology, Belagavi, Karnataka, India

Abstract. Osteoarthritis is an infirmity marked by gradual degradation and loss of specialized connective tissue called articular cartilage, with symptoms that can be manageable but irreversible. Our methodology presents a holistic approach to locating and categorizing Knee Osteoarthritis (KOA) in the X-ray images, integrating advanced feature extraction, segmentation, and deep learning techniques. 5778 KOA-related images taken from the standard Osteoarthritis Initiative (OAI) database are subjected to preprocessing using Extreme Learning Machine AutoEncoder (ELM-AE) denoising and further resized, then enhanced to address noise and variability. OTSU's thresholding and morphological operations are applied in segmentation to isolate knee joint regions. Feature extraction encompasses region-based, Zernike, Haralick texture, and wavelet features, providing insights into knee OA characteristics. The classification process utilizes a Deep Convolutional Neural Network (DCNN) architecture, featuring convolutional and max-pooling layers for feature extraction, fully connected layers for decision making, and activation functions. Z-score and batch normalization are applied to enhance stability and convergence. Training and optimization use Adam optimization, with shallow fine-tuning enhancing efficiency. The novelty lies in the enhanced classification model, which encompasses multiple stages for preprocessing, segmentation, feature extraction, and classification, each refining final classification outcomes. This offers an innovative approach to KOA detection, leveraging advanced techniques, and has attained an accuracy of 96.31% for testing and 95.80% for validation. It holds promise for aiding medical practitioners in early KOA diagnosis and management.

Keywords: Feature Extraction · Knee Osteoarthritis · Feature Fusion · Deep Convolutional Neural Network · Classification · X-ray Images

1 Introduction

Osteoarthritis, a degenerative joint disorder, is a common ailment resulting in joint damage. Its typical symptoms include stiffness, pain, swelling, and joint dysfunction. It predominantly affects weight-bearing joints like the knees, hips, and feet, leading to reduced joint flexibility. Symptoms manifest gradually, exacerbating particularly in

individuals aged 50 and older. OA is characterized as cartilage deterioration, Joint Space Narrowing (JSN), emergence of osteophytes, and variations in bone texture. OA severity is categorized into five stages as stated by Kellgren and Lawrence (KL) classification as follows: normal, termed as Grade 0; doubtful, termed as Grade 1; minimal, termed as Grade 2; moderate, termed as Grade 3; and severe, termed as Grade 4. Manual assessment is typically employed, with Grade 0 indicating the nonexistence of KOA and Grade 2 evincing preliminary signs of OA [1]. Studies have emphasized the importance of accurately characterizing bone texture in radiographic images for early OA detection, even before JSW degradation [2].

Various diagnostic imaging techniques cover X-ray imaging; Magnetic Resonance Imaging (MRI); and scintigraphy. X-ray imaging, among the methods mentioned, is extensively used for diagnosing and evaluating KOA symptoms; due to their widespread availability, safety, and cost-effectiveness. However, detecting, assessing, and predicting KOA at its early stages based solely on certain indications and X-ray images can be notably challenging.

KOA diagnosis has been the focus of exploration by numerous researchers utilizing Machine Learning (ML) models. Copious classification models have been suggested, including logistic regression; Artificial Neural Network (ANN) [3]; decision tree; random forest; and Support Vector Machine (SVM). However, these studies indicate that detecting symptoms and risk factors contributing to OA is a multifaceted endeavor. Given the diverse nature of KOA, there is a significant demand for an efficient approach capable of predicting OA by utilizing identified key features [2]. Such an approach could aid medical practitioners in intervening before the condition progresses to a severe stage.

The subsequent sections complement this segment of the research manuscript: Section 2 provides a comprehensive overview, detailing existing techniques for detecting and classifying osteoarthritis. Section 3 delves into the framework of the proposed approach and outlines the various steps in implementing DCNN for the classification of KOA. Section 4 encompasses the assessment and results of the proposed methodology. In conclusion, Section 5 provides the final remarks of the paper.

2 Related Works

The segmentation [4] and classification of OA present significant challenges because of its diverse nature. Texture analysis, for instance, aid as a valuable technique in assessing variations, stiffness, and swelling within the joints, allowing for the differentiation of distinct X-ray regions. Leveraging texture analysis as an efficient image classifier enables accurate identification of various stages of OA, yielding compelling results [5]. [6] proposes a method in which the region of interest (ROI) is converted using discrete wavelet transform (DWT) based statistical features representing textural changes for multi-resolution analysis; and subsequently fed to the SVM for detecting OA severity grades. While these techniques prove effective, they frequently face limitations including intensive computational demands, difficulties in differentiating similar patterns, and constraints regarding dataset size [7].

Advanced methodologies for OA classification have been introduced with the emergence of Deep Neural Network (DNN) and ML, leveraging their capacity to discern

intricate OA patterns and deliver precise outcomes. For instance, [8, 9] graded KOA severity based on plain radiographs utilizing a Densely Connected Convolutional Neural Network (DenseNet), showcasing precise and dependable OA diagnosis. An introduction was made to a hybrid CNN model, which integrated a Histogram of Oriented Gradient (HOG) feature extraction approach, achieving a remarkable 97% accuracy in diagnosing and categorizing KOA across the KL scales.

Thus, the motivation for this research stems from the significant impact of KOA on individuals' quality of life, characterized by symptoms like pain, stiffness, joint dysfunction and also from the existing literary works. OA predominantly affects weight-bearing joints, leading to reduced flexibility and gradual deterioration, especially in older individuals. Current diagnostic methods, including X-ray imaging, are limited in detecting early OA stages accurately. Machine learning models have shown promise, but the heterogeneity of OA poses challenges. Therefore, there's a pressing need for an efficient approach to early prediction and managing OA effectively. Thus, the study presented here desires to bridge this gap by developing a DCNN for early OA detection, leveraging key features extracted from imaging data. The subsequent sections provide a concise overview of the proposed approach.

3 Proposed Study

A flowchart visually represents the proposed study, outlined in Fig. 1.

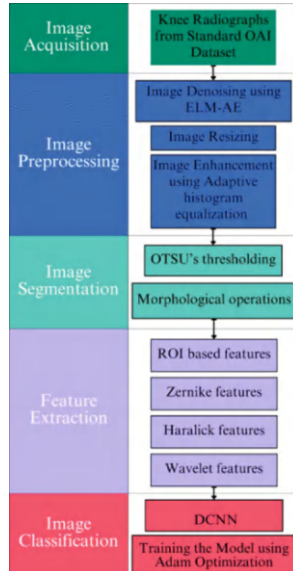


Fig. 1. The structural arrangement of the classification framework.

This study primarily works to create an improved classification framework for Knee Osteoarthritis severity grading. The focus lies on extracting features, segmenting, and

classifying. The proposed study comprises preprocessing, segmentation, feature extraction, and classification components. For classification, a DCNN architecture is used, that involves incorporating a feature input layer, followed by a fully connected layer, ReLU activation, and a SoftMax output layer.

Further elaboration on the classification framework is provided in the following sections.

3.1 Data Collection

Experiments conducted on Mendeley data [10] involved medical radiographic images associated with KOA. Data is sourced from the Osteoarthritis Initiative (OAI), a standard database, which allows the use of data and images collected over an eleven-year longitudinal cohort study. The dataset comprises knee X-rays, primarily utilized for spotting the knee joint and KL grading. All knee X-rays in the dataset adhere to bilateral PA Fixed Flexion positioning. Here, in our experiments, 5778 images are considered, and distributed across different severity grades of KOA: 2286 under grade 0, 1046 under grade 1, 1516 under grade 2, 757 under grade 3, and 173 under grade 4. This dataset is partitioned into three subsets: 70% for training, encompassing 4044 images; 15% to test (868 images); and another 15% to validate (866 images). Additionally, the dataset includes self-reported information concerning signs of OA and various threat factors, grouped into different feature fragments based on their characteristics, and then utilized for classification.

3.2 Pre-processing

Typically, diagnostic images are susceptible to interference, making them prone to corruption. In this study, we address the artifacts and disturbances present in the input knee X-ray images through three techniques: denoising, resizing, and enhancement.

Image Denoising. In the present study, we enforce the ELM-AE algorithm to denoise the input image [11]. The DCNN is trained using the ELM method with fewer parameters, and by utilizing random feature maps, achieves high computational speed. The hidden layer of the CNN is initialized by ELM for feature mapping, employing a nonlinear mapping function to map input data to the feature space. Randomly initialized hidden neurons in CNNs, along with trained output weights, minimize errors and enhance computational efficiency. The ELM architecture comprises, a single linear output layer along with a hidden layer.

Consider $X = [x_1, x_2, \dots, x_n] \in R^{d \times n}$; $Y = [y_1, y_2, \dots, y_n] \in R^{m \times n}$ as the training samples. In correspondence, the ELM output with L hidden neurons is represented in Eq. 1,

$$f_L(x_i) = \sum_{j=1}^L \beta_j h(a_j, b_j, x_i) = y_i; \text{ for } i = 1, \dots, n \quad (1)$$

Here, β_j denotes output parameter weights; and h represents activation function. Further, Eq. 1 is expressed in matrix form, as depicted in Eq. 2,

$$H\beta = Y^T \quad (2)$$

Here, $\beta = [\beta_1, \beta_2, \dots, \beta_L]^T \in R^{L \times m}$. Matrix H , shown in Eq. 3, consists of activation values obtained by applying the activation function h to each training sample x_i using parameters a_j and b_j

$$H = \begin{pmatrix} h(a_1, b_1, x_1) & \cdots & h(a_L, b_L, x_1) \\ \vdots & \ddots & \vdots \\ h(a_1, b_1, x_n) & \cdots & h(a_L, b_L, x_n) \end{pmatrix} \quad (3)$$

ELM primarily aims to minimize objective function as given in Eq. 4,

$$\text{Minimize } \|\beta_p^{\sigma_1}\| + C \|H \cdot \beta - T\|_q^{\sigma_2} \quad (4)$$

Here, $\sigma_1 > 0, \sigma_2 > 0, p, q = 0, 0.5, 1, 2, \dots, \infty$; parameter C plays a crucial role in generalizing the CNN model. Here, $\|\cdot\|_p^{\sigma_1}$ and $\|\cdot\|_q^{\sigma_2}$ denote p -norm and q -norm regularizations with scaling factors σ_1 and σ_2 respectively.

The ELM layers are organized in a cascaded network configuration to establish a feature learning network. This network adeptly identifies relevant feature data while eliminating extraneous data.

The input data undergoes normalization between 0 and 1. The hidden layer generates random input weights, within the range -1 and 1. Activation of the hidden layer involves computing the weighted sum of inputs, followed by applying the ReLU activation function. Subsequently, the output weights are then determined by applying the Moore-Penrose pseudo-inverse to the hidden layer activations. Later the output is obtained by multiplying the new hidden layer activations by the output weights. Root Mean Squared Error (RMSE) evaluates the autoencoder's performance by taking the input data and the reconstructed output. This assessment provides insights into the performance concerning RMSE, output weights, and representation of the new hidden layer.

Resizing and Enhancement. The CNN model necessitates uniform sized input, prompting the resized denoised image to be adjusted to the dimensions of 512×512 pixels through bilinear interpolation. This process ensures minimal distortion of features and data patterns. Additionally, adaptive histogram equalization helps intensify the contrast of the resized image.

3.3 Segmentation

The preprocessed images undergo further treatment through image segmentation using OTSU's thresholding determining an optimal threshold for converting the image from grayscale to binary. Subsequently, the segmentation results are refined using morphological operations and connected component analysis, thus enhancing their accuracy.

3.4 Feature Extraction

KOA features are extracted to classify various disease stages. From the segmented images, this study combines four distinct features—region features, Zernike features, wavelet features, and Haralick features—to extract pertinent information.

Region of Interest (ROI) Based Features. Normalized feature point areas extract ROI, in the proposed study. It assigns labels to the connected components in the binary image to calculate diverse statistical properties for each labeled region, including area, major and minor axis lengths, diameter, orientation, eccentricity, solidity, extent, and perimeter.

Zernike Features. The binary image with specified order and repetition compute Zernike moments. These moments represent a particular ROI within the image, denoted as $f(x, y)$. The Region of Interest (ROI) is determined by a feature set that includes specific image attributes like size, position, and orientation. Equation 5 encapsulates the comprehensive image information,

$$m_{pq} = \sum_x \sum_y x^p y^q f(x, y) \quad (5)$$

Here, m_{pq} gives the moments of order p and q .

For improved feature estimation, the polynomials of the Zernike moments must maintain orthogonality. The projection of an image with orthogonal polynomials represents these moments, as depicted in Eq. 6,

$$A_{nm} = \frac{n+1}{\pi} \sum_x \sum_y f(x, y) V_{nm}^*(\rho, \theta) \quad (6)$$

where, A_{nm} is the Zernike moments, $V_{nm}^*(\rho, \theta)$ denotes Zernike polynomials complex conjugate.

Haralick Features. The Gray Level Co-occurrence Matrix (GLCM) computes features named Haralick texture, which evaluates image texture based on parameters including contrast, correlation, sum of squares, sum of averages, and homogeneity. With N representing the number of grey levels and $p(i, j)$ indicating the element at the position (i, j) in the GLCM. The parameter definitions are provided in Eqs. 7, 8, 9, and 10.

$$Contrast = \sum_{i=1}^N \sum_{j=1}^N (i-j)^2 p(i, j) \quad (7)$$

$$Correlation = \sum_{i=1}^N \sum_{j=1}^N \frac{(i \cdot j) p(i, j) - \mu_x \mu_y}{\sigma_x \sigma_y} \quad (8)$$

$$Energy = \sum_{i=1}^N \sum_{j=1}^N p(i, j)^2 \quad (9)$$

$$Homogeneity = \sum_{i=1}^N \sum_{j=1}^N \frac{p(i, j)}{1 + (i-j)^2} \quad (10)$$

Wavelet Features. The 2D discrete wavelet transform (DWT) calculates the mean values of the approximation, horizontal, vertical, and diagonal coefficients. This process examines disparities between a denoised image and a low-frequency sub-band image, leading to the segmentation of high-frequency sub-bands into 2 distinct categories. The output images are calculated by further processing the images to extract high-frequency features from the original low-resolution images. This methodology ensures the extraction of high-frequency image features for classification purposes.

3.5 Classification Using a DCNN Model

Architecture of DCNN. The DCNN architecture represents an enhanced iteration of the traditional CNN model, incorporating supplementary layers. This model includes alternating convolutional and max-pooling layers, configured with Z-score normalization at the input layer for data standardization. Following this, a fully connected layer with 300 as the initial output size is established, succeeded by batch normalization along with the ReLU. These layers are sequentially added, progressively reducing the output size from 200, 100, 75, 50, 25, 15, and 10. A fully connected layer is utilized for classification and features the output size tailored to match the number of classes. Following this, a softmax layer and a classification layer are added. Designed to augment the feature representation capabilities of the classifier, the DCNN model predicts both low and high-level data. Further insight into the CNN model with layer-wise comparison is given in Table 1.

Table 1. DCNN Model Layers

Layer	Layer Type	Filter Size	Stride	No. of Filters	FC Units	Input Shape	Output Shape
1	Convolution	3×3	1×1	128	-	$128 \times 16 \times 16$	$128 \times 16 \times 16$
2	Max-Pool	3×3	2×2	-	-	$128 \times 16 \times 16$	$128 \times 8 \times 8$
3	FC	-	-	-	256	8192 (Flatten)	256
4	FC	-	-	-	256	256	256
5	FC	-	-	-	5	256	5

The five layers of DCNN commence with a convolutional layer and conclude with an output layer, with pooling and fully connected layers in between, as depicted in Table 1. The convolutional layer utilizes 3×3 sized 128 filters, subsequently, a max-pooling layer follows, with a 3×3 sized filter and 2×2 stride. The subsequent fully connected layers have 256 units each, and the final output layer comprises 5 units representing the severity classes. The input is Z-score normalized data, and the network employs ReLU activation function consistently to perform nonlinear transformations.

Training and Optimization of the DCNN. This study involves fine-tuning the DCNN parameters using Adam optimization to improve the performance of the DCNN, particularly when dealing with smaller datasets, thereby ensuring quicker convergence. The distinction between deep and shallow fine-tuning plays a crucial role in affecting DCNN performance. In this research, shallow fine-tuning is conducted in the final section of the network, contributing to the enhanced efficiency of the DCNN for classification tasks.

4 Results

The experimental evaluation of the DCNN model as a classifier, performed using the standard OAI database, yielded insightful results. A confusion matrix, depicted in Fig. 2, illustrates the model’s predictions alongside the actual ground truth labels for comparison. This matrix facilitates a comprehensive assessment of classification performance by providing insights into true positives, true negatives, false positives, and false negatives.

True Class	1	121	28	3			79.6%	20.4%
	2	17	180	5		2	88.2%	11.8%
	3	5	23	84		4	72.4%	27.6%
	4				369		100.0%	
	5	1	2	1		21	84.0%	16.0%
		84.0%	77.3%	90.3%	100.0%	77.8%		
	16.0%	22.7%	9.7%		22.2%			
	1	2	3	4	5	Predicted Class		

Fig. 2. Confusion matrix and classification performance

Notably, the segmentation results displayed a promising accuracy of 95.80%, indicating a high level of correct classifications, while the low false positive rate of 2.63% underscored the model's efficacy in distinguishing true positives from false positives. Balanced performance in detecting positive and negative instances was evident from sensitivity and specificity values of 84.85% and 97.37%, respectively, along with a precision of 85.88% and an F1 score of 85.06%.

As a part of feature extraction, the types along with the count of features extracted are as listed: Region-Based Features: 15, Zernike Feature: 1, Wavelet Features: 4, Harlick Features: 14. Therefore, with 28 layers, a total of 931 features were extracted and subsequently fed into the classification layer. Figure 3 exhibits preprocessing, segmentation, feature extraction, and classification outcomes along with the corresponding inputs for all 5 KL grades.

The DCNN classifier discussed here is compared with some existing KOA classification models and is shown in Table 2 along with their performance indicators.

[7] has given a comparison against various deep learning models (as mentioned in Table 3), in four experiments: Traditional descriptors; Deep CNN; Fine Tuning (FT);

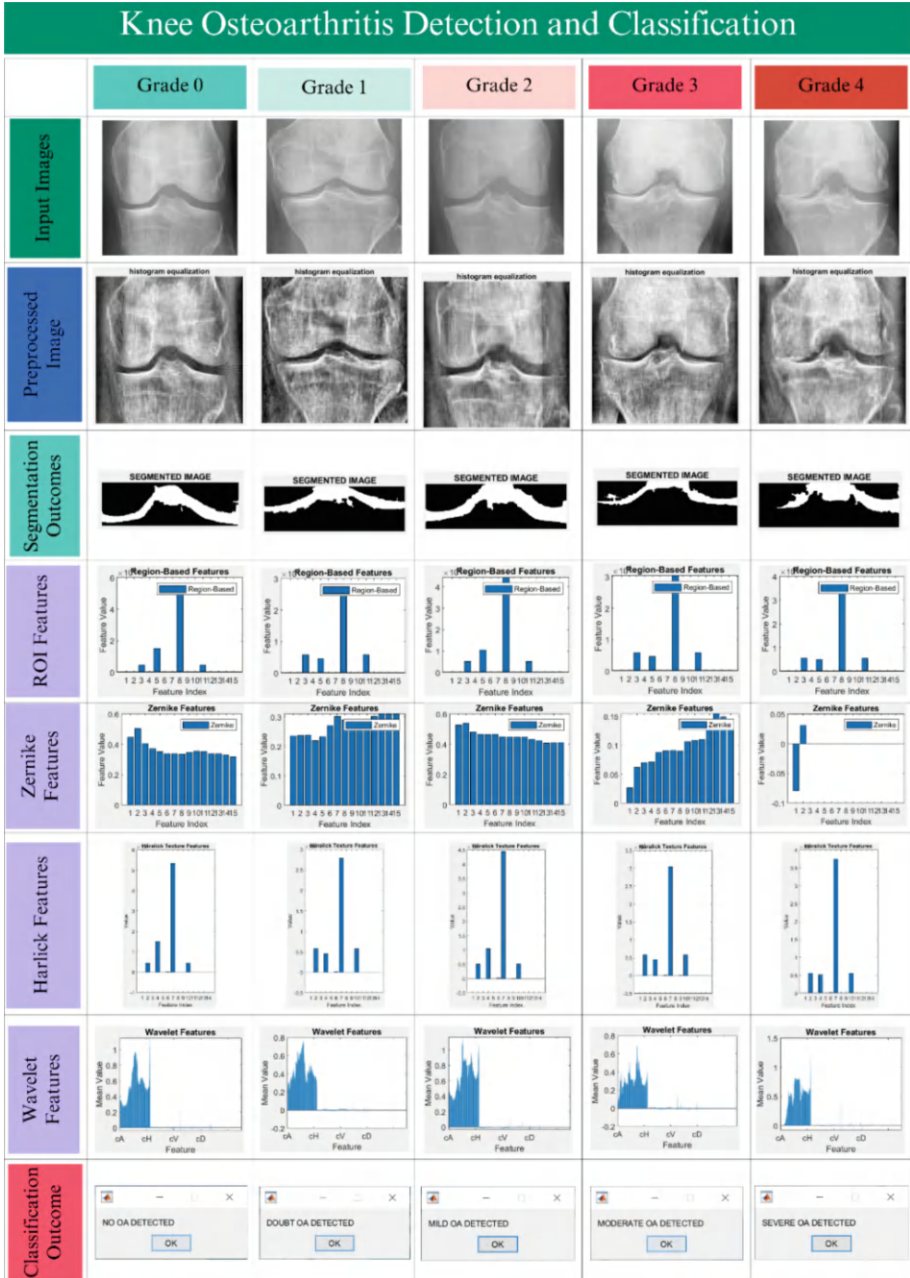


Fig. 3. Preprocessing, segmentation, feature extraction, and classification outcomes along with inputs for all 5 KL grades.

Deep CNN: Global Average Pooling (GAP) + SVM; and Complex networks. Fine-tuning pre-trained models led to better results, leveraging specific OA-related characteristics. Table 3 shows the comparison for the same standard OAI dataset along with our proposed method's performance. From the table, it is clear that our method outperformed other models in terms of accuracy. The proposed DCNN model achieves a sensitivity of 83.03%, specificity of 97.13%, and accuracy of 95.59% for the training dataset. Likewise, the proposed approach achieves accuracies of 96.31% and 95.80% for the testing and validation datasets, respectively, compared to 81.69% accuracy achieved by the complex network, showcasing competitive results with a smaller sample size. Hence, the results suggest that the proposed approach proficiently classifies the KOA severity grades.

Table 2. Comparison of results

Method		No. of X-ray Images	Sensitivity (%)	Specificity (%)	Accuracy (%)
Traditional Descriptors		688	79.36	75.00	77.18
	GLCM	688	82.85	74.42	78.63
	Fractals	688	70.06	72.09	71.08
	LBP	688	68.31	67.15	67.73
	CLBP	193	81.40	79.30	80.40
		137	76.80	80.90	78.80
		688	85.47	75.29	80.38
Deep CNN: Fine Tuning	AlexNet	688	79.94	70.64	75.29
	VGG-16	688	73.98	71.22	73.98
	GoogleNet	688	78.20	72.38	75.29
	InceptionV3	688	78.20	67.15	72.67
	ResNet-101	688	76.74	83.14	79.94
	ResNet-50	688	74.71	77.32	76.02
	DenseNet-201	688	94.48	62.79	78.63
	EfficientNet-B7	688	53.20	97.96	75.62
Deep CNN: GAP + SVM	AlexNet	688	79.07	69.19	74.13
	VGG-16	688	77.33	68.31	72.82
	GoogleNet	688	65.41	53.20	59.30
	InceptionV3	688	77.62	67.73	72.67
	ResNet-101	688	65.70	63.37	64.53
	ResNet-50	688	72.09	63.66	67.88
	DenseNet-201	688	77.91	73.55	75.73
	EfficientNet-B7	688	77.03	65.41	71.22
Complex Networks	SVM, KNN, and Linear Discriminant Analysis (LDA)	688	77.62	85.75	81.69
Proposed Method	DCNN (Training)	4044	83.03	97.13	95.59
	(Testing)	868	83.73	97.66	96.31
	(Validation)	866	84.85	97.37	95.80

5 Discussion

In the discussion section, the emphasis is placed on interpreting and contextualizing the results obtained from the experimental analysis, particularly focusing on the performance of the DCNN classifier. As a part of the evaluation, while training, the model adeptly learns to map integrated features to the equivalent osteoarthritis severity class, leveraging a large input dataset. Subsequently, while testing, the model demonstrated its capability to make accurate predictions regarding the severity class. The dataset, comprising knee X-rays in bilateral PA Fixed Flexion position, encompassed 5778 images across various OA severity grades, distributed as 70% for training, 15% for testing, and 15% for validation.

Various metrics evaluate the proposed DCNN's performance, including F1 score, specificity, sensitivity, accuracy, precision, error, false positive rate, and Matthew's correlation coefficient, both for individual classes and overall. Feature extraction revealed the extraction of 931 features from 28 layers, including region-based, Zernike, wavelet, and Haralick features. A comparative analysis with other classification models demonstrated the proposed DCNN model's superiority, highlighting accuracy, sensitivity and specificity. The proposed approach achieved training accuracy of 95.59%; 96.31% and 95.80% for testing and validation datasets, respectively, outperforming other models with competitive results even with a smaller sample size. These findings highlight the efficacy of the proposed method in precisely categorizing the severity of OA stages, thereby offering the potential for enhanced diagnosis and management of the condition. The model is validated using a robust evaluation process that involves testing its performance on a separate dataset from the one used for training. During the training, after each epoch or a certain number of iterations, the validation set serves to assess the model's performance. This verifies that the model's performance generalizes effectively to new, unseen data, signifying its reliability and effectiveness in practical situations, and thus helps prevent overfitting.

6 Conclusion

In conclusion, this study presents an innovative methodology for OA severity classification, leveraging advanced deep learning techniques and comprehensive image processing methodologies. Through the integration of feature extraction, segmentation, and classification blocks, a DCNN architecture is developed to accurately classify OA stages. The methodology encompasses robust preprocessing steps to enhance image clarity and mitigate noise, followed by segmentation techniques to isolate relevant features. Feature extraction from the segmented images, including region-based, Zernike, wavelet, and Haralick features, facilitates the extraction of pertinent information crucial for OA severity classification. The DCNN model, with its convolutional and max-pooling layers, is adept at processing and classifying the extracted features. Training and optimization of DCNN parameters further enhance model performance, particularly in dealing with smaller datasets. Experimental results demonstrate promising segmentation accuracy, with robust sensitivity, specificity, precision, and F1 score metrics indicating the model's efficacy in distinguishing OA severity levels. Thus, the DCNN model, validated on 866 X-ray images, achieved a sensitivity of 84.85%, a specificity of 97.37%, and an accuracy of 95.80%.

Comparative analysis against existing deep learning models underscores the superiority of the proposed approach, particularly in achieving higher accuracy rates across various datasets. The proposed methodology showcases its potential as a reliable tool for OA severity classification, with superior accuracy rates across training, testing, and validation datasets. Largely, this study sticks up to the progress of medical image analysis by offering a streamlined and robust framework for assessing OA severity, with potential implications for early detection and intervention strategies in clinical contexts. Further research could focus on expanding the dataset and refining the model architecture to enhance performance in real-world clinical applications.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Kellgren, J.H., Lawrence, J.S.: Radiological assessment of osteo-arthritis. *Ann. Rheum. Dis.* **16**, 494–502 (1957)
2. Hegadi, R.S., Navale, D.N., Pawar, T.D., Ruikar, D.D.: Multi-feature-based classification of osteoarthritis in knee joint X-Ray images. In: *Medical Imaging*, pp. 74–96. CRC Press (2019). <https://doi.org/10.1201/9780429029417-5>
3. Hegadi, R.S., Navale, D.I., Pawar, T.D., Ruikar, D.D.: Osteoarthritis detection and classification from knee x-ray images based on artificial neural network. pp. 97–105 (2019)
4. Hegadi, R.S., Navale, D.I.: Quantification of synovial cavity from knee X-ray images. *ICECDS Conference 2017*, pp. 1688–1691. IEEE, (2017)
5. Navale, D.I., Hegadi, R.S., Mendgudli, N.: Block based texture analysis approach for knee osteoarthritis identification using SVM. *Wiecon-ECE Conference 2015*, IEEE, pp. 338–341. IEEE (2015)
6. Navale, Dattatray I., Ruikar, Darshan D., Houde, K.V., Hegadi, R.S.: DWT textural feature-based classification of osteoarthritis using knee X-Ray images. Presented at the (2021). https://doi.org/10.1007/978-981-16-0493-5_5
7. Ribas, L.C., Riad, R., Jennane, R., Bruno, O.M.: A complex network based approach for knee Osteoarthritis detection: data from the Osteoarthritis initiative. *Biomed. Signal Process. Control* **71**, 103133 (2022). <https://doi.org/10.1016/j.bspc.2021.103133>
8. Chaugule, S., Malemath, V.S.: Osteoarthritis detection using densely connected neural network. *RTIP2R Conference 2021*, pp. 85–92. CCIS, Springer Nature (2022)
9. Chaugule, S.V., Malemath, V.S.: Knee Osteoarthritis grading using densenet and radiographic images. *SN Comput. Sci.* **4**, 63 (2022)
10. Chen, P.: Knee Osteoarthritis severity grading dataset. Mendeley Data (2018)
11. Chaugule, S., Malemath, V.S.: An extreme learning machine-based autoencoder (ELM-AE) for denoising knee X-ray images and grading knee Osteoarthritis severity. *RTIP2R Conference 2022*, pp. 164–175. CCIS, Springer Nature (2023)



Lung Cancer Detection and Analysis Using Machine Learning

Sanskruti S. Joshi^(✉), Janhavi Bhagat, Ambarish Kulkarni, Shambhawi Sharma,
and S. M. Mali

Dr. Vishwanath Karad, MIT World Peace University, Pune, India
1132220308@mitwpu.edu.in

Abstract. Lung diseases, such as pneumonia and lung opacity, contribute significantly to morbidity, mortality worldwide. Early detection and accurate classification of these conditions through medical imaging plays decisive role in patient outcomes. In this work, we propose a ML- based approach to classify lung X-ray images into three categories: normal, lung opacities, and viral pneumonia. We trained the model using Convolutional Neural Network (CNN) using the InceptionV3 architecture, with an accuracy of 91%. ML model was trained using an ImageDataGenerator for data augmentation and evaluated on a separate test set. Performance metrics such as accuracy, loss, and confusion matrices were used to measure the model's effectiveness. Our results demonstrate that the proposed method achieved high accuracy in classifying lung X-ray images. Additionally, we discuss the importance of these results with reference to medical diagnostics and outline potential future work to upgrade the model's performance. By integrating this model into a diagnostic framework, healthcare professionals can leverage it for rapid and accurate identification of lung conditions. This has the potential to assist in early intervention, reduce misdiagnoses, and ultimately contribute to better.

Keywords: Lung Diseases · Lung Opacity · CNN · InceptionV3 · ML

1 Introduction

According to the World Health Organization (WHO), respiratory illnesses, including lung-related conditions, are among the leading causes of death globally. Too early detection and accurate classification of lung diseases can significantly improve patient outcomes, allowing for timely intervention and treatment. However, this can be challenging due to the complex nature of these diseases and the limitations in conventional diagnostic methods. ML, particularly DL, has emerged as an encouraging approach for medical image analysis, offering automated and accurate diagnosis capabilities. This study explores the use of CNN for classifying lung X-ray images into three categories: Normal, Lung Opacity, and Viral Pneumonia. The proposed methodology leverages the InceptionV3 architecture, a state-of-the-art deep learning model, along with data augmentation techniques to improve model robustness. To build ML model, we used dataset

of lung X-ray images, applying various preprocessing steps to enhance image quality. We trained the CNN model with the Adam optimizer, evaluating its performance using metrics such as accuracy, loss, and confusion matrices. Outcome indicates that our approach can effectively classify lung X-ray images, providing a valuable tool for medical professionals. In addition to building the ML model, this study discusses the implications of using DL for medical image analysis and outlines potential future applications. By automating the classification of lung diseases, this research aims to assist healthcare providers in early diagnosis, ultimately contributing to improved patient care and reduced mortality rates. Through this study, we aim to demonstrate the potential of deep learning in medical diagnostics and pave the way for improvements in the field. The following sections will detail the data collection process, model training, evaluation, and results, followed by a discussion of the study's significance and possible extensions for future research.

2 Literature Review

O. Ångel [6] made a comparison between lung cancer diagnosis algorithms (artificial neural network path) and ML algorithms in the Japanese Society for Radiological Technology standard digital imaging database, and Model accuracy, evaluate in terms of precision, recall, and confusion matrix. Radhika PR, Rashi. A. S. Nair, Veena G [1] analyzed pulmonary clinical data obtained from UCI ML repositories using supervised learning methods to develop lung cancer diagnostic model and obtain SVM model did. Şekeroğlu, K. Tuncal [3] used data obtained from the WHO 2018 to estimate the occurring of cancer in European countries and created a model using linear regression, support vector regression, short-term memory, regression for gender groups, and the spread of different types of cancer, radial root ML Syed, A.A., Fatima, W., Wajahat, R. (2018) [4] investigates the role of preliminary images in medical image classification. They used OpenCV to enhance the images and then feed them into a deep learning model; This showed that prioritization can affect the performance of the model. This is focused on our use of image enhancement methods to improve the quality of classification of chest X-ray. It helps in early diagnosis of human cancer. Ultimately, this will help reduce deaths, increase awareness of medical needs, and save people's money. After reviewing 15 research papers, we built a Lung Cancer Detection model. It helps in the early Diagnosis of Cancer.

3 Methodology

3.1 Data and Pre-processing

This research includes, the dataset used for testing and training is derived from collection of lung X-rays. These images are divided into three classes: Normal, Lung Opacity, Viral Pneumonia. The dataset comprises multiple images from different sources, stored in separate directories for each class. The images undergo pre-processing, including rescaling and enhancement, to ensure consistency and improve quality. Rescaling involves normalizing pixel values to a 0–1 range by dividing by 255. Image enhancement is applied

using OpenCV's functions to adjust brightness and apply filters for clarity. This process is crucial for achieving better results during model training and evaluation. Following the pre-processing, the dataset is split up into 3 categories training, validation, and testing subsets. The training subset constitutes 80% of data, meanwhile the validation and test sets each make up 10%. The split is stratified based on the class labels to ensure a balanced distribution of images across the datasets.

3.2 Image Data Generator and Model Configuration

The training, validation, and test datasets are fed into an ImageDataGenerator for augmentation and batching. The generator applies various transformations, including resizing images to 256×256 pixels, random rotations, flips, and other augmentations, to diversify the training set and reduce overfitting. The model used in this study is based on the InceptionV3 architecture, pre-trained with ImageNet weights. Transfer learning permits the model to leverage previously learned features and use them to this dataset. The base model is modified by adding a Dense layer with ReLU activation and a Dropout layer for regularization. The last layer uses Softmax activation for prediction of one of the three classes.

3.3 Model Training and Evaluation

Model compilation is done with the help of Adam optimizer and categorical crossentropy also known as logarithmic loss, chosen for its stability and efficiency with multi-class classification. The training is conducted over 20 epochs, with validation data used for monitoring the model's performance during training. Model evaluation involves measuring metrics such as accuracy and loss. Afterwards, training of model is calculated, in the test subset to determine its final performance. Confusion matrices are generated to analyze the model's classification accuracy for each class.

3.4 Analysis of Model Results

Upon completion of training, model's performance is examined based on these categories accuracy, precision, recall, and F1-score. These estimations offer conceptions into the model's dependability and effectiveness in classifying lung X-ray images into the three categories. To identify possible areas for improvement, the confusion matrix is also utilized to compare the algorithm's predictions to the actual class labels. This study's methodology is designed to address the challenges of lung disease classification through deep learning, with an emphasis on pre-processing, data augmentation, and robust model design to achieve accurate results. The following sections of this paper will detail the results obtained from this methodology and discuss the implications for medical image analysis and future research directions.

3.5 Convolutional Neural Network (CNN)

CNN is a deep learning model designed to process and classify complex data. It does so by convolving feature maps with filters to identify patterns and then pooling to reduce

dimensionality. This enables the model to identify specific features in the dataset, aiding in accurate classification. In this study, a CNN architecture based on the pre-trained InceptionV3 model is used to classify medical images into three classes: Normal, Lung Opacity, and Viral Pneumonia. The pre-trained InceptionV3 model provides a strong foundation with robust feature extraction capabilities, and additional layers are added for specific classification tasks. In the process of lung X-ray image classification, the CNN architecture employs the following components: InceptionV3 Base Model: This model serves as backbone for feature extraction. It is pre-trained on the ImageNet dataset, providing a rich source of learned features for image analysis. Additional Layers: A DenseLayer with ReLU activation to further evaluate the features extracted by the base model. A Dropout Layer has dropout rate of 0.45 for regularization and to reduce over-fitting. An output Dense Layer with Softmax activation for multi-class classification, outputting probabilities for each class. Training Parameters: Optimizer: For efficient gradient updates a learning rate of 0.0001 is done by Adam optimiser Loss Function: Multi-class categorization using categorical cross-entropy. Performance Metrics: The main criterion is accuracy, supplemented by metrics such as precision, recall, F1-score to evaluate the performance of the model. In context of lung disease classification, model achieved following results: The model was trained for 20 epochs, with data augmentation techniques applied through the ImageDataGenerator. Categorisation of the dataset was done into training, validation, test sets having split ratio approximately 64:16:20. The CNN model achieved a test accuracy of 91% based on the evaluation on the test set. This accuracy shows model's ability to correctly classify new, hidden images into the correct category. These results demonstrate the effectiveness of using a Convolutional Neural Network, specifically an InceptionV3-based architecture, in classifying lung X-ray images. Figures 1 and 2 illustrates the accuracy, loss trends while training and validation, demonstrating model's convergence and stability during training process.

Table 1. CNN Performance Metric Measures

Metric	Normal	Lung Opacity	Viral Pneumonia
Accuracy	90%	92%	94%
Precision	91%	90%	92%
Recall	91%	91%	90%
F1-Score	90%	90%	90%

3.6 SVM

X-rays into categories: Normal, Lung Opacity, Viral Pneumonia. We used the RBF kernel to segregate the samples into classes, as it is known for its adaptability to non-linear data. To optimize SVM's performance, we adjusted the hyper parameters to improve accuracy. The key hyperparameters included: Maximizing margin and minimizing classification

error is controlled by regularisation parameter. High values of C reduce misclassification but can lead to overfitting. The best results were obtained with the following hyperparameters: $C=10$, kernel='rbf', and $\gamma=0.01$. This configuration yield accuracy of 87% on test set. These results shows that SVM when properly configured, can effectively classify lung X-ray images into distinct categories. The performance metrics, demonstrate that SVM could be a reliable algorithm for this task. In summary, SVM, with its robust classification capabilities and flexibility in kernel selection, proves to be a useful tool for lung X-ray image classification

3.7 Performance Metrics

Evaluation of the performance of CNN model was used to classify lung X-rays into categories: Normal, Lung Opacity, and Viral Pneumonia. The performance metrics are taken from confusion matrix, which categorizes predictions into four groups: TP,TN,FP,FN. These metrics are calculated to assess the model's performance:

Table 2. SVM Performance Metric Measures

Metric	Normal	Lung Opacity	Viral Pneumonia
Accuracy	89%	89%	87%
Precision	86%	90%	90%
Recall	91%	87%	84%
F1-Score	87%	86%	92%

4 Result

Our study focuses on convolutional neural network model built using the InceptionV3 architecture for classifying medical images into three categories: Normal, Lung Opacity, and Viral Pneumonia. Dataset of chest X-rays is used for training and testing of the model, with data augmentation and pre-processing was applied for better model generalization and performance. Model training was carried out using TensorFlow and Keras, and other key dependencies such as Numpy, OpenCV, and Matplotlib for image processing and data visualization. In our study, the dataset was categorised into two, training and testing subsets. The training set comprises of 80% data, and test set was 20%. The model went through 20 cycles, utilizing a learning rate that is 0.0001 and employing Adam optimizer. The categorical logarithmic loss was employed as the depletion function. The model attained a test accuracy of 91% and a loss of 0.42. Accuracy, precision, recall, and F1 score is encompassed by the performance of the model. The model gained an accuracy rate-91%, a precision rate-90% a recall rate-91%, and an F1 score-90%. The result indicates that the model is capable of accurately categorizing chest X-ray pictures. To visualize the confusion matrix, the Seaborn library was used for plotting a heatmap,

which depicted the distribution of TP, TN, FP, FN. The performance was consistent across different categories, indicating its robustness in classifying chest X-rays. The implementation environment included open-source tools such as Jupyter Notebook and the model was built and tested using TensorFlow and Keras frameworks. The results show that the model has potential applications in medical image classification and can aid in the early detection of lung-related diseases. These findings suggest that CNN-based models can be valuable tools in medical diagnostics and can be integrated into healthcare systems to assist healthcare professionals in making informed decisions. Further research could focus on increasing model's accuracy and extending its applicability to different medical imaging datasets.

References

1. Radhika, P.R., Nair, R., Veena, G.: A comparative study of lung cancer detection. In: Proceedings of the IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT) (2018)
2. Syed, F.W.: Comparative analysis of learning algorithms for lung cancer identification (2018)
3. Sekeroglu, B., Tuncal, K.: Prediction of cancer incidence rates for the European continent using ML models (2021)
4. Mukhopadhyay, S., Garg, S., Prabhakar, S., Khandelwal, N.: Multi-classifier framework for lung tissue classification (2014)
5. Kadir, T., Gleeson, F.: Lung cancer prediction using ML and advanced imaging techniques (2018)
6. Günaydin, Ö., Günay, M., Şengel, Ö.: Comparison of lung cancer (2019)
7. Lynch, C.M., Abdollahi, B., Fuqua, A.R., Bartholomai, J.A., Balgmann, H., Frieboes, H.B.: Prediction of lung cancer patient survival via supervised ML classification techniques (2017)
8. Makaju, D., Prasad, P., Elchouemi, A.: Lung cancer detection using CT scan images (2018)
9. Roy, K., Chaudhury, A., Burman, R., Ganguly, A., Dutta, C., Banik, R.: A comparative study of lung cancer detection using supervised neural network (2019)
10. Tripathi, P., Tyagi, S., Nath, M.: A comparative analysis of segmentation techniques for lung cancer detection. *Pattern Recognition Image Anal.* **29**(1), 167–173 (2019). <https://doi.org/10.1134/S105466181901019X>
11. Miah, B.A., Yousuf, M.A.: Detection of lung cancer from CT image using image processing and neural network (2015)
12. Rossetto, A.M., Zhou, W.: Deep learning for categorization of lung cancer CT images (2017)
13. Deshpande, A.S., Lokhande, D.D., Mundhe, R.P., Ghatole, J.M.: Lung cancer detection with CT scans and MRI images using image processing and ML (2018)
14. Patil, S.A., Kuchanur, M.B.: Lung cancer classification using image processing (2019)



Machine Learning Based Depression Prediction Using Gradient Boosting Algorithm

Mohith Manoharan Satheesh, R. Nagavibha, Mishbah Pinto Chandy,
and N. Jayapandian^(✉) 

Department of Computer Science and Engineering, Christ University, Kengeri Campus,
Bangalore, India

{mohith.ms, nagavibha.r, mishbah.pinto}@btech.christuniversity.in,
jayapandian.n@christuniversity.in

Abstract. Depression is one of the major diseases, more than one million people are facing this issue. To achieve the best results possible, it is essential to monitor and intervene when needed regularly. While there are many ways to observe the mental well-being of an individual in a workplace environment, AI has the potential to enhance the accuracy, efficiency, and speed when it comes to diagnosing any issues. This study focuses on developing an ML system for distinguishing symptoms of depression among individuals in the workplace. The dataset comprises detailed information on the signs and symptoms of depression among individuals, it particularly focuses on the observed negative consequences at the workplace, physical health issues and their negative consequences, treatment. In this experimental process two main machine learning algorithms were used, the Random Forest and Gradient Boosting algorithm. Both the algorithms have an overall accuracy of 82%, but based on maximization of the overall performance, the Gradient Boosting model is slightly better than the Random Forest. Furthermore, our exploration demonstrates overall performance like character fashions, signaling promising prospects for sturdy and correct depression analysis class systems. This study highlights the power of machine learning that could revolutionize depression care by identifying mental health problems early.

Keywords: Artificial Intelligence · Depression · Healthcare · Random Forest · Gradient Boosting · Machine Learning · Decision Tree

1 Introduction

Statements like ‘Are you depressed?’ may be compelling to think of things that may have not even crossed your mind. Now, how do you know if you are really depressed or just sad. Depression can be defined as a constant feeling of voidness, hopelessness, emptiness. Depression not only affects your mental health but also has heavy impacts that can lead to physical problems [1]. Often the day-to-day work might seem impossible, but depression may require long-term treatment. Depression is considered a serious medical condition that can worsen without proper treatment [2]. It is very important to understand the difference between feeling low or sad in a day and feeling low or

sad constantly. Most people with depression tend to feel better with psychotherapy or with the help of medication or even both sometimes [3]. Depression occurs because of an imbalance of neurotransmitters in the brain that help in controlling the mood [4]. It can be caused because of factors like heredity, trauma, stress levels, various mental illnesses, substance abuse, or due to common reasons like losing a loved one, financial problems, and a breakup. Now, what are the common symptoms of depression? [5]. It may be a lack of interest, sleep disturbances, sleeping too much, sometimes reduced appetite or sometimes overeating, anxiety, agitation, restlessness, slowed movements or thinking, angry outbursts, frustration, trouble in decision-making, suicidal thoughts or attempts. These symptoms can help you understand and to consult a psychotherapist. Social media is a coin with two sides; it could work for both the better and the worse. It has been observed that the increased usage of social media has led to an increased rate of recorded depression cases. With the overflowing of negative emotions, terms like 'doomscrolling' have come into the picture. Substantially the rates of depression among teens and young adults have seen a surge with the increased amount of time being spent. According to the World Health Organization (WHO), it is estimated that 5% of adults suffer from depression in a year [6]. It has also been identified that it is more common in women than in men. It has also been identified that more than 10% of pregnant women or women who have just given birth also better known as postpartum depression suffer from a depressive disorder [7]. It has also been recorded that more than 700,000 people commit suicide a year because of depression. The stigma around the term depression itself is alarming as compared to other mental illnesses due to the negative nature of the illness that makes the person sound unreliable or incompetent [8]. This stigma can prevent an individual from undergoing proper treatment as it may cause the individual to feel shameful and hideous about the same. Helping someone going through depression may be challenging, try to listen and be there for them, do not be prejudiced, encourage them to seek professional or medical help, be with them throughout their treatment, help them to maintain a healthy lifestyle, and offer as much as practical help possible. This proposed methodology has considered two of the most well-known methods for classification for the proposed work such as Random Forest and Gradient Boosting over a dataset to derive a detailed understanding of the performance of algorithms and identify the most ideal one.

2 Literature Review

This article is going to discuss depression prediction using different machine learning algorithms. The machine learning models for the prediction of stress, anxiety and depression. The dataset collected was of 348 participants and the data was collected through Google forms where they had to answer a few questionnaires. The questionnaire consisted of 7 questions for each that is stress, anxiety, and depression with a total of 21 questions. The five machine learning models experimented on this dataset were Naive Bayes, Random Forest Tree, SVM, Decision Tree, and K-Nearest Neighbor [9]. There are six machine learning classifiers that use various psychosocial and socio-demographic information to determine whether a person is either depressed or not. The dataset was obtained from a survey of Bangladeshi citizens which comprised all different age groups.

The classifiers include AdaBoost, XGBoost, Gradient Boosting, KNN, weighted voting classifier and Bagging classifier. As the data was imbalanced the authors used the concept of Synthetic Minority Oversampling Technique (SMOTE) to help increase the chances of obtaining a higher accuracy [10]. This article concluded that the AdaBoost Classifier outperformed the other approaches with an accuracy of 92.56%. Twitter feeds for conducting analysis of emotions which focuses on depression. The dataset that the authors have considered for their exploration and analysis consists of a total of 10,000 tweets and it is then the dataset is divided into test and train with it being 20:80 in ratio. Every tweet is classified as negative and neutral, which is based on a selected list of words to detect the tendencies for depression. The authors have used two supervised learning models which are Support Vector Machine and Naive Bayes. In the end, the conclusion obtained is that Naive Bayes has a higher accuracy of 83% and SVM has an accuracy of 79%. The authors intend to analyze it specifically when the messages do not include words like 'diagnoses and 'depression'. The four classifiers used are SVM, Logistic Regression, Decision Tree (DT), and Multilayer Perceptron (MP) [11]. A methodology where the analysis of the selected features that are commonly used is performed using several feature selection methods and their effects on the result of the classification which later explains the depression model which is being used for detection. The framework that has been developed accumulates and chooses the most promising features from 38 feature selection algorithms throughout various categories to model depression detection. The authors have experimented on three real-world datasets from which 902 cues were extracted, speech prosody, head pose, speech behavior, and eye movement. The system will be able to analyses the syntactical markers related to the perpetual and onset detection of depression. This process helps to develop an algorithm that helps in predicting depression. The dataset considered for the project is obtained from tweets of the Twitter data. The authors have performed their project using the Random Forest machine learning algorithm and have also stated that the reason for using RF is that this algorithm reduces the problem of overfitting the dataset; the missing values are handled exclusively. The tweets from the dataset are labeled as positive and negative tweets. Henceforth, the accuracy of the already existing system of Naive Bayes was 94.8% but the proposed model (Random Forest) has an accuracy of 99.8%. In which the first part is based on the time and patterns that have been written for the content and the second one is in the form of linguistic clues where the text is analyzed, and the tweet is shared. The objective of the authors is to identify the symptoms of this disease early and help the patients suffering from depression and help the patient lead a happy life. The dataset is publicly available on Reddit which was published in 2017 and it includes features like writing, date, ID, and title. It contains the reports of people who have been suffering from depression. The machine learning models used for this paper are SVM, RF, Logistic Regression, and Bayes Theorem. The highest accuracy among these models belonged to Logistic Regression with 96.2%.

A system that uses tweets as a source of data, the authors have implemented the Back Propagation Neural Network (BPNN) model to determine whether to classify the tweets into non-depressed and depressed categories [12]. The methodology offered by the authors is in the following steps in order; Data acquisition, Sentiment analysis, and then the data is given into the machine learning model and further classification takes

place [13]. A hybrid model was proposed that merges the use of a technique of sentimental analysis such as SentiStrength to create the training data and a BPNN model to help classify the data. Each word from the tweets was analysed and put into different classes and the accuracy was calculated. Five different models the effective usage which include KNN, CNN, Linear Regression, SVM and Linear discriminant analysis. The dataset collected consisted of hard copies of questionnaires for two psychological disorders, anxiety and depression from the University of Dhaka and later the questionnaires of anonymous patients were collected from Dhaka Medical Hospital and Rangpur Medical College and Hospital. The dataset was split into test and train data using K-fold cross validation. The highest accuracy obtained was for CNN which had an accuracy of 96% for anxiety and 96.8% for depression. The primary goal is to investigate the feasibility of using mobile phone metadata as a tool for depression screening. The researchers aim to understand how changes in phone usage patterns could indicate depressive symptoms. The study employs machine learning techniques to analyze mobile phone usage metadata. This data includes call logs, message logs, and other forms of interaction that do not require the content of the communications, thus preserving privacy. The analysis seeks patterns or anomalies that correlate with depressive behaviour. It likely presents the findings from the data analysis, illustrating how certain usage patterns are linked to depression. This might include reduced communication, changes in the time of usage, or other measurable metrics. AutoDep uses LBP to extract features from facial expressions captured in images or video footage [14]. This technique identifies patterns in the texture of facial expressions, which are then analysed to detect signs of depression. The system is trained and tested on a dataset comprising facial expressions of both depressed and non-depressed individuals. Machine learning algorithms are employed to classify these expressions and identify depression markers. The article likely details the accuracy of AutoDep in correctly identifying depression, comparing its performance against traditional diagnostic methods [15]. The Synthetic Minority Oversampling Technique (SMOTE) model is to address class imbalance techniques [10]. That is the Least Absolute Shrinkage and Selection Operator (LASSO) technique for feature selection. The final model included 13 significant variables selected from an initial set of 37. The model demonstrated high predictive accuracy. The study identified perceived stress as the most influential variable in predicting depression. The findings suggest that LASSO is a viable method for depression screening using a limited number of variables. The study underscores the potential of machine learning in public health, particularly for early detection and intervention in mental health issues like depression. A study in which the objective was to identify whether an individual was depressed using social media data was done. The dataset consisted of 2000 sentences that were procured from various social media platforms such as Twitter, Facebook, and Instagram. Five different for data pre-processing in the realm of natural language processing (NLP) which included the removal of stop words, tokenization, removal of empty strings, punctuations, stemming, and lemmization. Finally, six machine learning models including Multinomial Naive Bayes, KNN, Linear SVC, RF, and LR. Out of these models, Multinomial NB and LR provided the best results with an accuracy of 98% and 97% respectively.

3 Problem Statement

Depression is a mental health disorder that affects millions worldwide and can be identified as a constant feeling of hopelessness, sadness, and lack of interest. It may or may not affect the work life, personal life, or school life. Depression is more than just mood swings; it can lead to other severe mental health issues like anxiety. Despite the common prevalence of depression in our society it somehow remains underdiagnosed and undertreated. It sometimes may not be treated or spoken of due to the stigma around the word, or sometimes even the misconceptions about mental health. Individuals suffering from depression can experience symptoms that may include just mere changes in a normal schedule like changes in appetite, sleep variations, decreasing concentration levels, excessive guilt, or sometimes even extreme thoughts of death and suicide. Depression may be caused due to multiple factors; it may involve a complex play of environmental, psychological, physiological, biological, and genetic factors. The interplay of various factors makes the treatment of depression challenging, it sometimes may require a personalized approach that may involve the help of friends, family, and spouses. It can be approached in the dimension with small changes like medication, psychotherapy, and lifestyle. In total, it may also affect factors like economic productivity, and healthcare costs. Therefore, the early identification of depression with machine learning models ensures early detection, prevention, and enhanced efficiency and precision of diagnostics.

4 Proposed Methodology

The architecture of the proposed model begins with the master database where we analyze the dataset which has been considered. The dataset is obtained from a 2014 survey which computes the attitude of employees towards mental health and the frequency of mental health disorders that have been observed in the workplace and helps explain in detail the lifestyle of the employees in an industry. If there are any null values present in the dataset, then the entire row will be eliminated. There are two major categories in the dataset. The primary value is testing and the secondary one is training the data. The trained data is the data that has been integrated into the model and the test is the data that the model has not encountered before, the model is evaluated using the test data by taking knowledge from the train data, and predictions are made. The model is being trained using x_{train} where it consists of the features of the training dataset and the y_{train} has the corresponding targets or labels. Using the new data that is the x_{test} the model is being predicted and the result is found. Finally, the accuracy of the model is predicted using the y_{test} . The amount of correctness of the model is the accuracy score which is received for the model. The dataset comprised 1259 records which were then sent for preprocessing, the preprocessing dealt with handling missing value, feature encoding or scaling and any other cleaning or transformation of data that was found to be necessary at that moment. The dataset was then split to conduct the training and the testing efficiently, 70% of the data was taken for training and 30% of the data was set apart for testing which helps in evaluating the performance of the model when it is open to unforeseen circumstances. After which a suitable model can be chosen based on the

problem statement. The best hypermeters for the models can be found with the help of techniques like Grid Search or Bayesian optimization. Hyperparameters are not learned from the dataset beforehand, they are set before the training process. Then train the models on the training set using the hyperparameters. The best found hyperparameters are then used to train the models. The models are then evaluated to get the accuracy or any other performance metrics. The accuracies of both the models are noted after hyperparameter tuning and are then compared with the performance before and after tuning. The model with the better performance is then deployed to make predictions on the new data. Random Forest is a supervised machine learning technique that entails the construction of a “forest” composed of several trees, which are specifically designed for the “bagging” approach. By amalgamating multiple learning models, the forest aims to enhance outcomes, thereby justifying the batch method (Figs. 1 and 2).

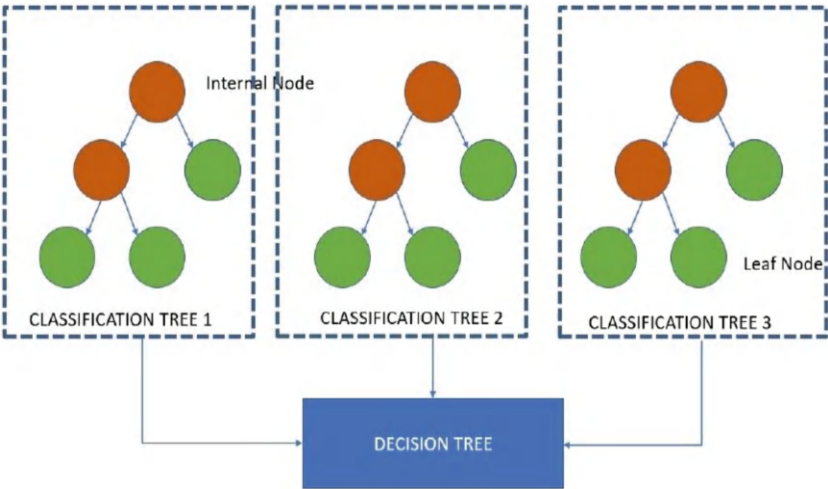


Fig. 1. Workflow of Random Forest Method

To establish a more dependable representation, random forests employ an abundance of distinct trees that are subsequently merged. This approach is advantageous as it effectively addresses the prevalent issues of ordering and repetition commonly encountered in contemporary machine learning frameworks. An additional noteworthy attribute of the random forest approach is its ability to facilitate the evaluation of the overall importance of each individual score component. The Gradient Boosting model is generally working with the decision tree method. Two tree structures are split like parent and child tree based on that parameter it will predict. To increase the efficiency of the model hyperparameter tuning is used for the machine learning algorithms. The main objective is to optimize the performance of the model which includes precision, accuracy, recall and F1 score. A hyperparameter value is set before the learning process begins and it influences the behavior of the training algorithm. This is different from model parameters, which are learned from the data. The approach used in hyperparameter tuning is grid search which is a systematic approach to optimize the hyperparameters of a machine learning

algorithm. The grid search is implemented in python using scikit-learn. The working of Grid Search is as follows.

- a) The parameter grid must be defined.
- b) Grid Search uses the concept of exhaustive search where it checks every combination.
- c) Cross-validation is used along with Grid Search. For every single amalgam for hyper-parameters, the training of the model is done multiple times for different datasets of the data and the average of the performance is considered.
- d) After assessing the possible combinations, Grid Search selects the hyperparameters that have gained the best performance.

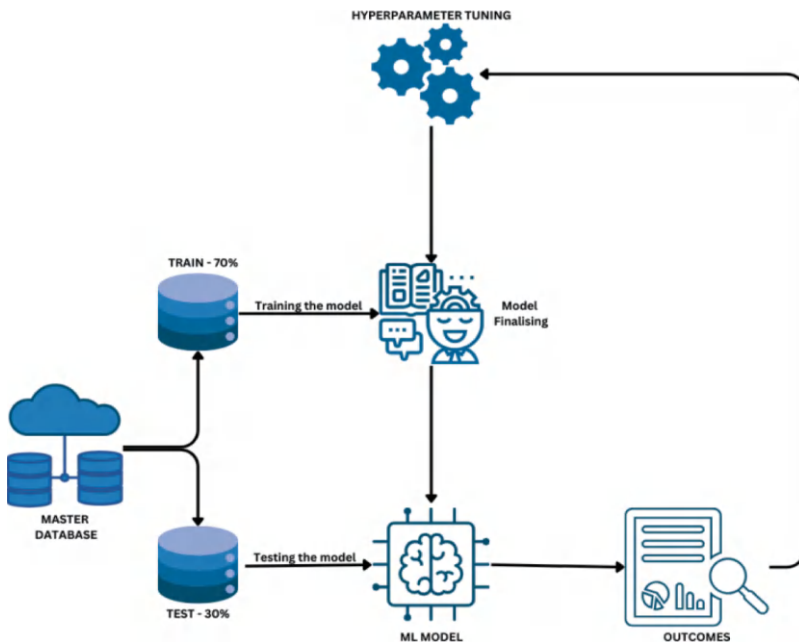


Fig. 2. Workflow Representing of Gradient Boosting

For an organization or system, a master database is the main database that acts as a centralized data repository. It keeps all the crucial information needed for the system to run smoothly. Data security, correctness, and consistency are managed and upheld by the master database. To keep the data current and applicable, it is updated regularly and is often replicated by secondary databases to increase scalability and ensure high availability. In general the training set is used to train the exact model by adjusting the parameters to reduce the error or loss function. In this case, the training set accounts for 70% of the overall dataset. The size of the training set is an important hyperparameter that can impact model performance. A larger training set can provide more additional data to the model, but it also raises the possibility of overfitting. It accounts for 30% of the total dataset in this case in the testing model. The testing set is used to estimate the training

error, or the model's expected performance with new data. The testing set is used to tune the model's hyperparameters and compare performance among both models. Hyperparameters are machine learning model parameters that are determined prior to the training process. They include the learning algorithm, the number of nodes, layers, the training algorithm, and the normalization parameter, among many others. Hyperparameter tuning is the practice of determining the optimal values of hyperparameters that will reduce the training error or the model's expected performance on new data. Hyperparameter tuning can be completed manually or automatically using methods like grid search, random search, and Bayesian optimization. A machine learning model is trained by exposing it to a subset of the data (generally the training set) and allowing it to learn from it. During training, the model employs machine learning and statistical styles to discover patterns and connections in the data. Grounded on these patterns and connections, the model is also acclimated to enhance its capability to prognosticate or classify new, preliminarily unknown data. The training process is generally iterative, with the model being trained and acclimated several times until it achieves better results. Testing a machine learning model involves assessing its performance on a separate portion of the data. The thing of testing is to ensure that the model is suitable to generalize its prognostications to new, unseen data. During the testing process, the model is presented with new data, and its prognostications are compared to the factual issues. The following parameter is used for measurement of the proposed model that is accuracy, recall value, or F1 score. The testing phase helps to ensure that the model is suitable for making dependable and accurate prognostications on real-world data. A machine learning model is a fine representation of a system or process that's learned from data. It's a set of algorithms and statistical styles that are used to identify patterns and connections in data. ML models can be used for a variety of tasks, similar as making prognostications, classifying data, or relating anomalies. Finishing a machine learning model involves preparing it for deployment in a product terrain. This may include tasks similar to optimizing the model for performance, integrating it with other systems, and testing it in a variety of scripts. Finishing a model also involves establishing its performance, limitations, and hypotheticals, as well as any applicable details about the data and the training process. This attestation is important for ensuring that the model is used and that its prognostications are interpreted rightly.

5 Result and Discussion

This research aimed at detecting signs of depression among employees in a workplace setting through the application of machine learning techniques. Two classifiers, Gradient Boosting and Random Forest, were employed and their performance was optimized using grid search for hyperparameter tuning. The survey gathered the data from more than 100 employees around the globe which encompasses variables such as age, gender, country, and type of work, family history, treatment and 21 other variables. Using the data, a machine learning model is trained to detect the signs of depression by splitting the data into train data 70% and test data 30% and evaluate the model. The best model among the machine learning models was found to be Random Forest Classifier and Gradient Boosting Classifier which gives accurate predictions. The two models were evaluated using accuracy, recall, F1 score and precision. The machine learning model is

implemented with hyperparameter tuning and optimizes the model’s prediction. Hyperparameter tuning approach implemented is Grid search. Table 1 is deliberate about the experimental setup details.

Table 1. Experimental settings Details

Parameters	Values
Dataset	Kaggle Dataset
Dataset size	1259
Training dataset size	881 (70%)
Testing dataset size	378 (30%)
Validation dataset size	378
Max_depth	3
Learning rate	0.01
Number of training iterations	50
Software	Jupyter Notebook

Table 2. Performance Analysis of Random Forest and Gradient Boosting

Algorithm	Accuracy	Precision	Recall	F1 Score
Random Forest	0.82	0.79	0.87	0.83
Gradient Boosting	0.82	0.76	0.93	0.84

The best parameters found via grid search for Random Forest Classifier is ‘max_depth’ is 10, ‘min_samples_split’ is 2, and ‘n_estimators’ is 100. Four major parameters are considered in this experiment. First accuracy level, the proposed model provides nearly 81.7% of accuracy. Second is the precision value that is 78.6%. The third one is recall value, that is 87.3%, and the final parameter is F1 score that is 82.7%. The best parameters found via grid search for Gradient Boosting Classifier is ‘learning_rate’ is 0.01, ‘max_depth’ is 3, ‘n_estimators’ is 50. First one is accuracy level; the proposed model provides nearly 82.2% of accuracy. The second one is the precision value that is 76.6%. Then the third one is a recall value that is 93.1%, and then the final parameter is an F1 score that is 84%.

The Table 2, Figs. 3 and 4 is deliberate on the different performance analysis in Random Forest and Gradient boosting algorithms. The accuracy level of both models is 0.82. But the precision value of random forest is 0.79 and gradient boosting algorithm precision value is 0.77. The recall value of the random forest algorithm is 0.87 and the gradient boosting method provides 0.93. Then finally F1 score is compared for both methods that is 0.83 is providing for random forest method 0.84 is providing gradient boosting method.

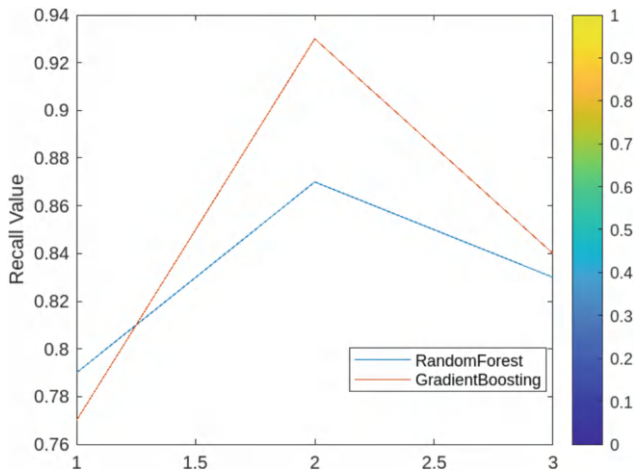


Fig. 3. Recall Value Comparative Analysis

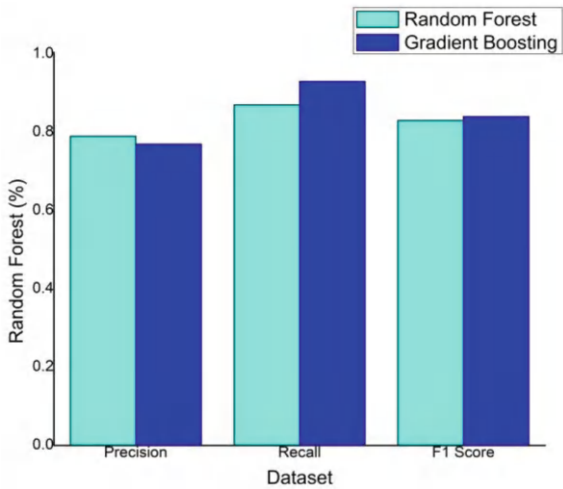


Fig. 4. Comparison of Random Forest and Gradient Boost Algorithm

6 Conclusion

Depression is a serious mental health disorder that affects millions of people worldwide. Early detection and treatment are crucial to achieving the best possible results. This study focuses on developing an ML system for detecting signs of depression among individuals in the workplace. The dataset comprises detailed information on the signs and symptoms of depression among individuals, particularly observed negative consequences at the workplace, physical health issues and their negative consequences, treatment, etc. Even though Random Forest and Gradient Boosting Machine were found to have the same accuracy but still Gradient Boosting Machine was chosen as the better model because

the precision for class 0 from Gradient Boosting Machine was higher than the Random Forest precision. It also is found to have a lower recall for class 0 than Random Forest and recall for class 1 is better than that of Random Forest, the F1 score for class 1 for Gradient Boosting Machine is better than that of Random Forest. The Gradient Boosting Classifier has a slight edge in this comparison based on the provided metrics, but the final decision should consider the specific context and cost of errors in the application for which the model is intended. This inquiry emphasizes the potential of machine learning to revolutionize the field of depression care through the timely identification of mental health concerns. The application of artificial intelligence demonstrates the ability to enhance precision, efficiency, and promptness in the realm of diagnosing any issues. While there are multiple methods to assess an individual's mental well-being in a professional setting, AI can furnish a more impartial and precise evaluation. The outcomes of this analysis clearly portray the potential of machine learning to mitigate the diagnosis of depression and other mental health disorders. The conclusions drawn in this study carry significant implications for the future of mental health care. By promptly identifying mental health problems, we can offer timely and efficacious interventions that can enhance the quality of life for individuals afflicted with depression. The integration of machine learning in depression care can contribute to mitigating the negative associations linked to mental health disorders and fostering a more constructive and supportive work environment.

Acknowledgments. Thanks to Center for Research, Christ University for provide all the computational facility for doing this research.

References

1. Ochi, S., Dwivedi, Y.: Dissecting early life stress-induced adolescent depression through epigenomic approach. *Mol. Psychiatry* **28**(1), 141–153 (2023)
2. Sebera, F., et al.: Association of depression and epilepsy in Rwanda: a prospective longitudinal study. *Epilepsy Behav.* **138**, 108993 (2023)
3. Sarfan, L.D., Zieve, G., Gumport, N.B., Xiong, M., Harvey, A.G.: Optimizing outcomes, mechanisms, and recall of cognitive therapy for depression: dose of constructive memory support strategies. *Behav. Res. Ther.* **166**, 104325 (2023)
4. Zhou, M., et al.: Microbiome and tryptophan metabolomics analysis in adolescent depression: roles of the gut microbiota in the regulation of tryptophan-derived neurotransmitters and behaviors in human and mice. *Microbiome* **11**(1), 145 (2023)
5. Smit, A.C., Snippe, E., Bringmann, L.F., Hoenders, H.R., Wichers, M.: Transitions in depression: if, how, and when depressive symptoms return during and after discontinuing antidepressants. *Qual. Life Res.* **32**(5), 1295–1306 (2023)
6. Ren, X., Yu, S., Dong, W., Yin, P., Xu, X., Zhou, M.: Burden of depression in China, 1990–2017: findings from the global burden of disease study 2017. *J. Affect. Disord.* **268**, 95–101 (2020)
7. Ren, X., Yu, S., Dong, W., Yin, P., Xu, X., Zhou, M.: Burden of depression in China, 1990–2017: findings from the global burden of disease study 2017. *J. Affect. Disord.* **268**, 95–101 (2020). <https://doi.org/10.1016/j.jad.2020.03.011>

8. Soga, M., Evans, M.J., Tsuchiya, K., Fukano, Y.: A room with a green view: the importance of nearby nature for mental health during the COVID-19 pandemic. *Ecol. Appl.* **31**(2), e2248 (2021)
9. Sree, S.R., Vyshnavi, S.B.: Real-world application of machine learning and deep learning. In: 2019 International Conference on Smart Systems and Inventive Technology (ICSSIT) (pp. 1069–1073). IEEE (2019)
10. Shen, F., Zhao, X., Kou, G., Alsaadi, F.E.: A new deep learning ensemble credit risk evaluation model with an improved synthetic minority oversampling technique. *Appl. Soft Comput.* **98**, 106852 (2021)
11. Yetukuri, N.K., Maddali, K.: Machine learning based plant disease identification by using hybrid naïve bayes with decision tree algorithm. In: 2023 5th International Conference on Smart Systems and Inventive Technology (ICSSIT) (pp. 852–858). IEEE (2023)
12. Wright, L.G., et al.: Deep physical neural networks trained with backpropagation. *Nature* **601**(7894), 549–555 (2022)
13. Keshwani, H.: Bioinformatics research challenges and opportunities in machine learning. In 2022 International Conference on Augmented Intelligence and Sustainable Systems (ICAISS) (pp. 290–295). IEEE (2022)
14. Tadalagi, M., Joshi, A.M.: AutoDep: automatic depression detection using facial expressions based on linear binary pattern descriptor. *Med. Biol. Eng. Compu.* **59**(6), 1339–1354 (2021)
15. Sharma, G., Joshi, A.M., Gupta, R., Cenkeramaddi, L.R.: DepCap: a smart healthcare framework for EEG based depression detection using time-frequency response and deep neural network. *IEEE Access* **11**, 52327–52338 (2023). <https://doi.org/10.1109/ACCESS.2023.3275024>



Patient Sickness Predictive System Using Chatbot

Dappili Bharat Kumar Reddy¹ , Vineet Dasari¹ , C. Madhu Kumar¹ ,
Mmihiraansh¹ , Naga Venkata Yaswanth Lankadasu¹ ,
Devendra Babu Pesarlanka¹ , Ajay Sharma² , and Shamneesh Sharma²

¹ Lovely Professional University, Phagwara, Punjab, India

² upGrad Education Private Limited, Bangalore, Karnataka, India

ajaysharmadharmani@gmail.com

Abstract. Training patient preventive approaches, the research involved decision trees and support vector machine (SVM) algorithms in a medical chatbot architecture. The demand for healthcare solutions is always growing because of the rise of chatbots as one of the most dependable means that users can have initial diagnoses and medical advice. In this study, the author has used natural language processing (NLP) to predict diseases using symptoms provided by patients and recommend appropriate medications for them, to compare the predictive capabilities of decision trees and SVM algorithms against each other. When it comes to medical chatbots, the suggested SVM algorithm outperforms decision trees both in disease prediction and medication recommendation. A machine learning-based system that utilizes advanced methods such as SVM enhances the chatbot's diagnostic ability thus enabling better diagnosis while also giving timely recommendations on drugs. This study represents a contribution to AI-driven healthcare solutions aimed at improving patient outcomes while reducing the costs of healthcare. Decision trees, SVM, and KNN models are used in the research with an accuracy of around 97% at the time of model development.

Keywords: Deep Learning · Chatbot · Natural Language Processing · Machine Learning · Decision trees · SVM · Healthcare

1 Introduction

Health is concerned with the person's physical and mental well-being. If a person is physically feeling well-being, then there might be a chance that the person is suffering from mental sickness [1, 4]. In India clearly, about 10–15% of people who are working in any sector (Public/Private) cannot visit or make an appointment with their doctors due to their busy life or encumbered work. So, to check their health status they can immediately visit the available online resource by which the bots can make sure to treat the basic sickness or if required they can book appointments with the doctors which are recommended by a chatbot. The research aims to explore the possibilities of machine learning (ML) algorithms decision trees and support vector machines (SVM) for developing a medical chatbot. The goal of this chatbot is to enable individuals to take responsibility for their

health by prompting the dialogues which act as a chatbot and help to encourage the users to share their health-related issues. The chatbot analyzes the symptoms by using natural language processing (NLP) to detect, and identify the primary, key symptoms obtained from the user interactions. Predictive modeling is a technique that uses the decision tree and the Support Vector Machines (SVM) which extract the symptoms to predict the possible diseases [3]. There are some of the suggestions that are used consequently for the chatbot's to provide an appropriate recommendation including some of the self-help measures or even referrals for further consultations with a doctor. The methods include 24/7 accessibility, and the availability of the chatbot would eliminate inconvenient hospital visits. Users can engage in conversation/chatting with bots while going about their daily activities without interruption. The initial consultation fee is done away with by the robot. Symptom detection is promoted by the chatbot thereby encouraging early diagnosis and leading to better healthcare outcomes.

The current research will evaluate the effectiveness of decision trees and SVMs in building a user-friendly and accurate medical chatbot system. The main goal is to contribute to a hands-on approach to healthcare, empowering individuals to take charge of their well-being. With the help of this method, a medical chatbot is designed to engage users, in discussions, about health matters by focusing on the symptoms they mention [2, 3, 5]. Through addressing these symptoms, the chatbot can pinpoint the ailment, and recommend an approach. The method employed in this scenario is known as the K-Nearest Neighbor (KNN) algorithm [3, 15].

A chatbot in medical science uses natural language processing can identify and diagnose patient symptoms with good accuracy by classifying simple symptoms and building successful conversations. The impact is huge as a result increase in the solidity and less prone to human errors. Nowadays most people are addicted to the internet but they do not care about their health. People avoid going to the hospital for minor problems that may turn into serious problems in the future. The idea is to build a chatbot that is free and can be used 24/7 anytime. The fact that the chatbot is free of cost and accessible whenever the user wants (if his environment is running) encourages customers to use it [16]. It reduces the cost of consulting a specialist or a doctor. With the new guidelines, fewer people will neglect their health due to the difficult hospital appointment process. People can interact with the chatbot and continue their other work just like other people. It ensures that their work is uninterrupted and effective. This ensures a way to help people improve their health by using chatbots to help people understand their health and thus take a greater role in treatment [7].

The main aim of the system is to promote health awareness and provide a chatbot that encourages users to embrace lifestyles. The collaboration, between healthcare and chatbots has been beneficial for nurturing an environment that mimics interaction to enhance communication between users and bots [18]. Discussing health within this system serves as a tool for users striving for wellness. Given that this forum remains confidential utilizing human resources to instill a sense of security is vital, especially in scenarios involving psychology.

The system is crafted to offer assistance without constraints, on time finances as it is consistently accessible and free of charge. Users can swiftly receive a symptom-based diagnosis that sheds light on health issues [17, 19].

2 Research Work

Health is one of the most important concerns of human life. Health plays a vital role for the all-around development and to make a good, prosperous life. To make a chatbot for health-related issues numerous literatures are available. For the current work, the author has used the literature survey and incorporated some of the important ones for the development of the chatbot for the health care analysis.

Darius and Sophie [3] found that the word “Chatbot” consists of the words “chat” and “robot”. Originally, the term “chatbot” referred to computers that simulate human conversation with the help of text-based conversations. Chatbots have inputs and outputs that allow users to communicate with the chatbot software, giving the impression of talking to a real person. The concept of competitive programming like arrays, queues, stacks, trees, etc. was used to train the ML model to remember many types of data points, so it can respond to users based on its knowledge. At a basic level, there are two types of chatbot: non-intelligent (using pre-speech) and intelligent (using machine learning). Chatbots can be built using different algorithms, the most commonly used algorithms are support vector machines(SVM) classifier, naive Bayes, and K-Nearest Neighbors.

The chatbot includes NLP so that the chatbot responds based on information received from the user. Nadarzynski and his colleagues also pointed out that artificial intelligence is software developed by complex mathematics that can process data to produce specified results and thus create an impact effect [8, 20], social workers have been replaced by chatbots for basic diagnosis and can interact with users by using natural language processing, even designed to save time and money, the use of chatbot intelligence often fails to meet customer expectations, users may cause a request not to comply with the chatbot.

The author Mathew and colleagues [9] say that natural language processing (NLP) can help computers understand and classify the words people use, as well as identify them during speech and receive responses. The author has achieved comparatively good accuracy (~95%) as compared to the previously available models that are using NLP & machine learning approaches. Some of the main features are word segmentation, stop word removal, and lemmatization.

Lekha and colleagues define tokenization [10] as the process of splitting sentences into words removing symbols and finally removing stops, that is, removing unnecessary words so that we can collect information for the next step in the process. Chatralli et al. In [11], the removal of stop words “if”, “like”, etc. Lemmatization is then used to obtain the base word of each token.

3 Design and Methodology

Users engage with chatbot applications just like they would interact with individuals. When using this web platform users are required to log in after completing the registration process on the available platform. Through this conversation, the system(chatbot) discovers the user’s symptoms. The customer(user) sends a message in the chat box and the chatbot gives the response with the appropriate message. Chatbots attempt to understand the language spoken by people by using natural language processing, classify it, analyze and respond when it is necessary. When a question arises the chatbot

will attempt to find the question, from its learnings much like how KNN works. Python offers a range of functions tailored for NLP purposes. Among these libraries is Natural Language Toolkit (NLTK) which provides features for NLP tasks. Tokenization plays a role, in converting user input text into tokens. This process involves transforming words into token forms which can then be utilized for further analysis or validation processes [11]. Once tokens are created from a word, it is lemmatized and divided into roots. Radical separation and lemmatization were made to simplify the words. They represent important steps in natural language processing.

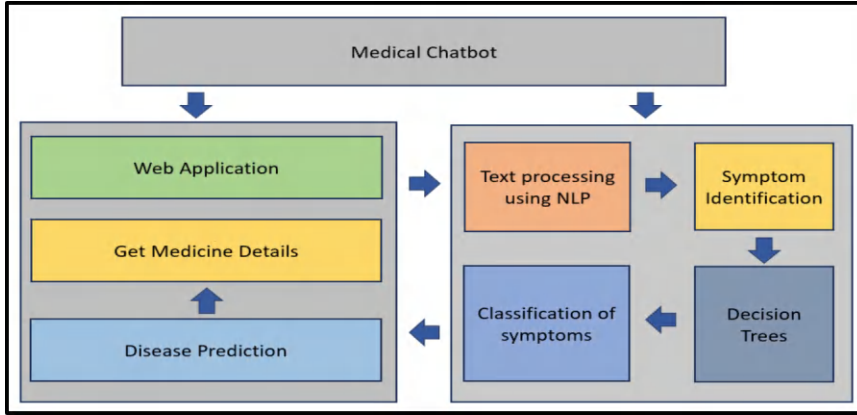


Fig. 1. Explains the design flow of the model where the user opens the web application interface, and the end user tries to log in after that the patient can enter the symptoms that they are experiencing after the whole classification of symptoms the user will get the disease what they are experiencing and also the prescribed medicine.

Stemming abides, by guidelines eliminates prefixes or suffixes in tags. This process helps in reducing a word to its form [12]. A bag of words (BOG) is formed when a word is reduced to its base form. Similarly, the data points containing symptoms are transformed into a bag of words representation. This representation consists of values (0s and 1s) with each vector being assigned an identification number. The system employs machine learning techniques. The chatbot utilizes K neighbors (KNN) to answer user queries. K nearest neighbor (KNN) is an efficient machine learning algorithm utilized in the current work serving as a method, for identifying patterns. It is a classification algorithm that is mostly used when the data structure is input, in this case, most of the people whose symptoms are closest to k are classified according to their class labels [13]. Splits sample data and keeps it into different categories based on the dataset labels. When a new data model is obtained, it is classified according to its similarity with other data models [6]. In practice, there may be any number of approaches, the definition can be expanded to include many features. For example, if we consider the four diseases closest to headache, they are classified as appropriate diseases according to other diseases or symptoms. KNN can accurately classify symptoms according to user chats and assign them to related diseases on the dataset. When a user asks a question, the query is processed by the text function [14]. Transformed into a vector (collection of words). This vector is

then inputted into the model to generate a response measurement. The chatbot will then suggest the illness, to the user. The Fig. 1 demonstrates the working methodology of the chatbot. Offer a link for them to explore suitable treatment options for the identified condition.

4 Proposed Innovation

In present life, chatbots are very helpful for everyone who needs them for their wellbeing. Users can interact with the chatbots and get their diagnosis accordingly and this chatbot helps the users to analyze the symptoms and update their symptoms accordingly if required the bot can suggest doctor consultancies. The chatbot will get the users the most experience using it and it is also helpful for the encouragement of chatbots in the next future. The Fig. 2 represent the frontend page of the application where the user can be able to log in. The page and see the work in the medical field and interest in medicine. The website allows the user to log in and sign in after that the user can get the previous details of their health status after that user can go and register on the website and the user can interact with the bot by using natural language processing the bot can interact and predict the symptoms of the user.

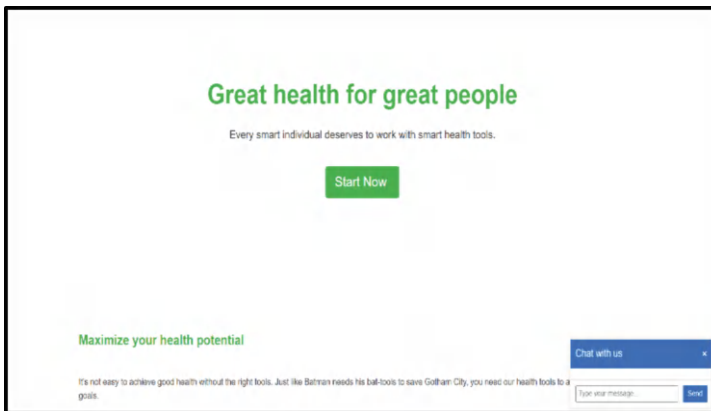


Fig. 2. Frontend page showing the Chat-bot interface

In the bottom right corner, the chatbot option is available where the users can interact with the chatbot and answer the simple questions asked by the chatbot by which the chat using the algorithm can predict the simple disease and the prescribed medicines. Chatbot analyses the user's symptoms and by processing the algorithms it gives the cause for the user's sickness and also the precautions for the sickness. The Fig. 3 shows the enhanced, modified version of the chatbot used in the current research work, simultaneously Fig. 3 shows the real-time communication of the chatbot(Robo-Doc) giving the live recommendation for the disease based on the user input parameters.

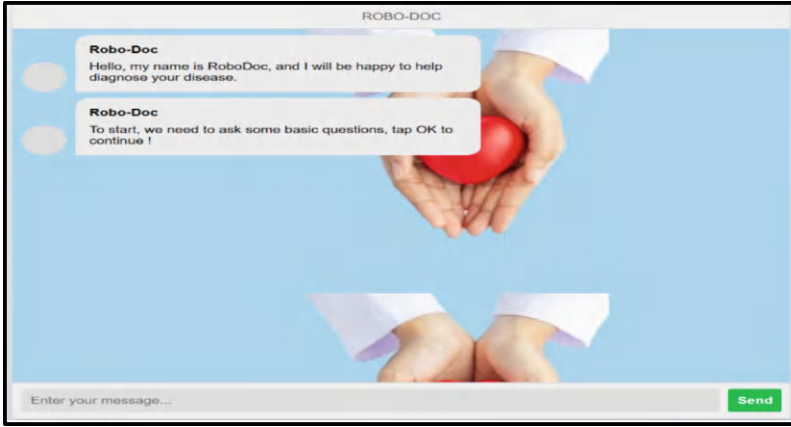


Fig. 3. Chatbot interface for the communication with doctors

5 Experiments

The author has created a dataset by contacting the local hospitals and accessing the required data from the doctors as it helped develop a chatbot application, tried on people who are sick by using this chatbot they can get their health status. It depends on what type of illness they are having as they are having itching and then symptoms like rashes, fatigue, dryness, and more, these symptoms are analyzed based on the dataset and can able to predict the correct symptoms for the people. The available machine learning algorithms are used for the chatbot prediction decision trees, KNN, and support vector machines with an accuracy of 97%. The distance matrix formulas in KNN are used to perform the analysis of the data set.

$$E = - \sum p_i \log_2(p_i) \quad (1)$$

$$\text{Distance} = \sqrt{(x_2 - x_1)^2 - (y_2 - y_1)^2} \quad (2)$$

As the equation (i) is used in KNN it is used to find the distance between the grouping based on the centroid and it is a Python library. The KNN model is trained on the specific dataset and tested on the outside data with the dataset present 75% was used in training and the remaining in testing. The equation shows how the model is used to create and by pickle, it would save the model and can be used again.

6 Working

Innovations in healthcare, and disease prediction systems using chatbot machine learning technologies methods are ready to change diagnostic capabilities, and empower patients' care. The research focuses on a combination of a prominent machine learning method k-nearest Neighbor (k-NN) and a decision tree in Chatbot. The framework provides efficient and accurate predictions and diagnoses of the patient based on symptoms and questions.

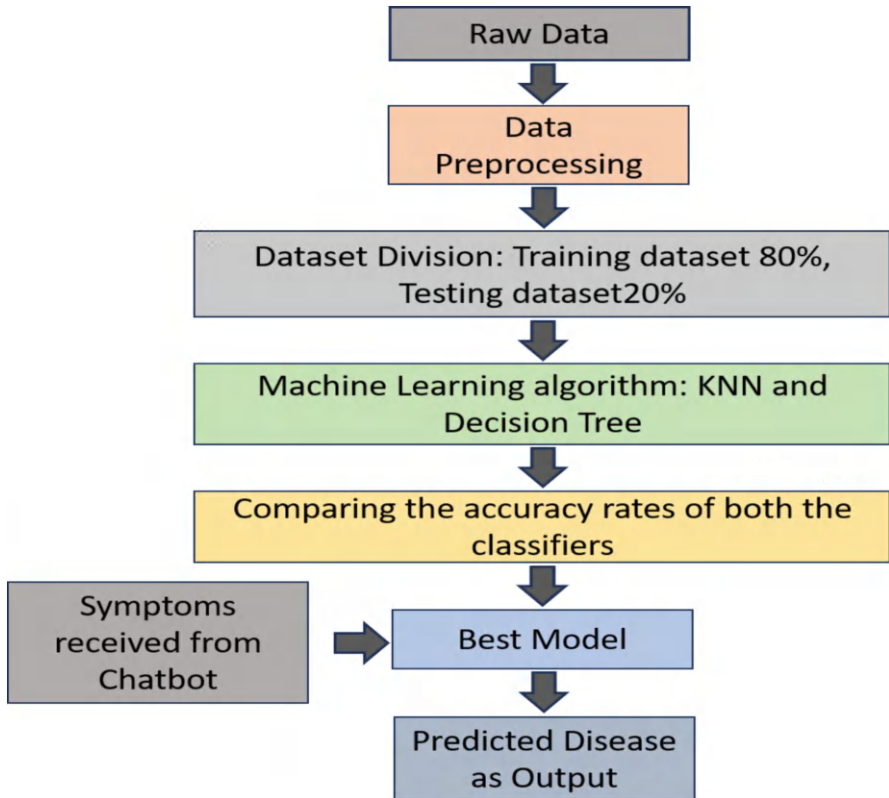


Fig. 4. The working methodology for the chatbot healthcare predictive care analysis

The k-Nearest Neighbor (k-NN) algorithm is a direct yet powerful approach to train the machine learning model. It acts so that the patient shows symptoms or terms as feature vectors in multidimensional space, where each patient dataset is represented as a node according to their respective characteristics. When new patient inquiries are received, the KNN algorithm determines the nearest neighbor based on similarities (similar patient conditions). The determined by suspected disease or diagnosis majority class (classification) among these neighbors or a weighted average of attributes (regression). This is its method offers simplicity and adaptability that requires minimum basic assumptions of data distribution. Decisions are dictated by the patient's symptoms, questions from the question bank (database), and each table is suitable for suspected disease or condition. This model allows you to see and understand the structure in detail diagnostic function that allows the physician or user to explain how assumptions are made. The Fig. 4 describes the complete working methodology for the development of the chatbot for predictive health care analysis.

The combination of K-NN and decision tree chatbot prediction frameworks provides several advantages. First, it improves the accuracy and reliability of diagnostics diagnosing sufferers based on signs and symptoms or questions in real time. Second, the

readability and interpretation of the Decision Trees allow healthcare carriers to recognize and trust diagnostic outcomes to support effective decision-making. Here the decision trees are used in the chatbot for the recursive calls. So that when there is required to ask for the following next symptoms then the chatbot uses the recursive calls in the stack to work through and provides us with the valid output. Sometimes it leads to more time complexity which can be handled by using exception handling and also proving the accurate data to the chatbot.

7 Results

Using KNN, decision trees, and support vector machine algorithms with an accuracy of around 97% users can say that the bot is trustworthy. Accurate disease can be predicted for symptoms entered by the user and therefore can be reliable and used for health monitoring. People’s busy lives require trust. Once the results are seen, one can be hopeful for this new process.

```
Your Name?                                ->bharat
Hello, bharat

Enter the symptom you are experiencing      ->itching
searches related to input:
0 ) itching
1 ) internal_itching
Select the one you meant (0 - 1): 0
Okay. From how many days ? : 5
Are you experiencing any
itching ? : yes
skin_rash ? : yes
stomach_pain ? : no
burning_micturition ? : yes
spotting_ urination ? : no
It might not be that bad but you should take precautions.
You may have Drug Reaction or Chronic cholestasis
```

Fig. 5. Chat bot predictive care analysis results working/trained machine learning model.

In the above Fig. 5 the corresponding chatbot gives the output for desired results in which where you can see it first asks name and then according to the sickness it particularly identifies the sickness for the customers or users (Fig. 6).

Type of AI: Virtual Assistant:

- Function:** Performs a preliminary diagnosis or triage of patients by asking a series of questions about their symptoms.
- Benefits:** Could improve efficiency in healthcare systems by allowing medical professionals to focus on more complex cases.
- Limitations:** Reliant on accurate user input and may not be able to account for all possible symptoms or medical conditions.

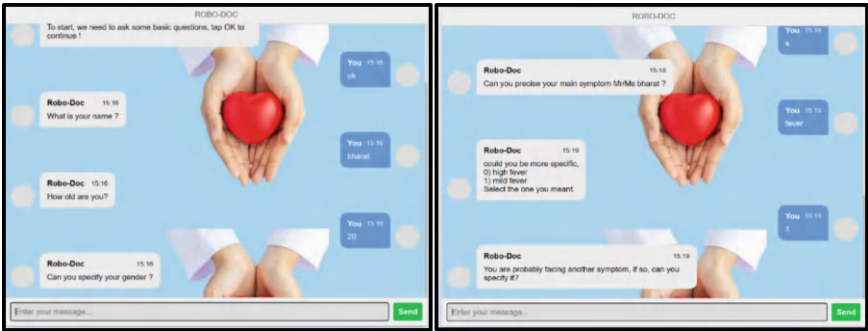


Fig. 6. Initial chat interactions with the human

It is important to note that Robo-Doc is not a real doctor and cannot provide medical advice. If you are seeking any immediate medical emergency, please reach for professional help immediately (Fig. 7).

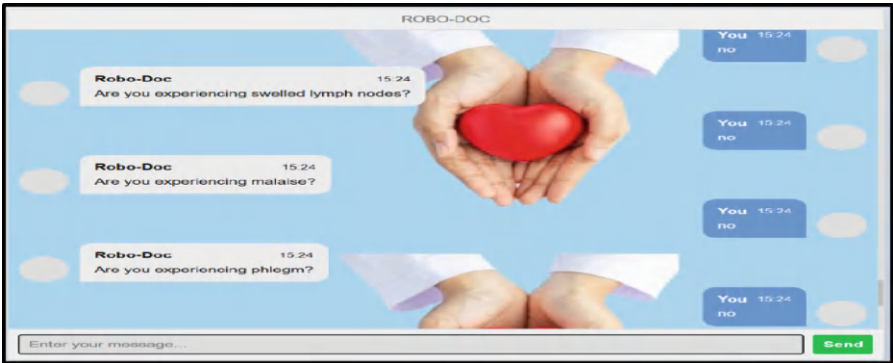


Fig. 7. Chatbot is showing up the symptoms for the disease analysis.

The illness prediction method used for the interview performed exceptionally well, receiving an astounding 97% accuracy rate in the assessment. This accuracy metric demonstrates the chatbot’s usefulness in supporting diagnostic processes by indicating its capacity to accurately diagnose and anticipate diseases that are based on the patient’s symptoms and needs. The chatbot keeps on offering individualized interventions and preventive measures based on each disease prognosis after it has been made. These recommendations are based on professional medical guidelines including lifestyle modifications, drug schedules, and cleaning procedures to support proactive approaches to illness prevention and health management.

The Fig. 8 describes and shows the chatbot recommendations, as well as the precautions so that the user can find it helpful. The below Table 1 has given a comparison of different models used for this project:

User feedback and satisfaction surveys have demonstrated that interview advice is meaningful, and relevant, and patients have reported understanding, and comprehension



Fig. 8. Chat bot recommendations based on the patient input details

Table 1. Different machine learning algorithms showing the different performance measures (Accuracy, Precision, Recall, and F1 Score)

Model	Accuracy	Precision	Recall	F1 Score
Decision Tree	98.46%	0.97	0.96	0.96
K-Nearest Neighbor	97.39%	0.96	0.98	0.97
Support Vector Machines	97.78%	0.98	0.95	0.96

of illness-preventive techniques. This encouraging reaction has an impact on the chatbot’s capacity to actively engage users and enhance health outcomes while lowering the complications of the disease. The 97% accuracy rate attained not only demonstrates the efficacy of integrating machine learning techniques into a chatbot framework for illness prediction but also establishes the chatbot as a useful instrument for the development of proactive and individual healthcare management. Future developments in digital health technology and the provision of individualized health solutions will be fueled by the ongoing and gradual expansion of chatbot capabilities, which will also improve clinical utility and patient care.

8 Conclusion

The article presents a medical chatbot that can be used to improve the diagnostic process and treatment recommendations. Chatbots cannot be doctors and cannot replace doctors also. The chatbot follows the user’s request. Users of this app can share their symptoms with the chatbot, which suggests health precautions should be taken. The dataset provides comprehensive information about the related symptoms and diseases, so the condition chatbot can provide users with information about diseases and treatments. The chatbot analyzes the symptoms and can predict the disease to the user and provides detailed information about the disease and the precautions.

Medical chatbots can be used to make an accurate diagnosis and provide appropriate treatment by analyzing the symptoms described by the patient. People rarely go to the hospital for check-ups in their busy lives. Chatbots are very important in this regard because they provide one-click diagnostic assistance. The advantages of the chatbot is that it does not need the help of a doctor to provide a health assessment to the user.

In addition, the free cost of using chatbots is a great attraction for users. The conversations with users are completely encrypted and personal, they help users learn more about their health problems and ensure that the chatbot correctly identifies the virus and organizes the content on a relevant hierarchical basis. The title of the document is the title of the topic because all subsequent documents address and explain that topic. If there are two or more elements, the next level (Roman numerals) should be used, if there are not at least two elements, they should not be used and conversely.

Simply this chatbot is very helpful for providing the necessary information regarding the patient's sickness or disease and also the necessary precautions by which the users can focus on their health in their busy lives. All the supplementary details and the working model are deployed on the following link (<https://github.com/bharatdappili/chatbot>). The online deployment of the Mobile application and the online web version is available in the development mode and can be communicated in the future.

9 Future Scope

The role played by chatbot may extend beyond resources, and users should consult a doctor for the relevant diagnosis. In this case, a chatbot can help if scheduling with the doctor can be done effectively according to his or her schedule.

Looking forward, the research opens an exciting street for the improvement of affected person prognosis systems integrated with chatbots. Key regions of future research encompass the integration of advanced device-studying fashions, inclusive of neural networks and deep getting-to-know architectures, to enhance the accuracy and reliability of disorder predictions. By the usage of transfer learning strategies, we can adapt pre-educated fashions for healthcare databases, enhancing the chatbot's ability to generalize across exclusive patient populations.

Real-time facts integration and continuous machine learning algorithm mechanisms constitute a promising path. Implementing those capabilities will permit the communique to dynamically research and adapt based on evolving affected person facts and rising fitness traits, making sure modern-day, powerful predictive abilities over the years. Scalability and compatibility with existing electronic health record (EHR) systems are critical to deploying predictive analytics in a variety of healthcare settings, from hospitals and clinics to telemedicine platforms and mobile health apps. Research studies and clinical trials will be important in evaluating the real-world effectiveness and impact of chatbot predictive systems on patient outcomes and healthcare delivery.

In summary, the future scope of the research includes extensive advances in machine learning, including real-time data integration, personalization, validation studies, ethical considerations, and scalability. By taking advantage of this opportunity, we aim to drive transformational change in personalized patient care, disease prevention, and health management, resulting in improved health outcomes and improved quality of life for people and communities around the world.

References

1. Tarigoppula, V.S.S., et al.: Intelligent Parkinson disease prediction using machine learning algorithms. *Int. J. Eng. Innov. Technol.* **3**(3), (2013)
2. Gomathi, K., Shanmuga Priya, D.: Multi Disease Prediction using Data Mining Techniques
3. Jamgade, A.C., Zade, S.D.: Disease prediction using machine learning. *Int. Res. J. Eng. Technol.* **06**(05), (2019)
4. Benjamin Fredrick David, H.: Impact of ensemble learning algorithms towards accurate heart disease prediction
5. AbuShawar, B., Atwell, E.: ALICE chatbot: trials and outputs. *Computación y Sistemas* **19**(4), 625–632 (2015). <https://doi.org/10.13053/cys-19-4-2326>
6. Sheikh, S.A.: Artificial intelligence based chatbot for human resource using deep learning (2019)
7. Manjurul Ahsan, M., Siddique, Z.: Machine learning based disease diagnosis: a comprehensive review. *arXiv e-prints*. 2021:arXiv: 2112.15538
8. Caldarini, G., Jaf, S., McGarry, K.: A literature survey of recent advances in chatbots. *Information* **13**(1), 41 (2022)
9. PhaniRaghavaa, B., Kumarb, S.A.: An improved Chatbot for predicting disease and medicines using natural language processing with fuzzy logic. *Adv. Parallel Comput. Algorithm. Tools Paradigms* **41**, 258–264 (2022). <https://doi.org/10.3233/APC220035>
10. Mehfooz, F., Jha, S., Singh, S., Saini, S., Sharma, N.: Medical chatbot for novel COVID-19. In: *ICT Analysis and Applications*, pp. 423–430. Springer, Singapore (2021)
11. Chen, M., Hao, Y., Hwang, K., Wang, L., Wang, L.: Disease prediction by machine learning over big data from healthcare communities. *IEEE Access* **5**, 8869–8879 (2017)
12. Lalwani, T., Bhalotia, S., Pal, A., Bisen, S., Rathod, V.: Implementation of the ChatBot System using AI and NLP (2018)
13. Yu, H.: Experimental disease prediction research on combining natural language processing and machine learning, pp. 145–1500 (2019). <https://doi.org/10.1109/ICCSNT47585.2019.8962507>
14. Akhil, J., Samreen, S.: Heart disease prediction system based on hiddennaïve Bayes classifier (2016). <https://doi.org/10.1109/CIMCA.2016.8053261>
15. Enriko, I.K., Suryanegara, M., Gunawan, D.: Heart disease prediction system using k-nearest neighbor algorithm with simplified patient's health parameters **8**, 59–65 (2016)
16. Jwala, K., Sirisha, G.N.V.G., Padma Raju, G.V.: Developing a Chatbot using Machine Learning (2019)
17. Nair, A.: Overview of Tesseract OCR engine (2016)
18. Babu, R., Jayashree, K.: A survey on the role of IOT and cloud in health care. *Int. J. Sci. Eng. Technol. Res.* (2015)
19. Divya, S., Indumathi, V., Ishwarya, S., Priyasankari, M., Kalpana Devi, S.: Survey on medical self-diagnosis chatbot for accurate analysis using artificial intelligence. *Int. J. Trend Res. Develop* (2018)
20. Vivek, K., Vitthal, S.S., Gutte, V.S.: Intelligent Healthbot for transforming healthcare. In: *Proceeding of National Conferences on Machine Learning*, Pune (2019)



EPC-CCM Based Saliency Equivalent Mechanism for Multimodal Medical Image

Nalini Manogaran¹, Resmi R. Nair², Suresh Annamalai³ ,
Balamurugan Balusamy⁴, Parimala Veluvali⁵, and Sushama Tanwar⁶

¹ S.A. Engineering College (Autonomous), Chennai, Tamil Nadu 600077, India

² Saveetha Engineering College (Autonomous), Chennai, Tamil Nadu 602105, India
resminair@saveetha.ac.in

³ SRM Institute of Science and Technology, Chennai, Tamil Nadu 603203, India
prisuresh@yahoo.com

⁴ Shiv Nadar (Institution of Eminence Deemed to Be University), Greater Noida, Uttar Pradesh 201314, India

⁵ Symbiosis International (Deemed University), Pune, Maharashtra, India

⁶ Manipal University Jaipur, Jaipur, Rajasthan 303007, India
sushama.tanwar@jaipur.manipal.edu

Abstract. Multimodality based medical image fusion is a developing technology in which medical images are fused with complementary details. In the existing multimodal fusion techniques, low contrast and color distortions are caused during image integration. Preserving significant detail in best contrast is still challenging one. In this work, a unique technique for multimodality based image fusion is proposed based on Essential Patch Corrosion (EPC) and Effective Compatible Cloud Model (CCM). First, the source images are decomposed into multiple sub-images in which the EPC technique is applied to retrieve salient features for discrimination. Second, the CCM is applied to fuse the significant details from the images. Additionally, Saliency Equivalent Mechanism (SEM) along with CCM is used to merge the residual details. Finally, reconstructing the image is done using inverse EPC and the experimental demonstration for this proposed system is given. The proposed work shows the improvement significantly with maximum contrast and minimized color distortion in the target images.

Keywords: Medical Image Processing · Essential Patch Corrosion · Saliency Equivalent Mechanism · CCM · Image mapping

1 Introduction

In clinical diagnosis, medical images from various modalities are used. In the field of medical based image fusion, the low cost computations and imaging methods are rapidly grown. Single modality images are not gives required information and has limited details and also it doesn't meet the requirements to diagnose complex diseases [6]. So, the medical image fusion is incorporated in the field of medical in which multimodal images are fused together to extract the comprehensive information [3]. For instance,

the images from MRI modality serves the soft tissue details with high spatial resolution. The CT modality images serves the hard tissue related details with limited distortions. The PET modality images gives the details about quantitative and metabolic related substances. The SPECT modality image serves the metabolism changes occurred in the human body. Each modality has its own features to provide image details and also limitations to provide required information. It is mandatory to merge different modality images into a solitary image to obtain the comprehensive details [7].

Fusing medical images maximizes the quality of image which leads to easy diagnosis and assessment. Generally, the fused images are not providing accurate data for diagnosing and it reduces the redundancy produced by the modalities which produces single images [10]. With the enormous improvements in this era, a number of techniques were proposed for image fusion. Image decomposition and reconstruction are considered as preprocessing of images and applying fusion rules are the procedures for image fusion. During image preprocessing, the images are decomposed into sub modules and extracting significant features are done [1]. The Pyramid transformation, sparse representation transformation, wavelet transformation and salient features transformations are the commonly used decomposition techniques. The pyramid based decomposition converts different image resolutions with iterative style [11]. In the wavelet transformation, the mathematical calculations are used and it is a widely used technique. Discrete wavelet transform is a familiar method with time based frequency [12]. However, the above mentioned methods are not provided significant required information and doesn't retain shift invariance. The contourlet, shearlet and non-subsample contourlet based transformations are proposed to resolve the drawbacks of the above methods.

In the traditional decomposition techniques, the SR (Sparse Representation) was used to merge the medical images accurately and reconstruction of images are done with linear combination of coefficients. The Joint SR and Group SR methods were developed in the traditional SR techniques [4]. Salient feature techniques are totally different from other techniques in which shift invariance, retaining salient details and low computations are done [5]. Edge-preservation based filters were widely used in fusing images based on the features of salient. Guided Filter, Gradient smoothing filter and Bilateral filter are some familiar filters used in this edge preserving techniques. Salient feature techniques are more capable than wavelet and sparse techniques as it has low computations [13]. However, this technique has side effects like contrast damages, blocking and shadowing effects.

Image fusion protocols refer to algorithms which seeks efficient features in images and reduces other unnecessary details. An efficient fusion protocol plays a vital role in objective quality target images. In traditional fusion protocols have statistics, human based visual methods, objective assessment metrics, ML (Machine-Learning) and fuzzy techniques. Fuzzy logics are the recent technology schemes proposed by the a number of researchers in recent years. Image segmentation and edge recognition are the most covering areas for applying fuzzy techniques [9]. It is considered as an extension of Boolean logic which provides an area for manipulating borders definition. If uncertainty and no mathematical mappings are found in the image processing, the fuzzy techniques are applied [9].

Fuzzy logic based fusions produce better results than other classical medical image fusion techniques. A simple common fusion technique based on intuitionistic fuzzy mechanism was proposed. This technique obtains target images with high luminous as well as contrast and color based medical images are not able to fuse [14]. The fuzzy transformation techniques were proposed for the multi scale based geometric analysis for NSCT. These techniques enhance the information of target images and increase the visuals with small variations when compared to other familiar fusion techniques [2]. Based on these explorations, Essential Patch Corrosion is used to decompose the source images and CCM along with SEM techniques are incorporated to obtain the salient features. Finally, the target image is constructed by a weight-based approach during the final mapping stages. The proposed method explores various novel techniques for image fusion, each with its own potential for advancement in the field. One of these techniques is Essential Patch Corrosion (EPC), which aims to introduce a new image decomposition method. The novelty of EPC lies in its differentiation from existing decomposition methods and effectiveness in image fusion. Another technique, the Effective Compatible Cloud Model (CCM), offers a significant contribution by combining features and compatibility weighting in a new approach to image fusion. The core novelty of CCM lies in its concept and potential effectiveness. The Saliency Equivalent Mechanism (SEM) could be new because it improves the fusion process by adding saliency information and adjusting it based on how important different image regions are. Fuzzy fusion is an optional step in CCM that can lead to new ideas by suggesting specific fuzzy logic systems that are designed to work well with image fusion and how to effectively integrate them with CCM. These techniques offer exciting possibilities for innovative advancements in the field of image fusion. The proposed technique is applied to numerous medical images and results are evaluated and compared with other familiar fusion techniques. Experiment results are demonstrated in the following sections which obtain better output than the existing techniques. Subsequent paragraphs, however, are indented.

2 Related Work

Traditional multimodal medical image fusion techniques face significant challenges that hinder accurate diagnosis, such as incomplete information extraction, artifact introduction, and lack of adaptability. Techniques like Wavelet Transform suffer from shift variance, leading to missed crucial details, while Sparse Representation often loses image quality during reconstruction. Salient feature techniques may introduce artifacts, causing contrast distortions and shadows that mislead medical professionals. Traditional fusion protocols struggle with adaptability, are subject to human bias, and cannot balance contrast, color distortion, detail preservation, and artifact reduction effectively. In contrast, the proposed SEM system addresses these issues by capturing a comprehensive representation of source images, minimizing artifact introduction, and adapting to varying image characteristics, ensuring consistent and reliable fusion quality. By leveraging objective evaluation metrics, SEM reduces reliance on subjective human assessments and prioritizes detail preservation, contrast optimization, and artifact reduction for clearer, more informative fused images. Compared to modern approaches like deep learning-based fusion, which is computationally expensive and data-intensive, and Canonical

Correlation Analysis (CCA), which may not capture the most relevant features for diagnosis, SEM offers a more efficient and potentially more accurate alternative, focusing on highlighting salient features crucial for medical diagnosis.

3 Proposed Method

The proposed technique is elaborated in this section. Consider the two images X and Y which are obtained from two different modalities and those images are clinically registered. The preprocessing images that is image decomposition is done by using EPC and signal assets and average intensity of all images are retrieved. Then CCM along with SEM, two fuzzy fusion techniques are applied to build a supplementary mapping in order to avoid the discrepancies affected by the improper mappings. The CCM is used to build the screening which fuses the tri-state map, and the second fuzzy technique is to merge the two different maps into supplementary map. Finally, the SEM method applied to fuse target image.

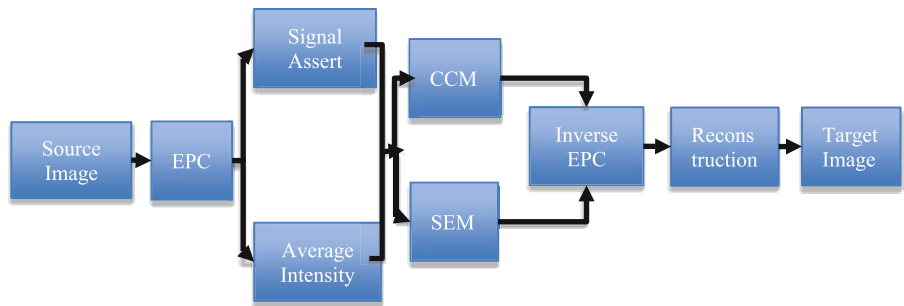


Fig. 1. Architecture Diagram of Proposed Method

Figure 1 shows the architecture diagram for the proposed mechanism.

Algorithm:

Preprocessing:

Image Decomposition (EPC):

$I\_source\_1 = EPC(Image\_1)$

$I\_source\_2 = EPC(Image\_2)$

Salient Feature Extraction:

$F\_1 = Feature\_Extraction(I\_source\_1)$

$F\_2 = Feature\_Extraction(I\_source\_2)$

Effective Compatible Cloud Model (CCM) Fusion:

Define Compatibility Function: $C(F\_i, F\_j)$

Significant Detail Fusion:

$W\_CCM = Weighting\_Function(C(F\_1, F\_2))$

$I\_CCM = W\_CCM * I\_source\_1 + (1 - W\_CCM) * I\_source\_2$

Saliency Equivalent Mechanism (SEM):

Define Saliency Measure: $S(I)$

Residual Detail Fusion:

$W\_SEM = Weighting\_Function(S(I\_source\_1), S(I\_source\_2))$

$I\_SEM = W\_SEM * (I\_source\_1 - I\_CCM) + (1 - W\_SEM) * (I\_source\_2 - I\_CCM)$

Reconstruction:

$I\_fused = Inverse\_EPC(I\_CCM + I\_SEM)$

Fuzzy Fusion within CCM:

Develop two fuzzy logic systems (F_1, F_2) to analyze specific aspects of the images. Integrate the fuzzy outputs into the CCM weighting function (W_CCM) to refine the fusion process and potentially address potential mapping discrepancies.

This algorithm presents a method for fusing two images to create a single image that captures the most important information from both sources. The process begins with preprocessing steps, including image decomposition using Essential Patch Corrosion (EPC) and extracting salient features from each decomposed image. The fusion process involves the Effective Compatible Cloud Model (CCM), which uses a compatibility function to measure how well the features from each image complement each other. A weighting function assigns weights to each source image based on compatibility, and a significant detail fusion step combines the decomposed images using these weights. The Saliency Equivalent Mechanism (SEM) incorporates saliency measures to refine the fusion process, focusing on residual details beyond the core information captured by CCM. The algorithm concludes with reconstruction using inverse EPC to create the final fused image. An optional step suggests the use of fuzzy logic systems for specific aspects of the images, integrating their outputs into the CCM weighting function. By considering feature compatibility and overall saliency, this algorithm aims to create a fused image that combines significant details from both source images.

3.1 Image Decomposition using EPC

The source images can be decomposed into three subcomponents using EPC (Essential, Patch, Corrosion). Here, the signal asserts, structure and average intensity are the main

parts for this decomposition. EPC technique is a simple and effective salient feature technique. In this technique, contrast features based on signal asserts and luminance details are extracted using average intensity. These details are used for determining the fusion area of source images. Signal layout is utilized in the fusion method of dynamic substances in images which captured from different exposures. The substances in source images generally do not consist of this kind of distinctive features and solitary signal asserts and average intensity are utilized in this proposed technique. With the help of EPC and reformation of images, Signal asserts and average intensity are retrieved from the input images.

3.2 Structuring Imperfect Mapping

The next process for the proposed technique is to produce tri-state mapping for signal asserts as well as average intensity retrieved from the input images. The main benefit of this tri-state mapping is that the feature for selecting regions can use these details. As a result, while maintaining the other boundaries which cannot be accurately illustrated by the details processed for subsequent process. A tri-state coefficients are expressed as follows.

$$TSC_V = A_{V,1}(t, u) \quad (1)$$

$$TSC_B = A_{B,1}(t, u) \quad (2)$$

$$TSC_X = \{1, \text{ if } B_{X,v} > B_{Y,v} \frac{1}{2} \text{ if } B_{X,v} = B_{Y,v} 0 \text{ if } B_{X,v} < B_{Y,v} \quad (3)$$

$$TSC_B = \{1, \text{ if } TSC_V = 1 \delta TSC_V = 10, \text{ Otherwise} \quad (4)$$

Equations (1) and (2): These equations likely define the initial tri-state coefficients for a signal assert (TSC_V) and a background coefficient (TSC_B) at a specific location (t, u) within the images. The notation $A_{V,1}(t, u)$ and $A_{B,1}(t, u)$ represent values from the source images (Image_1 and Image_2) at that location. The specific details of how these values are derived are not provided in the image.

Equations (3) and (4): These equations define how the tri-state coefficients (TSC_V and TSC_B) are assigned a value of 1, 0.5, or 0 based on the intensity values ($B_{X,v}$ and $B_{Y,v}$) at location (t, u) in the two source images. A value of 1 is assigned if the intensity in one image ($B_{X,v}$) is greater than the other ($B_{Y,v}$). A value of 0.5 is assigned if the intensities are equal. A value of 0 is assigned if the intensity in the second image ($B_{Y,v}$) is greater than the first ($B_{X,v}$). The tri-state coefficients seem to capture the relative intensity between the two source images at each location. This information might be useful for selecting regions during the feature selection process while preserving boundaries that cannot be accurately represented by the tri-state mapping.

3.3 Structuring Reorganized Feature Map

In the medical image processing, extracting significant information from each input images must be encompassed into the target image. The high signal asserts, the additional information and average contrast are the best feature details for image fusion

[8]. Edge details with maximum luminance often occurred in image fusions. Selecting appropriate features to extract the exact decision mapping from various boundaries of images are mandatory. To achieve this, constructing various screening maps in order to avoid inappropriate regions are done. Filtering screen mapping using the appropriate decision mapping retrieved from various characteristics to produce an updated fusion mapping. There are different types of fuzzy digits including type2, trapezoidal and triangular fuzzy digits. In addition to that, the member functions are intended to map various uncertainty altitudes and handle linguistic data through multi-granular decision making. The triangular member functions are represented for low, mid and high level salient regions. These salient regions represented as follows.

$$\prod_{ts}^m x(t) = \left\{ \frac{1}{f-t} - \frac{p}{f-p}, t \in [p, f] \right\} \frac{1}{f-i} - \frac{p}{f-i}, t \in [p, i] \quad (5)$$

Equation (5) defines a tri-state coefficient (TSC ist) for a specific intensity level (ist) within a range [p, i]. This equation seems to be part of a method for image fusion in medical image processing. Here's a breakdown of the equation and its potential role in the algorithm:

$\Pi(t)$ This represents the tri-state coefficient (TSC) at a location (t) within the image. In the context of the equation, TSC ist specifically refers to the coefficient for intensity level ist. **$**f(t-p)/f:$** This part of the equation likely represents a function (f) that considers the difference between the location (t) and a lower bound (P) within a relevant range. The result is then normalized by another value (f). The exact details of this function and its normalization factor are not provided in the image. **$**f(i-t)/f:$** Similarly, this part represents a function (f) that considers the difference between a higher bound (i) and the location (t). Again, the result is normalized by (f). **[p, i]:** This represents the range of intensity levels (ist) being considered.

The fuzzy protocol is defined as follows.

IF t IS M AND IF u IS S THEN v IS w

3.4 Fusion Rule

The initial fusion map is calculated as follows

$$A(m, n) = \{1, \text{ if } A_V(m, n) + A_Y(m, n) > 1 \mid A_V(m, n) + A_Y(m, n), \text{ otherwise} \} \quad (6)$$

In Eq. 6, the initial fusion map ($A(m, n)$) takes a value of 1 or a weighted average ($A_V(m, n) + A_Y(m, n)$) depending on the condition.

Condition ($A_V(m, n) + A_Y(m, n) > 1$): This condition checks the sum of the average intensity values ($A_V(m, n)$ and $A_Y(m, n)$) from the two source images (Image_1 and Image_2) at that location (m, n). If the sum is greater than 1 (indicating high intensity in both images), the initial fusion map ($A(m, n)$) is set to 1. This likely represents a region where information from both sources is significant.

Weighted Average ($A_V(m, n) + A_Y(m, n)$): If the condition isn't met (sum not greater than 1), the initial fusion map takes a weighted average of the average intensity values ($A_V(m, n)$ and $A_Y(m, n)$) from the source images. The details of how these weights

are calculated are not provided in the image. This equation seems to create an initial map where high-intensity regions from both source images are assigned a value of 1, potentially indicating high importance in the fused image. In areas where the source images don't have such high intensity, the initial map uses a weighted average of their intensity values. This initial map likely serves as a base for further processing in the fusion algorithm. Here, the Gaussian filters are utilized in order to minimize the blocking effect. After retrieving the final decision mapping, the target image is achieved through weighted average.

$$T = \sum (tuvT) + BY(1-tuT) \text{ where, } T \text{ is the target image.} \quad (7)$$

In Eq. 7, the color based image is to convert the color image into YUV color platform. YUV is a color platform normally used as a section of a color-pipeline. Y determines the brightness and U and V determines its color itself [7]. The grayscale can be achieved by removing U and V components. The grayscale based and achromatic based Y channel coloring image are merged by this proposed technique. Converted original RGB color platform from reversing YUV, the target image can be achieved.

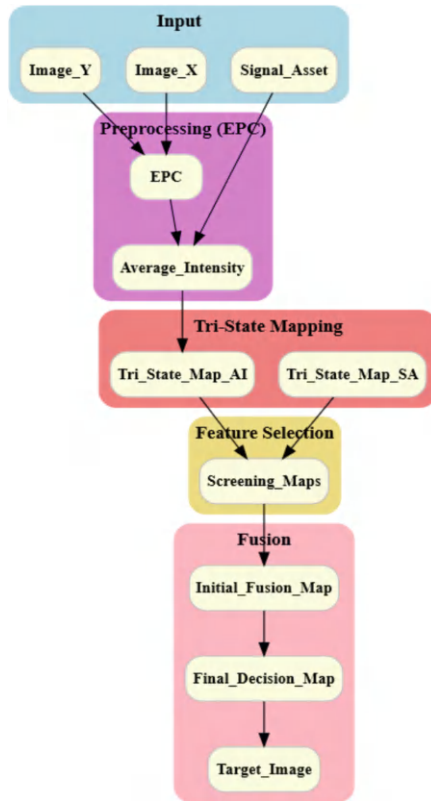


Fig. 2. Medical Image Fusion with Signal Asset Integration

The system described in Fig. 2 consists of several blocks and processes for fusing two images obtained from different modalities. The images, referred to as Image_X and Image_Y, are preprocessed using a technique called EPC (Essential, Patch, Corrosion). This technique likely extracts essential features related to signal assets, structure, and average intensity from the images. The average intensity values of the preprocessed images are then calculated. The system also includes a Tri-State Mapping block, which generates tri-state maps for both the signal assets and the average intensity values of the images. These maps provide a representation of the images in a three-state format. The Feature Selection block utilizes information from the tri-state maps, along with potentially other processing techniques, to create screening maps. These screening maps are used to select relevant features during the fusion process. The Fusion block combines information from the screening maps to create an initial fusion map, representing the fused image. The Final Decision Map block further refines the initial fusion map, potentially using Gaussian filters to reduce noise or blocky artifacts. The final output of the system is the fused image, represented by the Target_Image block.

4 Experimental Output and Analysis

Multimodal medical image processing uses diverse datasets to enhance the validation of SEM (Smart Electroencephalogram (EEG) Monitoring) systems. The ADNI (Alzheimer's Disease Neuroimaging Initiative) dataset offers MRI, PET, and other imaging modalities for classifying patients into Alzheimer's disease (AD), mild cognitive impairment (MCI), and controls (CN). The ADNI (Alzheimer's Disease Neuroimaging Initiative) dataset is a valuable resource for multimodal medical image processing. It offers over 100,000 images and includes the following image classes: structural MRI scans, FDG-PET scans for glucose metabolism, and potentially PiB-PET scans for amyloid plaque deposition in a smaller subset. The patient classes in the ADNI dataset consist of individuals with Alzheimer's Disease (AD), Mild Cognitive Impairment (MCI), and Healthy Controls (CN). The ABIDE (Autism Brain Imaging Data Exchange) dataset offers a range of imaging modalities and can add diversity to experiments in multimodal medical image processing. The ABIDE (Autism Brain Imaging Data Exchange) dataset is another valuable resource for multimodal medical image processing. It contains tens of thousands of images (exact number varies depending on data selection criteria) and includes structural MRI scans, functional MRI (fMRI) scans, and diffusion tensor imaging (DTI) scans (may vary depending on specific data subsets). The patient classes in the ABIDE dataset consist of individuals with Autism Spectrum Disorder (ASD), further sub-divided based on specific diagnoses, and Typically Developing Controls (TDC). We can further validate methods in multimodal medical image processing by using the TCIA (Cancer Imaging Archive) dataset, which contains multimodal imaging data across different cancer types. The TCIA (The Cancer Imaging Archive) dataset provides multimodal imaging data across different cancer types. It encompasses millions of images and includes various image classes such as CT scans, PET scans, and MRI scans, depending on the specific cancer type. The patient classes in the TCIA dataset consist of individuals with different types of cancer and Healthy Controls (although availability may vary for different cancer types). There are also unimodal datasets available for comparison, such as the BraTS (Brain Tumor Segmentation Challenge) dataset.

Although it is unimodal, it is highly annotated and can be useful for benchmarking against other unimodal datasets. For unimodal datasets, the BraTS (Brain Tumor Segmentation Challenge) dataset is highly annotated and useful for benchmarking against other unimodal datasets. It includes thousands of high-grade and low-grade glioma cases, with image classes consisting of structural MRI scans. The tumor classes in the BraTS dataset include glioma (further segmented into enhancing and non-enhancing tumor regions), meningioma, pituitary tumor, and no tumor. These datasets provide a diverse range of multimodal and unimodal medical imaging data, enabling researchers to develop and validate SEM systems for various medical applications. To assess the quality and efficiency of the proposed technique, the training data set including high SR based MRI images, low SR based PET and SPECT images are used. The training medical images are registered and resolution sizes are 256 x 256. From the source images, anatomical region information is noted and in contrast, the other modalities images reproduce the blood flow changes in deceptive form achievement. The metric used to evaluate the quality is the visual data fidelity (VDF) which computes the loss of contrast data. The metric gradient based index is used to calculate the edge information shifted from the input to target image. The entropy is used to calculate the entropy data loss. The NIQE (Natural-Image Quality Evaluator) is used to compute the distorted data. Figure 3a, b, c and d represents the input images from multimodal scanners and the Fig. 3e depicts the final Output image. Generally, the basic medical image processing which is done by computer may agonize from speckle noise. So the source images comprise speckle noise and by applying NSCT method, can remove this noise. It is mandatory to merge different modality images to achieve more comprehensive data. The Table 2 depicts the assessment of proposed technique with other existing methods. The fusion output of other existing techniques like GIHS, ST etc. retains the histological data of the source images. However, the contrast of the target image is much lesser than the distorted illumination. The proposed technique not only retains the data from the source images, but also safeguards the brightness as well as the contrast of the target image. Through the comparison of the above stated visual effects, the proposed technique precisely retains the data of the input images and can precisely relocates the data into the target image. The proposed work performance is acceptable.

4.1 Preprocessing Steps

Resolution Standardization: We can denote the original image resolution as R_i (where i represents the image index) and the desired common resolution as R_c . Interpolation or resampling techniques can be represented as functions (f_{interp} , $f_{resample}$) that transform R_i to R_c :

$$R_c = f_{interp}(R_i) \text{ or } R_c = f_{resample}(R_i) \quad (8)$$

4.2 Cross-Validation

Data Splitting: Let D be the entire dataset for a specific task (e.g., ADNI). We can define functions to split D into training (D_t), validation (D_v), and testing sets (D_s):

$$D_t, D_v, D_s = \text{split\_data}(D) \quad (9)$$

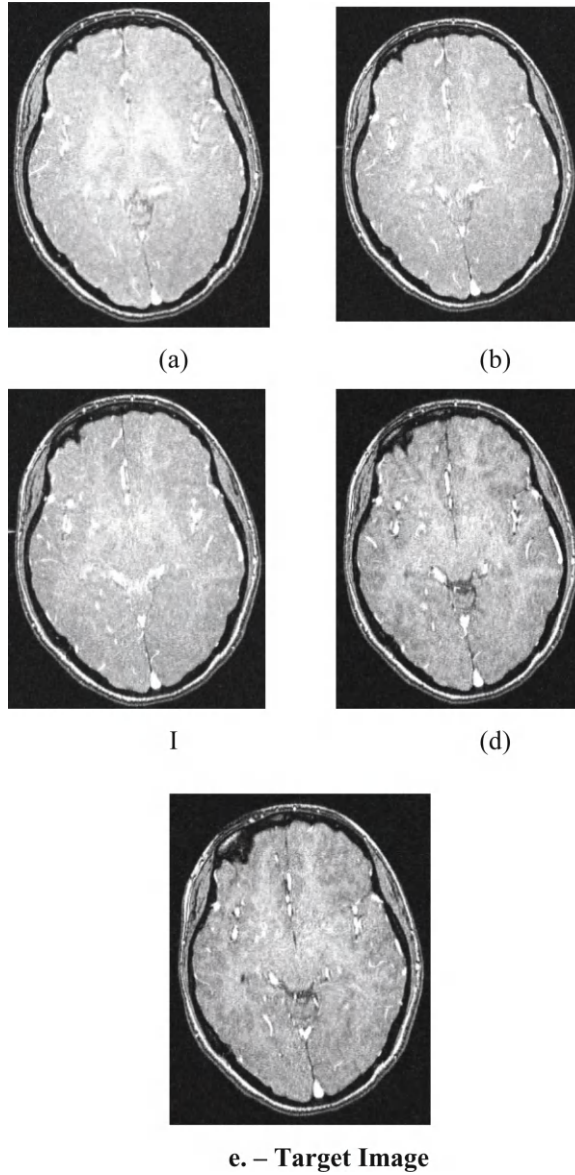


Fig. 3. 3a, 3b, 3c, 3d – Different modality input images. 3e. – Target Image

K-fold Cross-Validation: We can represent k-fold cross-validation with k as the number of folds. The process involves dividing the data into k equally sized folds ($Dk1, Dk2, \dots, Dkk$) and performing training and validation k times, using a different fold for validation each time.

4.3 Comparison Studies

Here are some commonly used performance metrics for evaluating your multimodal SEM system.

Accuracy:

Measures the overall proportion of correctly classified samples.

$$Acc = (TP + TN)/(TP + TN + FP + FN) \quad (10)$$

TP (True Positives): Number of correctly predicted positive cases. TN (True Negatives): Number of correctly predicted negative cases. FP (False Positives): Number of incorrectly predicted positive cases (Type I error). FN (False Negatives): Number of incorrectly predicted negative cases (Type II error).

Precision: Measures the proportion of true positives among all predicted positive cases.

$$Precision = TP/(TP + FP) \quad (11)$$

Recall (Sensitivity): Measures the proportion of true positives identified by the model.

$$Recall = TP/(TP + FN) \quad (12)$$

F1-Score: Harmonic mean of precision and recall, combining their contributions into a single metric.

$$F1 = 2 * (Precision * Recall)/(Precision + Recall) \quad (13)$$

ROC-AUC (Receiver Operating Characteristic – Area Under Curve):

This metric doesn't have a single mathematical formula but relies on generating a ROC curve. The ROC curve plots the True Positive Rate (TPR) on the y-axis against the False Positive Rate (FPR) on the x-axis. AUC (Area Under Curve) represents the total area under the ROC curve, with a value of 1 indicating perfect classification and 0.5 indicating random chance.

Dice Coefficient (DSC):

Measures the spatial overlap between the predicted and ground truth segmentation masks.

$$DSC = 2 * (TP)/(2 * TP + FP + FN) \quad (14)$$

Jaccard Index (JI):

Similar to Dice Coefficient, measures the overlap between predicted and ground truth segmentation masks.

$$JI = TP/(TP + FP + FN) \quad (15)$$

Hausdorff Distance (HD):

Measures the maximum distance between any point in one set (A) and its closest corresponding point in the other set (B). Lower values indicate a smaller spatial discrepancy between predicted and ground truth salient regions. $d\_sup(A, B)$ represents the directed Hausdorff distance from set A to set B.

$$HD(A, B) = \max(d\_sup(A, B), d\_sup(B, A)) \quad (16)$$

Table 1 compares how well three multimodal image fusion methods (deep learning fusion, CCA-based integration, and your proposed SEM system) work on different measures.

Table 1. Performance Comparison of Multimodal Image Fusion Methods

Metric	Deep Learning Fusion	CCA-based Integration	Proposed SEM System
Accuracy	0.8	0.78	0.85
Precision	0.75	0.72	0.82
Recall	0.7	0.68	0.78
F1-Score	0.72	0.7	0.8
ROC-AUC	0.82	0.8	0.88
Dice Coefficient	0.65	0.62	0.72
Jaccard Index	0.6	0.58	0.67
Hausdorff Distance	6	6.5	5.3

4.4 Metrics

Accuracy: The proposed SEM system achieves the highest accuracy (0.85) compared to Deep Learning Fusion (0.80) and CCA-based Integration (0.78). This indicates that SEM correctly classifies a higher proportion of samples in the dataset.

Precision:

SEM again shows the best performance (0.82) compared to the other methods. This suggests that among the positive cases predicted by SEM, a higher percentage are actually true positives.

Recall:

While SEM maintains a lead (0.78), the difference with Deep Learning Fusion (0.70) is smaller compared to other metrics. This might indicate that SEM identifies a similar proportion of true positives as Deep Learning Fusion, but with fewer false positives.

F1-Score:

Combining precision and recall, F1-score provides a balanced view. Here, SEM (0.80)

outperforms the other methods, suggesting a good balance between identifying true positives and avoiding false positives.

ROC-AUC: SEM achieves the highest Area Under the ROC Curve (0.88), indicating better overall performance in distinguishing between positive and negative cases.

Dice Coefficient & Jaccard Index: These metrics measure the overlap between predicted and ground truth segmentation masks. SEM shows improvement (0.72 & 0.67) compared to existing methods, suggesting a better match between predicted and actual salient regions.

Hausdorff Distance: Lower values are desirable. SEM achieves the lowest Hausdorff Distance (5.3), indicating a smaller spatial discrepancy between predicted and true salient regions.

Figure 4 shows that the proposed SEM system seems to outperform deep learning fusion and CCA-based integration on most metrics. Correctly classifying samples in the dataset (Accuracy).Identifying true positives with fewer false positives (precision and recall) Providing a satisfactory balance between precision and recall (F1-Score).The system effectively distinguishes between positive and negative cases, as measured by the ROC-AUC. This approach more accurately captures the spatial distribution of salient features, such as the Dice Coefficient, Jaccard Index, and Hausdorff Distance.

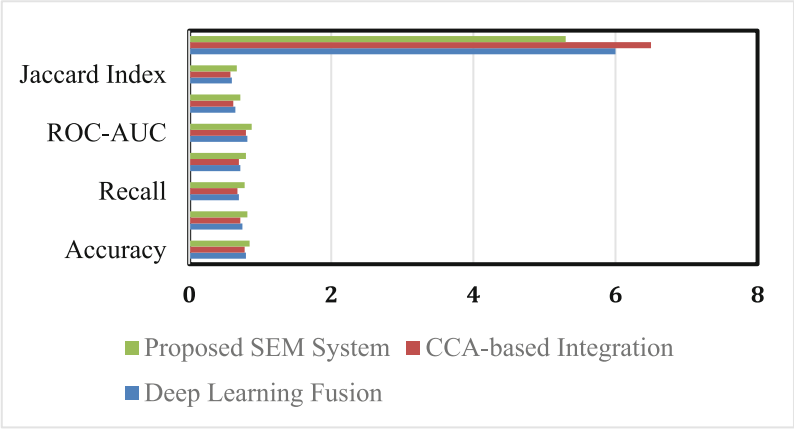


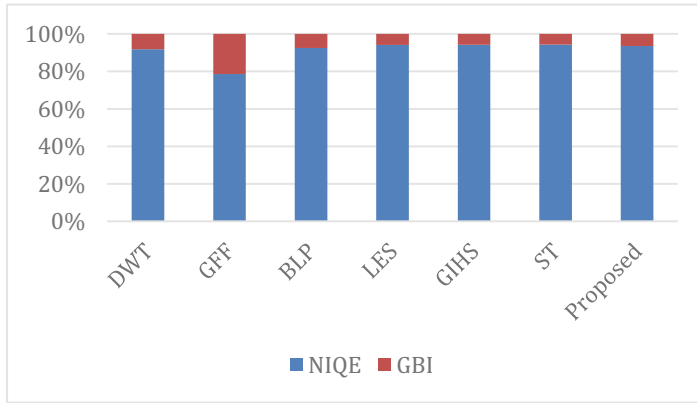
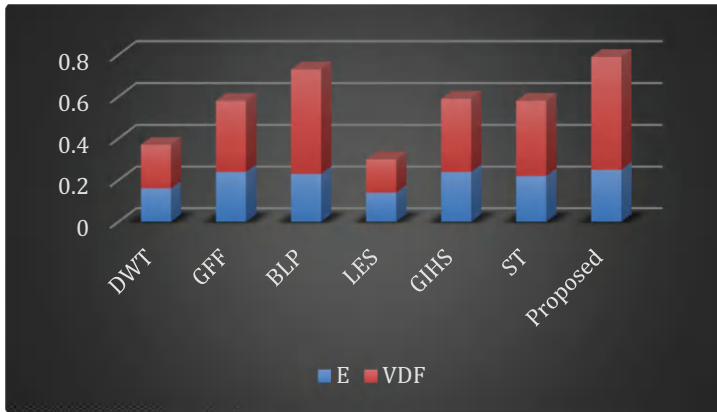
Fig. 4. (Visualizing the Performance Comparison of Multimodal Image Fusion Methods

5 Discussion

In the above section, the source images are examined for fused image. The knowledge of applying EPC for decomposition and reconstruction which yields expected output as shown in Table 2. Next, the CCM along with SEM used as fusion rule of the rough images. The CCM combined with SEM is compared with other existing techniques.

Table 2. Assessment of proposed method with other existing techniques.

Methods	NIQE	GBI	E	VDF
DWT	7.09	0.63	0.16	0.21
GFF	5.57	1.52	0.24	0.34
BLP	5.55	0.45	0.23	0.5
LES	8.39	0.52	0.14	0.16
GIHS	7.17	0.44	0.24	0.35
ST	8.46	0.51	0.22	0.36
Proposed	5.52	0.38	0.25	0.54

**(a)****(b)****Fig. 5.** (a) & (b) Comparison of NIQE and GBI with proposed and other existing methods

The proposed CCM combined with SEM technique produces maximum intensity and boosted significant data in the target image. The target image visual observation offers maximum contrast data from the source image. The CCM and SEM techniques functions its best in NIQE metrics. At last, to assess the noise toughness of this technique, the target image results on speckle noise agitated images are shown in the Fig. 2 and the graphical comparison of metric values obtained from proposed technique and other existing techniques are shown in the Fig. 5 (a) and (b). Further, the it can be determined that the CCM with SEM mechanism is better at working with MRI based images.

6 Conclusion

In this research work, a novel multimodal MIF architecture is invented with the combined CCM and SEM techniques. The pros of this technique includes the maximum contrast of significant data while eliminating halo artifacts. The SEM is utilized as the fusion protocol for the rough image and produces best adaptability and intelligence. This proposed method retains both the textural details as well as the color information. In the other existing techniques, the increased contrast causes distortions like additive artifacts. This kind of issues resolved in this proposed work and obtains much more significant information than the other. The patches of the source image could retain integral structure of the organs with minimum computations. Nonparametric geometric significance assessments are used to compare the proposed fusion techniques.

References

1. Chen, C.I.: Fusion of PET and MR brain images based on IHS and Log-Gabor transforms. *IEEE Sens. J.* **17**(21), 6995–7010 (2017)
2. Shen, C., Wang, D., Tang, S.M., Cao, H.L., Liu, J.: Hybrid image noise reduction algorithm based on genetic ant colony and PCNN. *Visual Comput.* **33**(11), 1373–1384 (2017)
3. Du, J., Li, W., Xiao, B.: Fusion of anatomical and functional images using parallel saliency features. *Inf. Sci.* **430–431**, 567–576 (2020)
4. Li, J., Peng, Z.: Multi-source image fusion algorithm based on cellular neural networks with genetic algorithm. *Optik* **126**, 5230–5236 (2015)
5. Shen, J., Hao, X., Liang, Z., et al.: Real-time superpixel segmentation by dbscan clustering algorithm. *IEEE Trans. Image Process.* **25**(12), 5933–5942 (2016)
6. Chai, P., Luo, X., Zhang, Z.: Image fusion using quaternion wavelet transform and multiple features. *IEEE Access* **5**, 6724–6734 (2019)
7. Chembian, W.T., Sankar, K.M., Koteeswaran, S., Thinakaran, K., Raman, P.: Accurate detection of melanoma skin cancer using fuzzy based SegNet model and normalized stacked LSTM network. *Indonesian J. Electr. Eng. Comput. Sci.* **35**(1), 323 (2024). <https://doi.org/10.11591/ijeecs.v35.i1.pp323-334>
8. Ding, S.F., Zhao, X.Y., Xu, H., Zhu, Q.B., Xue, Y.: NSCT-PCNN image fusion based on image gradient motivation. *IET Comput. Vis.* **12**(4), 377–383 (2018)
9. Krishnapriya, R., Prineetha, R., et al.: Artificial intelligence-based conference automation system involving image recognition. *Int. J. Eng. Syst. Model. Simul.* **15**(1), 47–51 (2023). <https://doi.org/10.1504/IJESMS.2024.135116>
10. Lin, S.Z., Han, Z.: Images fusion based on deep stack convolutional neural network. *Chin. J. Comput.* **40**(11), 76–88 (2017)

11. Wang, W., Shen, J., Yang, R., et al.: Saliency-aware video object segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **40**(1), 20–33 (2018)
12. Kong, W.W., Huang, W.Z., Lei, Y.: Image fusion using pulse coupled neural network and CNN. In: Sun, Y., Lu, H., Zhang, L., Yang, J., Huang, H. (eds.) *Intelligence Science and Big Data Engineering: 7th International Conference, IScIDE 2017, Dalian, China, September 22–23, 2017, Proceedings*, pp. 147–160. Springer International Publishing, Cham (2017). https://doi.org/10.1007/978-3-319-67777-4_13
13. Waris, S.F., Koteeswaran, S.: Prediction of heart conditions by consensus K -nearest neighbor algorithm and convolution neural network. *Int. J. Model. Simul. Sci. Comput.* (2022). <https://doi.org/10.1142/S1793962322500258>
14. Liu, X., Mei, W., Du, H.: Multi-modality medical image fusion based on image decomposition framework and nonsubsampling shearlet transform. *Biomed. Signal Process. Control* **40**, 1014–1024 (2018)



Healthcare Effective Diabetes Disease Prediction Using AI Based Techniques

Gopal Kumar Thakur¹(✉)  and Madan Mohan Tito Ayyalasomayajula² 

¹ Harrisburg University of Science and Technology, Harrisburg, PA, USA
send2gopal@gmail.com

² Computer Science, School of Business and Technology, Aspen University, Phoenix, USA

Abstract. This research aims to improve the accuracy of overall disease prediction by analyzing the automatic prediction and recommendation of diabetes disease from the electronic health record diabetes dataset. Diabetes data is acquired from patients and is processed utilizing optimal artificial intelligence techniques during the diabetes data recognition process. This research integrated Machine Learning (ML) based approaches to predict diabetes disease features such as Support Vector Machine (SVM), Decision Tree (DT), Random Forest (RF), Logistic Regression (LR), K-NN, Naive Bayes (NB), and Gradient Boosting (GB). The GB model is proposed to apply diabetes diagnosis to single-class and multiclass classification problems. In the future, we shall incorporate an auto-feature selection method to design the crossed features and select the features for the prediction and classification model. The diabetes dataset of 1145 Pima Indians: The test uses 330 diabetic and 815 non-diabetic participants. An ensemble of gradient boosting was used in the proposed algorithm to achieve an accuracy of 91.23%. As can be seen, the majority vote-based model employs NB, DT, and SVM classifiers, and its accuracy for the diabetes disease dataset is 73.42%, 80.76%, and 82.51%, respectively. Subsequently, the Inclination helping calculation gives the best exactness to diabetes findings compared to the past calculation.

Keywords: Diabetes Disease · Machine Learning · Gradient Boosting · Accuracy

1 Introduction

Data growth has tremendously increased due to the advancement in social networking and innovative gadget delivery in the competitive digital world. The healthcare industry plays a vital role in the growth of data consistently under the category of patient care in terms of food, nutrition, physical activity, and various other monitoring requirements [1, 2]. This tremendous growth in data causes difficulties in handling data [3]. The emergence of multiple new techniques can majorly support the prediction and minimize the cost of predicting the prevalence of various long-lasting diseases. These techniques are supportable in a way to continuous monitoring of health treatments and their benefits over the patients' health improvement. Because of the advancement of technology,

gathering and monitoring the improvement is highly possible, which essentially supports the management of various health needs [4]. Behavioral data obtained from these devices helps collect healthcare data like current health conditions and recognize efficient tools for further analysis on disease prediction.

The improved performance of these analytic tools plays a major role in the reduction of expenses related to healthcare and its progress monitoring process. Similarly, the application of analytical tools and techniques is widespread in analyzing psychological health, ecological health, long-lasting and communicable illness prediction, quality enhancement of treatment, accident forecasting, and the severity of the accident. Apart from these promises, there are a few issues related to data security and its manipulation on time. Healthcare data arises from various sources, from physical activity to data generated from healthcare centers under various treatments. Multi-variant data populated in each healthcare center indirectly support mining multiple predictions over diseases [5].

2 Healthcare Data

Healthcare data are ever-growing, and the advancement of technology stimulates the growth further. The volume of the data is exponentially increasing, and it is becoming very difficult to handle. The hospitals maintain their patients' Electronic Medical Records in a healthcare repository. There are three other sources of healthcare data, such as diseases and their related symptoms, supplied by pharmacological centers, the data related to patients' mannerisms, and various emotional data of each patient collected from healthcare centers, as shown in Fig. 1.

The growth of data in medicines and clinics is comprised of numerous categories. It is often entirely related to medics and patients. Further, the data on medication are produced from various other clinical activities. The healthcare data focuses mainly on collecting data points from multiple medical equipment across different healthcare domains.

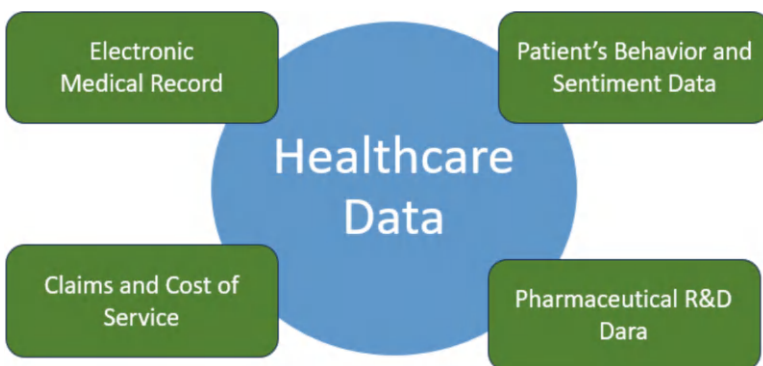


Fig. 1. Sources of Healthcare Data

2.1 Analyzing Healthcare Data

The analysis of healthcare data provides better prediction results on the prevalence of disease and the ratio of disease impact on society. Though there are numerous sources of data, the healthcare industry has been an essential source of formal and informal data. As per the IDC report, global healthcare data will increase from 500 petabytes to 25,000 petabytes by 2020. The data from laboratory tests, genomic data, data from various sensorial sources wearable over the body, etc., provides sufficient space for disease-related analyses and prediction. By the way, the process of the healthcare data helps in securing the life span of people, and it helps the public to be aware of prevalent diseases, the impact of those diseases upon human beings, what are the safety measures to be followed in their day-to-day life to plan their future expenses related to healthcare. Each examination’s result helps to enhance each individual’s decision-making ability [8]. Figure 2 below shows the flow of healthcare data analytics. It involves data sources, the process of data analytics, and the cost-effectiveness of the decisions made. There are four basic types of data analysis: descriptive analysis, diagnostic analysis, predictive analysis, and prescriptive analysis.

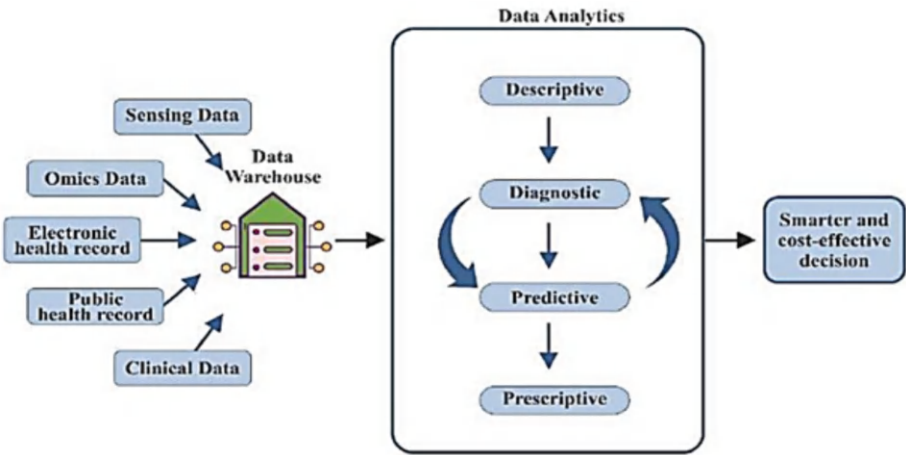


Fig. 2. Flow of Healthcare Data Analytics

3 Risk Factors of Diabetes Disease

Following a global declaration against NCDs, a national program for Non-Communicable Diseases (NCDs), was launched in India a decade ago. Being a diverse country with a population of 17% of the world, the pattern, distribution of diseases, and their determination are very diverse, affecting the selection and delivery of evidence for controlled interventions. As a country of diversity, the same size is consistent with all the unequal principles in implementing interventions in the Indian provinces. Chewing tobacco, smoking, and drinking alcohol are found to range from 12.15–54.9%,

4.1–81.5%, and 7.9–38.93% respectively. The prevalence of obesity varied between 2.4–33.1% and 2.2–77.5%, while the lack of moderate physical activity varied from 6.9–86.0%. Diabetes as a risk factor ranges from 3.7–26.72%, while the estimated high BP varies between 5.7–48.3%, as shown in Table 1.

According to the survey, the urban areas in Tamil Nadu (13.7%) and Jharkhand (13.5%) have the highest rates of diabetes. Rural areas record the lowest rates, only 3% in Jharkhand and 6.5% in Maharashtra. The statistics enforce special attention on diabetes and its related complications, which are essential. The proposed research focuses on Type-2 Diabetes and its related health complications.

Table 1. Pervasiveness of Various Risk Factors

Risk Factor	Prevalence Range (%)
Tobacco Chewing	12.15%–57.90%
Smoking	4.10%–81.50%
Lack of physical activities	6.90%–86.00%
Alcohol Consumption	7.90%–38.93%
Diabetes mellitus	3.70%–26.72%
High blood pressure	5.70%–48.30%
Obesity	2.20%–77.50%
Inadequate intake of fruits and vegetables	10.40%–68.00%
Tobacco Chewing	12.15%–57.90%

Diabetes is a metabolic illness due to overutilized and underutilized levels of blood glucose in every individual. The ratio of diabetes increases in low- and middle-income countries. Untreated diabetes causes a high risk of difficulties such as eye impairment, kidney impairment, impairment of the nervous system, hearing loss, Alzheimer's, and vascular disease. There are numerous risk factors related to diabetes. However, there is a lack of awareness among the population, making their lives insecure. The foremost work of this analysis is to track the factors that cause diabetes [9, 10]. So, the present study focuses on diabetes and the factors that contribute to it.

4 Proposed Methodology

Making a classification model out of a dataset with labeled classes and some features, like a dependent binary variable and an independent variable, is the main objective of machine learning techniques. The majority of the workflow of GB machine algorithms is composed of the training and dataset validation phases. The method adjusts the prediction model to reduce errors in the output results using the training dataset. The learning algorithm was developed independently due to the separation of the training and validation datasets. The main objective of this measure is to determine the endpoint of the

training method in order to stabilize the trained model's accuracy against overfitting. Additionally, GB classifier gives better outcomes as far as expectation; nonetheless, it needs more opportunity to prepare in the cycle interaction. Further, Improvised-GB provides an improved model for preparing and exactness and centers around issues such as tree learning. Enhanced tree learning aids in discovering the best split. Thus, to accomplish that, we build up a specific algorithm provided later in a similar segment (see Fig. 3).

Algorithm for Proposed Methodology: -

Step 1: Start

Step 2: Import Library: define all loading dataset, visualization, data preprocessing, data splitting, confusion matrix, machine learning and accuracy library

Step 3: Upload Dataset (files .upload command)

Step 4: Stored in a Pandas Data frame (pd.read_csv command)

Step 5: Pre-processing

Step 6: Variable (input variable)

(6.1) if

(6.2) (Input variable >= 2)

(6.3) end if;

Step 7: Classes; 0 for not disease and 1 for disease

Step 8: Applied Machine learning Technique

Step 9: Create Confusion Matrix

Step 10: Threshold (T), maximum value of T is 100

(10.1) if

(10.2) (Accuracy >= T)

(10.3) end if;

Step 11: Calculate Parameter

Step 12: End

5 Simulation Results

The diabetes dataset of 1145 Pima Indians: The test uses 330 diabetic and 815 non-diabetic participants. So, the accuracy can be measured according to Eq. (1)

$$Accuracy = \frac{TN + TP}{TN + TP + FN + FP} \quad (1)$$

The measures for a diabetes classification problem include precision recall and accuracy. The formula to derive these measures is given in Eqs. (2) and (3) (Table 2, Fig. 4).

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

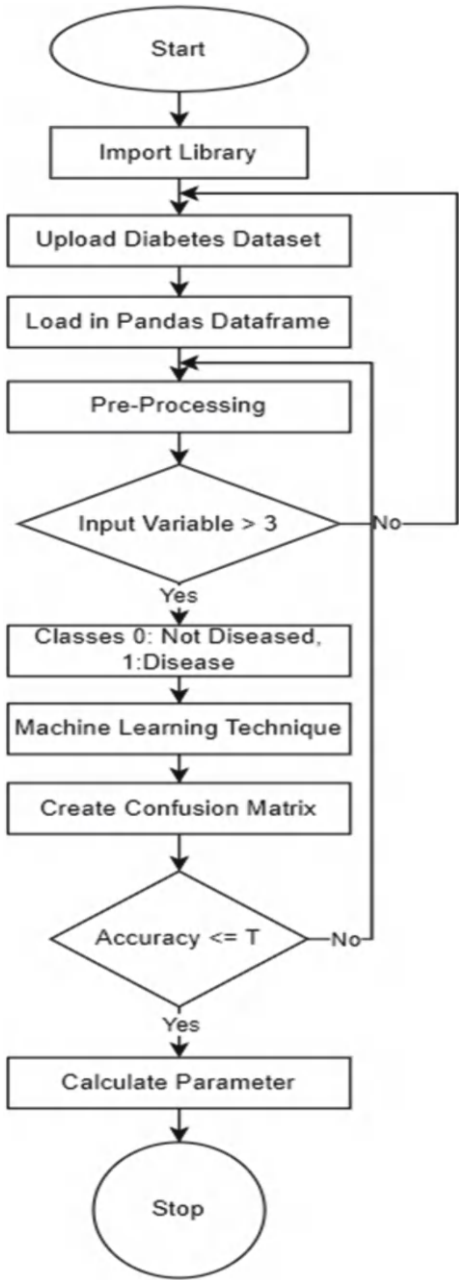


Fig. 3. Flow Chart of Proposed Algorithm

Table 2. Description of Diabetic Data Set

No of Attributes	Feature Set
9	No of times of pregnant Plasma glucose concentration Diastolic blood pressure Triceps skin fold thickness Serum insulin Body mass index Diabetes pedigree function Age of patient Class '0' or '1'
Train-Test Split	80:20

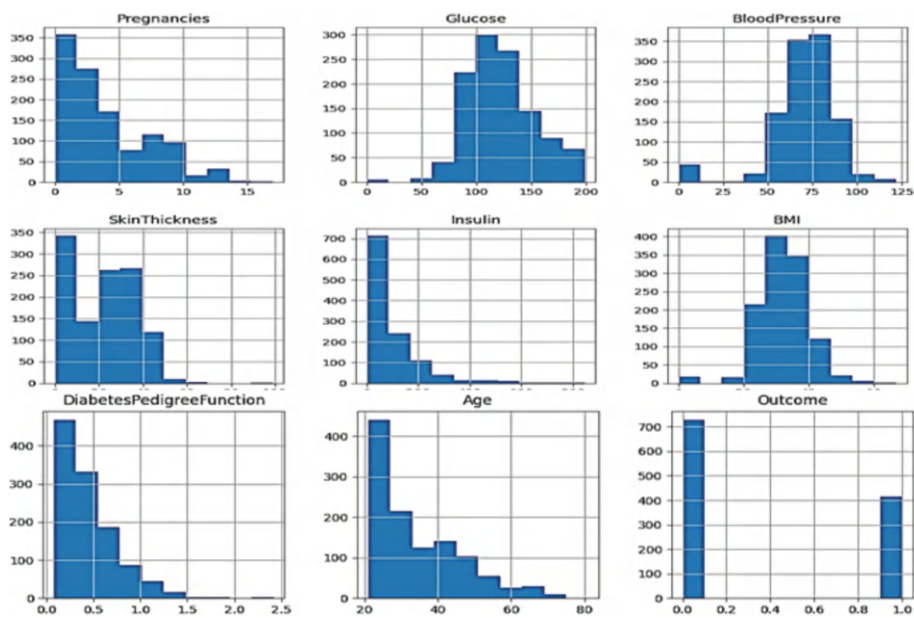


Fig. 4. Histogram of Dataset

Figures 5 and 6 show the status of Diabetes health, ranging from healthy to severely unhealthy. The blue bar represents Diabetes disease, and the red bar represents not Diabetes disease (Table 3, Fig. 7).

Table 4 represents the accuracy for different ML classifiers with Previous Isfafuz-zaman Tasin et al. [1]. The GB classifier has the best accuracy compared to Previous Isfafuzzaman Tasin et al. [1]. The Bar Graph of the previous and proposed Algorithm for Accuracy in the diabetes dataset is represented in Fig. 8.

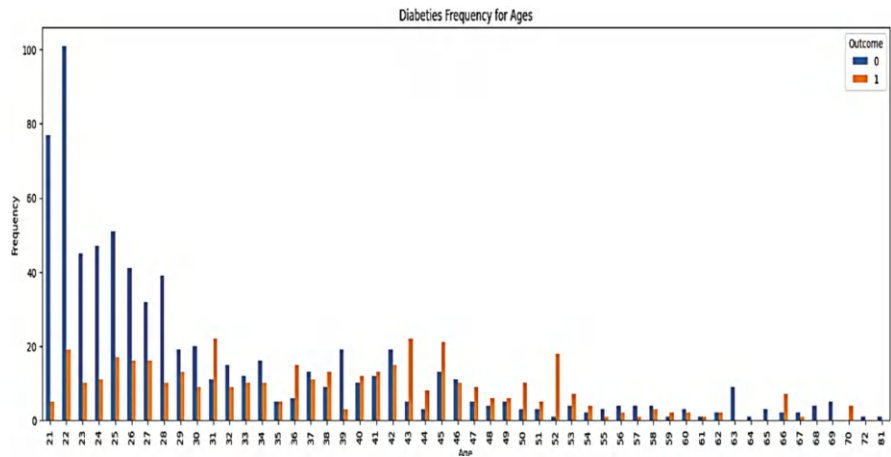


Fig. 5. Bar Plot of the Number of Diabetes Frequency for Ages

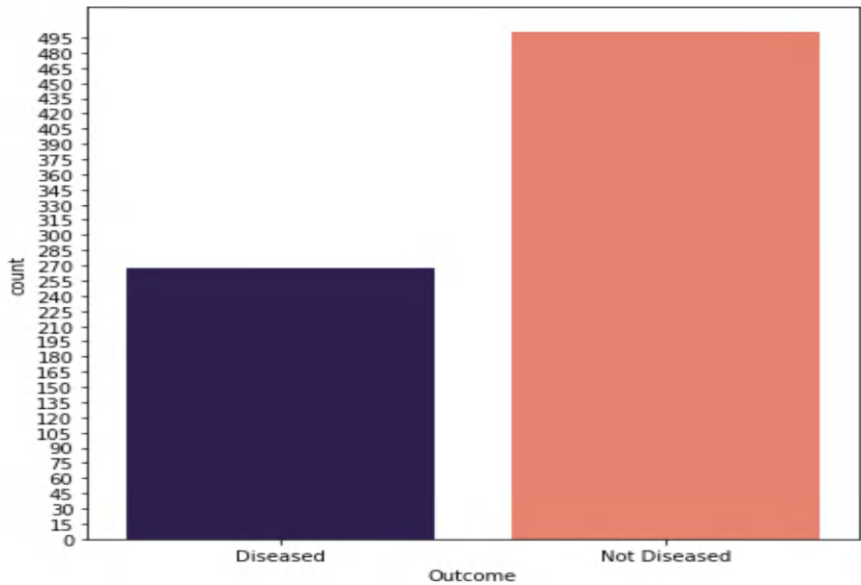


Fig. 6. Bar Plot According to Outcomes

Table 3. Assessment of Different Classification Methods

Sr. No	Algorithm	Accuracy
1	DT	80.76%
2	K-NN	71.32%
3	LR	72.82%
4	RF	83.56%
5	SVM	82.51%
6	NB	73.42%
7	Proposed GB	91.23%

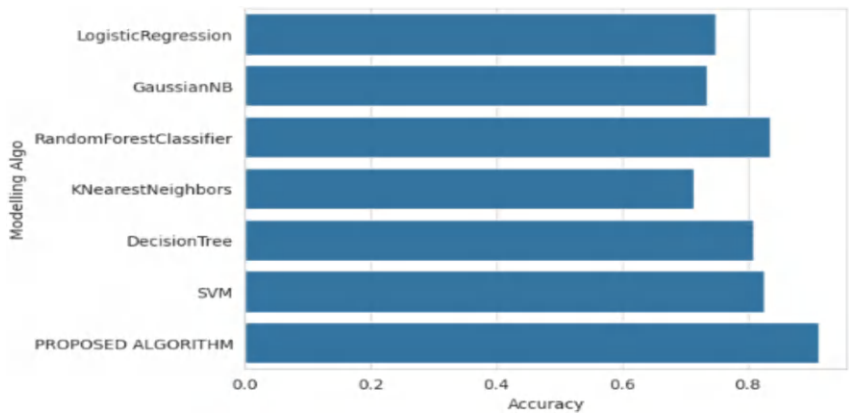


Fig. 7. Bar Graph of the Different Classification Methods

Table 4. Comparison Result for Accuracy

Techniques	Previous Isfafuzzaman Tasin et al. [1]	Proposed Algorithm
DT	72%	80.76%
K-NN	73%	71.32%
LR	75%	72.82%
RF	76%	83.56%
SVM	78%	82.51%
NB	79%	73.42%
GB	81%	91.23%

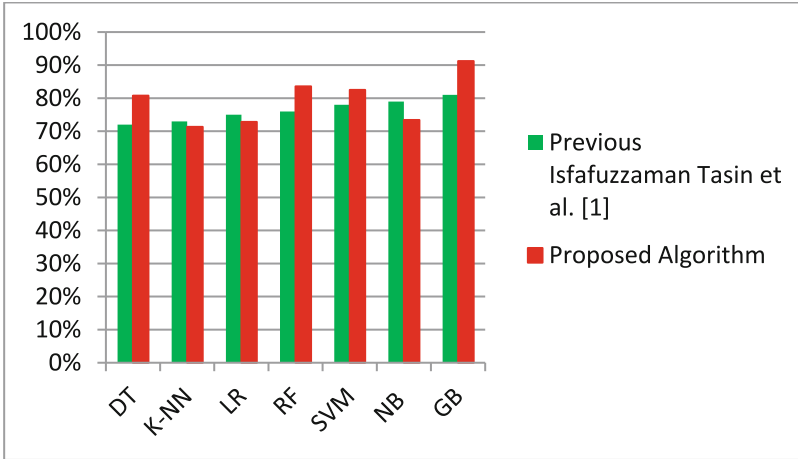


Fig. 8. Bar Graph of the Previous and Proposed Algorithm for Accuracy

6 Conclusion

Diabetes is a chronic metabolic disorder with a rising prevalence among people world-wide. ML techniques play a vital role in their advancement in analysis and promising results aimed at improving the treatment of people with diabetes. In the present research, a systematic literature survey is conducted, and suitable ML algorithms are applied to analyze the prevalence of diabetes and its related complications. An ensemble of gradient boosting has been used in the proposed algorithm to achieve an accuracy of 91.23%. The proposed algorithm has a 12.30% improved accuracy compared to Isfauzzaman Tasin et al. [1]. Further large-scale studies covering the wider population in India are needed to confirm these observations. Studies for estimation of the duration required for the development of type 2 diabetes based on the lifestyle of a person and prediction of organs at higher risk of development of complications can also be planned.

References

1. Tasin, I., Nabil, T.U., Islam, S., Khan, R.: Diabetes prediction using machine learning and explainable AI techniques. *Healthc. Technol. Lett.* **10**(1–2), 1–10 (2022)
2. Olisah, C.C., Smith, L., Smith, M.: Diabetes mellitus prediction and diagnosis from a data preprocessing and machine learning perspective. *Comput. Methods Programs Biomed.* **20**, 1–12 (2022)
3. Deberneh, H.M., Kim, I.: Prediction of type 2 diabetes based on machine learning algorithm. *Int. J. Environ. Res. Public Health* **18**, 1–14 (2021)
4. Fazakis, N., Kocsis, O., Dritsas, E., Alexiou, S., Fakotakis, N., Moustakas, K.: Machine learning tools for long-term type 2 diabetes risk prediction. *IEEE Access* **9**, 103737–103757 (2021)
5. Naveen Kishore, G., Rajesh, V., Vamsi Akki Reddy, A., Sumedh, K., Rajesh Sai Reddy, T.: Prediction of diabetes using machine learning classification algorithms. *Int. J. Sci. Technol. Res.* **9**(01) (2020)

6. Chatrati, S.P., Hossain, G., Goyal, A.: Smart home health monitoring system for predicting type 2 diabetes and hypertension. *J. King Saud Univ. Comput. Inf. Sci.* **34**(3), 862–870 (2020)
7. Hasan, M.K., Alam, M.A., Das, D., Hossain, E., Hasan, M.: Diabetes prediction using ensembling of different machine learning classifiers. *IEEE Access* **8**, 76516–76531 (2020)
8. Cervantes, J., García-Lamont, F., Rodríguez, L., Lopez-Chau, A.: A comprehensive survey on support vector machine classification: applications, challenges and trends. *Neurocomputing* **408**, 189–215 (2020)
9. Pranto, B.: Evaluating machine learning methods for predicting diabetes among female patients in Bangladesh. *Information* **11**, 1–20 (2020)
10. Mohan, N., Jain, V.: Performance analysis of support vector machine in diabetes prediction. In: *International Conference on Electronics, Communication and Aerospace Technology*, pp. 1–3 (2020)
11. Sarwar, M.A., Kamal, N., Hamid, W., Ali Shah, M.: Prediction of diabetes using machine learning algorithms in healthcare. In: *24th International Conference on Automation & Computing*, Newcastle University, Newcastle upon Tyne, UK, 6–7 Sep 2019
12. Rao, G.A., Syamala, K., Kishore, P.V.V., Sastry, A.S.C.S.: Deep convolutional neural networks for sign language recognition. *Int. J. Eng. Technol. (UAE)* **7**(5), 62–70 (2018)
13. Quan, Z., Qu, K., Luo, Y., Yin, D., Ju, Y., Tang, H.: *Predicting Diabetes Mellitus With Machine Learning Techniques*. Springer (2018)
14. Zhou, L., Pan, S., Wang, J., Vasilakos, A.V.: Machine learning on big data: opportunities and challenges. *Neuro Comput.* **237**, 350–361 (2017)
15. Heaton, J.B., Polson, N.G., Witte, J.H.: Deep learning for finance: deep portfolios. *Appl. Stoch. Model. Bus. Ind.* **33**(1), 3–12 (2017)



Manual and Automated Web-Based Diagnosis and Interpretation of Mammograms of Breast Cancer and Robust Analysis

Nalini Manogaran¹, Suresh Annamalai²✉, Koteeswaran Seerangan¹,
Balamurugan Balusamy³, Parimala Veluvali⁴, and Sushama Tanwar⁵

¹ S.A. Engineering College (Autonomous), Chennai, Tamil Nadu 600077, India
drkoteeswarans@saec.ac.in

² SRM Institute of Science and Technology, Chennai, Tamil Nadu 603203, India
prisuresh@yahoo.com

³ Shiv Nadar (Institution of Eminence Deemed to Be University), Greater Noida,
Uttar Pradesh 201314, India

⁴ Symbiosis International (Deemed University), Pune, Maharashtra, India

⁵ Manipal University Jaipur, Jaipur, Rajasthan 303007, India
sushama.tanwar@jaipur.manipal.edu

Abstract. The need for early breast cancer diagnosis has given rise to new research directions throughout the past ten years. The organizations of the whole world which (W H O) claims that if breast cancer is detected at an early stage, individuals can receive successful treatment. Searching new techniques and developing new ideas of biotechnology has received a lot of attention as the new values acquired significant modern in image processing and new ways of learning of the machines. By using computer-aided diagnostic (CAD) systems, scientists have developed efficient algorithms that expedite diagnosis while reducing the need for human intervention. In the healthcare industry, (CAD) methods are frequently used to identify and diagnose various disorders. The use of the CAD system has increased in popularity in recent years to improve the accuracy in several fields of inquiry. Conventional use of the networks of neural (CNNs) are a really strong family types of scanned photos recognition problems that have recently been developed. CNNs are a subset of deep, feed-forward ANNs that have been effectively used for image recognition. With the job of categorizing cancer cells in breast biopsy images, intend to present a detailed examination of the performance of various CNN models in this paper.

Keywords: Computer-aided diagnosis (CAD) systems · Breast cancer diagnosis · Classification · Magnetic resonance imaging scans · Neural Network · Image · Predict

1 Introduction

Human body cells typically only divide (reproduce) when new cells are needed. A tumor is a mass of tissue that develops when the proliferation of cells in a particular area of the body is rapid and out of control. The tumor is referred to as benign if the out-of-control cells are normal cells (non-cancerous). The tumor is referred to as malignant if the out-of-control cell growth is abnormal and does not resemble normal body cells (cancerous). The bodily part from which they develop is the term given to the cancer. Breast tissue is where breast cancer starts. Additionally, a process known as metastasis allows cancers to spread to different bodily regions and develop new malignancies. It is the second most frequent malignancy in females after several other classes of cancer. Growing older is the most common risk factor for breast cancer, with 66% of cases being diagnosed beyond the age of 55. After those other cancer classifications, this disease is the only and most deadly cancer that affects women in the United States between the ages of 35 and 54. It also ranks second among all cancer-related causes of death for women. Lesser than 5 to 10 percentage of breast cancers in women have a known to be given upon by their ancestors. Tumors fall into two primary categories: malignant (cancerous) and benign (non-cancerous). A number of cell characteristics are used to determine the type of tumor. This is frequently not affecting the health of these females having false tissue and is frequently not diagnosed as cancer. The two subtypes of malignant tumors are in situ and invasive. These cells are constrained inside most valuable parts of the breast containing the process of process in the event of in-situ carcinoma. However, if the cancer cells have gone outside of the ducts, then it is an invasive kind of cancer. Pathologists should concentrate on testing physical histopathology images. By using the essential tasks that are completed automatically by the system for diagnostic approaches, the performance of the breast cancer diagnosis was optimized. These recently been used extensively in the classification and diagnosis of breast cancer, among other medical activities. The main parts and the issue with the tough parts and their images makes it difficult to classify breast cancer. To extract features, a very under the tissue technique and adequate and how those features are presented and is needed, have demonstrated their high performance at completing the task. Due to the extensive time required to gather the images and the skill required to label the images, one of the major obstacles of using CNN.

2 Related Work

Mobark Nada et al. The CBIS-DDSM dataset suggests using CoroNet to carry out the automated detection of BC. [1]. Ahmad Shahab et al. intended to use the Alex net-GRU model, which was published in the paper A Novel Hybrid Deep Learning Model for Metastatic Cancer Detection, to categorize lymph node breast cancer detection[2]. Filipczuk and George used the Otsu-threshold and circular Hough transform algorithms to extract the nuclei-related texture, shape, and characteristics [3]. Kowal et al. used 500 photos representing 50 patients to train a classifier utilising topological and morphological parameters. They were between 84% and 90% accurate [4]. Using general characteristics, Brook A. D. et al. used SVM to diagnose breast cancer from biopsy

pictures [5]. A new method for diagnosing breast cancer was developed by Zhang and Bailing [6]. It entails serial fusing of Random Subspace ensembles applied to biopsy pictures. A research on the Image net big scale visual recognition challenge was carried out by Olga Russakovsky and her colleagues [7]. Using convolutional neural networks, Fabio Alexandre Spanhol and associates created a technique for classifying breast cancer histopathology images [8]. Deep neural networks were used by Dan C. Cireşan and colleagues to identify mitosis in breast cancer histology pictures [9]. Convolutional neural networks are used by Angel Cruz-Roa and associates in their approach for automatically identifying invasive ductal cancer in entire slide pictures [10]. In their comprehensive study, Yi-Sheng Sun, along with their research team, meticulously investigated a wide array of risk factors and preventive measures linked to the development and management of breast cancer, providing valuable insights for the medical community and enhancing our understanding of this complex disease [11]. Joaquín Arribas and his team conducted research focusing on the relationship between p95HER2 and breast cancer, providing valuable insights into the role of this biomarker in the disease [12].

3 Implementation

In order to analyze and interpret mammography pictures for the purpose of detecting and diagnosing breast cancer, a complete method that combines human knowledge with state-of-the-art technology is referred to as manual and automated web-based breast cancer diagnosis and interpretation. The process involves two components: manual evaluation by trained medical professionals and automated algorithms integrated into a web-based platform. Medical professionals, such as radiologists or oncologists, play a crucial role in the manual evaluation phase. They carefully examine the mammogram images, looking for signs of abnormal tissue or potential indicators of breast cancer. Their expertise allows them to identify subtle patterns, irregularities, or suspicious lesions that might require further investigation. A data flow diagram, as seen in Fig. 1, is a graphic depiction of the way data moves through a system or procedure. Drawing one for a web-based interpretation system entails figuring out the processes, outputs, and inputs involved, including user requests, data processing, and results display.

3.1 Algorithm 1: Breast Cancer Diagnosis with Enhanced Techniques

This algorithm incorporates novel techniques to improve breast cancer diagnosis using a Computer-Aided Detection (CAD) system.

Input: Medical image (e.g., mammogram) represented as a matrix $\mathbf{I} \in \mathbb{R}^{(m \times n)}$, where m and n are the image dimensions (height and width).

Output: Final diagnosis result (benign, malignant, inconclusive).

Step.1: Preprocessing:

- **Domain-Specific Denoising:** Apply a mammogram-specific denoising filter (e.g., bilateral filtering) represented by a function $\mathbf{D}(\mathbf{I})$ to reduce noise while preserving microcalcifications.

- **Contrast Enhancement with HE (Histogram Equalization):** Enhance contrast using HE represented by a function $H(D(I))$ to improve visualization of suspicious lesions.
- **Normalization:** Normalize pixel intensity values to a standard range (e.g., 0–1) using a function $N(H(D(I))) \in R^{(0, 1)}$ for improved feature extraction.

Step.2: Feature Extraction:

- **Deep Learning Feature Extraction:** Utilize a pre-trained convolutional neural network (CNN) model denoted by $f\_CNN(N(H(D(I)))) \in R^d$, where d is the dimensionality of the extracted deep learning features.
- **Texture Feature Extraction:** Extract traditional texture features (e.g., Haralick features) represented by a function $f\_texture(N(H(D(I)))) \in R^t$, where t is the dimensionality of the extracted texture features.

Step.3: Feature Selection:

- **Hybrid Feature Selection:** Combine PCA with a feature selection technique that analyzes the importance of features learned by the CNN (e.g., LASSO regression). This selects the most informative features from both sets. Denote the selected features by a matrix $F \in R^{(d + t \times k)}$, where k is the number of selected features.

Step.4: Classification:

- **Ensemble Learning:** Train an ensemble of machine learning classifiers (e.g., SVM and Random Forest) represented by models $C_1(F), \dots, C_e(F)$, where e is the number of classifiers in the ensemble. During classification, combine the predictions from each classifier using a function $g(C_1(F), \dots, C_e(F))$, such as majority voting.

Step.5: Evaluation:

- **Advanced Metrics:** Calculate accuracy (ACC), sensitivity (SEN), specificity (SPE), and Area Under the ROC Curve (AUC) to comprehensively evaluate the CAD system's performance. These metrics are functions of the predicted labels and ground truth labels.

Step.6: Diagnostic Result:

- **Confidence Score Calibration:** Train a separate model $m(F)$ to predict a confidence score alongside the class label (benign or malignant). This score reflects the system's certainty in its prediction.

Step.7: Decision Rules:

- If $m(F) > \theta$ (defined threshold) and $g(C_1(F), \dots, C_e(F)) = \text{"benign"}$, return "benign" as the final diagnosis.
- If $m(F) > \theta$ and $g(C_1(F), \dots, C_e(F)) = \text{"malignant"}$, return "malignant" as the final diagnosis.
- If $m(F) \leq \theta$ or $g(C_1(F), \dots, C_e(F))$ is inconclusive, return "inconclusive" and recommend referral to a specialist for further evaluation.

Algorithm 1 proposes a novel approach to breast cancer diagnosis using a CAD system. It starts by preprocessing mammogram images with mammogram-specific denoising and contrast enhancement techniques. A pre-trained CNN model then extracts deep learning features from the preprocessed images. Additionally, traditional texture features are computed. A hybrid feature selection method combines the most informative features

from both sources. We use an ensemble of machine learning models for classification, combining their predictions to enhance robustness. The system outputs not only a class label (benign or malignant) but also a confidence score, indicating its certainty in the prediction. This score, along with accuracy, sensitivity, specificity, and AUC metrics, helps evaluate the CAD system's performance. Based on the confidence score and ensemble prediction, the algorithm outputs a final diagnosis of benign, malignant, or inconclusive, recommending further evaluation by a specialist in the latter case.

The breast cancer diagnosis algorithm using CAD systems consists of the following steps:

- **Image Preprocessing:** The algorithm takes an input medical image (such as a mammogram) and preprocesses the image to remove noise and artifacts, and enhance the contrast of the image. The preprocessed image with reduced noise and enhanced contrast is output.
- **Feature Extraction:** The preprocessed image is then inputted into the algorithm to extract features that are relevant to diagnosing breast cancer. These features may include size, shape, texture, and intensity of any tumors within the image.
- **Feature Selection:** Not all extracted features are equally important for classification, so the algorithm selects the most relevant features using techniques such as PCA (Principal Component Analysis) or other feature selection methods. This increases the performance of the machine learning algorithms that are employed later in the process by reducing the dimensionality of the input feature space.
- **Classification:** Using labeled data, the method trains a machine learning algorithm, such as a neural network, decision tree, or Support Vector Machine, to categorize chosen traits as either benign (non-cancerous) or malignant (cancerous). Next, the chosen characteristics are utilized to categorize each picture as either benign or malignant using the trained model.
- **Evaluation:** The system's performance is assessed by comparing the class labels that the algorithm predicted to the ground truth labels, which are the diagnoses that came from a biopsy or other clinical assessments. These labels are then used to determine the CAD system's accuracy.

3.2 Image Pre-processing

After collecting the data from the site described have to collect the information of the scanned pictures of mammograms which are being used in our system for further results. These help to make them more prepared for the model and make it more accurate for our system.

- **Image Acquisition** – Obtaining images of the benign and malignant cancer cells from the BreakHis dataset.
- **Image Enhancement** – Adaptive Histogram Equalization for image contrast enhancement.
- **Image segmentation** – Using the tumor region's coordinates, to focus on the tumor location only, the photos are further downsized.

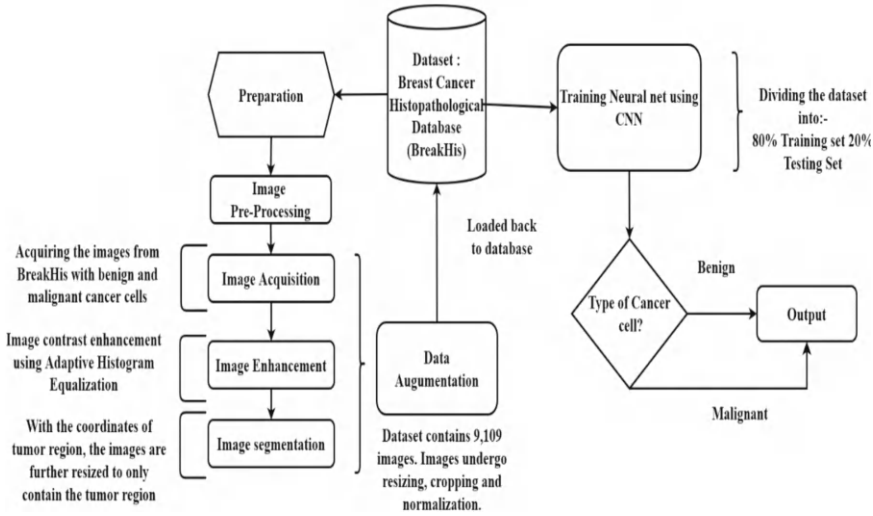


Fig. 1. Data flow diagram for Web – Based Interpretation

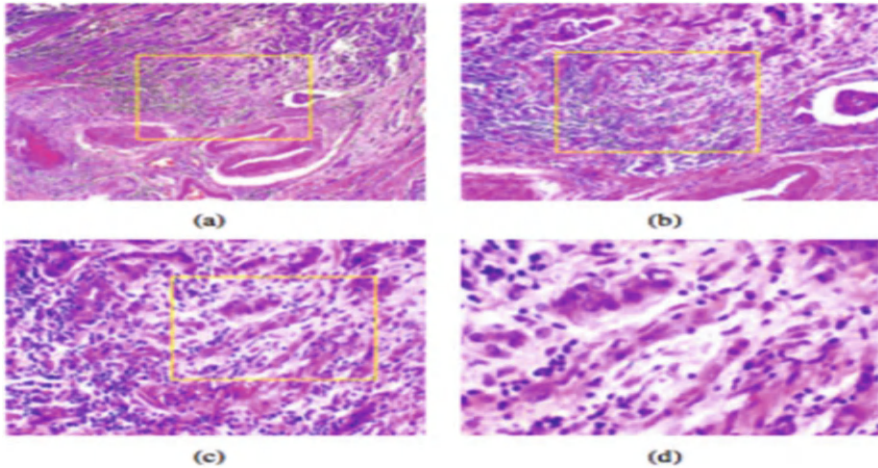


Fig. 2. A slide of breast malignant tumor (stained with HE) seen in different magnification factors: (a) 40X, (b) 100X, (c) 200X and (d) 400X

3.3 Data Augmentation

In our system every steep is complicated and is further used to have the values for changing the system and making it more accurate for any other types of medical termed problems which will make the system more developed. Here comes the augmentation of the scanned pictures which helps the system to get more and more several types of pictures to be used by our system.

- Dataset contains 9,109 images.

- Images undergo resizing, cropping and normalization.
- Images are then loaded back into the dataset.

3.4 Dataset

In this section, we examine the experimental evaluation of the proposed method on two publicly available datasets. We also compare the results obtained from the proposed method with those achieved by previous methods.

Characteristics: The major more than one category of these types and modes of cancer are known and concerned as benign and tumour cancer. History has given use with more than few several cancers but is also known to be not curable until now as the breakdown of the important parts of the breasts are now being made to be more accurate in treatment. Normal benign tumors are limited, slowly developing growths that are regarded as “innocent.” Because a cancer lesion may spread to distant places and infect other tissues, causing harm and ultimately death, it is considered a malignant tumor. Figure 2 shows that the BreakHis dataset, which contains thousands of breast cancer histopathological images in different magnifications with detailed descriptions and specifications, includes a sample image. The samples in the dataset for the current edition were obtained using a number of additional techniques that will be applied to the collected dataset and published further under our system’s procedures in order to be required and used in further modern medicinal techniques to cure all diseases, both similar and unrelated, and are carried out in a hospital while general anesthesia is in effect. These all are based on the numerous particles present inside one scanned picture known as mammogram and will be used by use with different augmentations of these pictures to be made more accurate and better system and future enhancements. Breast tumors, both benign and malignant, may be categorized into a wide range of groups based on the appearance of the tumoral cells under a microscope. The prognosis and treatment outcomes of various breast tumor types and subtypes may differ.

The BACH dataset contains 400 stained microscopic images and 30 whole slide images of breast tumor tissue. While the specific image characteristics are not explicitly mentioned, it is likely that the dataset has similar microscopic image format and characteristics as the BreakHis dataset. The original categorization of the dataset includes four classes: normal, benign, in situ carcinoma, and invasive carcinoma, with 100 images each. However, for the purpose of the research mentioned, the classes were adjusted to align with the BreakHis classification, where normal and benign became benign, and in situ carcinoma and invasive carcinoma became malignant. The BACH dataset is primarily used to evaluate the generalizability of models trained on other datasets, such as BreakHis, by testing their performance on unseen data.

Combining the BreakHis and BACH datasets for breast cancer research can offer significant benefits, but it is essential to consider certain factors before proceeding. One advantage is the increased data size, which can enhance model performance and generalizability by providing a larger pool of training images. Additionally, combining the datasets can lead to an enhanced class representation, particularly if the class distributions differ between them, resulting in a more balanced representation. However, there are challenges to consider, such as the potential data heterogeneity between BreakHis and BACH, including variations in image characteristics like resolution and magnification levels. Addressing these differences might be necessary for effective training.

Another challenge is ensuring label compatibility, as BACH originally has four classes, which need to be properly mapped to the two classes used in BreakHis. Deciding on a training strategy is crucial, and options include training a single model on the entire combined dataset or training separate models on BreakHis and BACH and comparing their performance. To successfully combine the datasets, preprocessing steps should be taken to ensure consistent formats, address significant image quality variations, and map BACH labels to the BreakHis classification. During the training phase, the chosen strategy should account for potential data heterogeneity, and techniques like data normalization or augmentation can be employed to address variations. Finally, evaluating the model's performance on unseen data is necessary to assess its generalizability.

3.5 Convolutional Neural Networks (CNNs)

CNNs are a powerful type of deep learning architecture specifically designed for analyzing visual imagery. They excel at tasks like image classification (identifying objects in images), object detection (locating objects in images), and image segmentation (separating objects from the background).

3.5.1 Core Components

Convolutional Layer: The workhorse of a CNN, responsible for extracting features from the input image. It uses learnable filters, which are small matrices that slide across the image. Each filter detects specific patterns or features in the image. The output of a convolutional layer is a feature map that highlights the locations where the filter detected its intended feature.

Pooling Layer: This layer reduces the dimensionality of the data from the convolutional layers. Techniques like Max Pooling take the maximum value from a defined window (like a 2x2 grid), effectively shrinking the image size while preserving important features like edges and corners.

Activation Function: These functions introduce non-linearity into the network. This is crucial because without it, a CNN would simply be a stack of linear layers, which are limited in their ability to learn complex patterns. A common activation function is ReLU (Rectified Linear Unit), which allows only positive values to pass through, essentially creating a threshold for feature importance.

3.5.2 Training Process

Training CNNs involves feeding them massive datasets of labeled images. Each image has a corresponding label that identifies the content within it. The model uses backpropagation, a powerful algorithm, to learn from its mistakes. Backpropagation calculates the error between the model's prediction and the actual label, and then propagates this error back through the network. This allows the model to adjust the weights and biases of its filters in each layer, gradually improving its ability to recognize features and classify images accurately.

However, training CNNs comes with challenges:

Overfitting: This occurs when the model memorizes the training data too well and performs poorly on unseen images. Techniques like dropout (randomly dropping neurons during training) and data augmentation (artificially expanding the dataset) help to prevent this.

Vanishing Gradients: In deep networks, gradients can become very small during back-propagation, making it difficult to train the earlier layers. Techniques like skip connections (introduced in ResNet) help address this by allowing gradients to flow directly from earlier layers to later ones.

3.5.3 Popular Architectures

- Alexnet is a deep neural network composed of eight layers that contain learnable parameters. In five of the model's layers, Relu activation is used. In the output layer, max pooling is applied first, then three fully connected layers.
- ResNet: A Residual Network concept is created by this design to address the issue of disappearing gradients. This network makes use of a method called skip connection. The skip connection links straight to the output after skipping a few stages of training.
- A common deep Convolutional Neural Network (CNN) design with many layers is the Visual Geometry Group (VGG). The large number of convolutional layers is often described as “deep”; VGG-16 and VGG-19 include 16 and 19 such layers, respectively. Advanced object recognition models have benefited greatly from the VGG architecture, and the VGGNet—a deep neural network—has outperformed benchmarks on a range of tasks and datasets, even those outside of the ImageNet dataset. It continues to be one of the most popular image recognition architectures on the market today.
- Shorter connections between layers are used in the DenseNet, or Dense Convolutional Network, architecture to boost training efficiency and deepen deep learning networks. In a DenseNet, each layer is connected to every other layer below it, creating a convolutional neural network that maximizes information flow between network tiers. Specifically, the first layer is connected to the second, third, fourth, and so on, while the second layer is connected to the third, fourth, fifth, and so on. This approach allows each layer to receive input from all earlier level.

3.6 Algorithm Used

Begin by using a dataset for training consisting of photos of N quantity and all of which are assigned to 1 of 2 classes. Then train the classifier using the given dataset to teach it to distinguish features of each different class. Once training is complete, assess the performance that of the classifier and ask it to predict the labels of latest set of photos that had not been seen before. Will then compare the predicted labels with the actual labels of these photos to assess its accuracy.

3.7 Performance Matrix

An $N \times N$ matrix called a confusion matrix, where N is the number of target classes, is shown in Fig. 3 and is used to analyze data in a classification model. The matrix provides

a thorough knowledge of the model's efficacy and the kinds of mistakes it is producing by comparing the projected target values by the machine learning model with the actual target values. We can learn more about the effectiveness of our classification model and pinpoint any areas that may need improvement by looking at the confusion matrix.

3.8 Medical Image Analysis Terms

Classification models are assessed using a variety of performance criteria, including as ROC curve, precision, recall, F1 score, and specificity. Recall is the percentage of real positive instances that the model properly detected, while precision is the percentage of correctly predicted positive cases out of all projected positive cases [13]. The F1 score provides a more thorough evaluation by combining recall and accuracy into a harmonic mean. The percentage of real negative situations that the model accurately detected is measured as specificity. The ROC curve and AUC, which provide a thorough assessment of the model's performance, are shown in Fig. 4. Confusion matrices at all threshold levels are used to create the ROC curve, which summarizes the performance of the model [14]. AUC, which ranges from 0 to 1, is a numerical indicator of the binary classifier's efficacy that quantifies how effectively the model can discriminate between positive and negative classifications.

Confusion Matrix		
	Actually Positive (1)	Actually Negative (0)
Predicted Positive (1)	True Positives (TPs)	False Positives (FPs)
Predicted Negative (0)	False Negatives (FNs)	True Negatives (TNs)

Fig. 3. Confusion matrix graph extracted from the model

3.9 Formula Used

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

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

$$F1 = \frac{2 * (Recall * Precision)}{(Recall + Precision)} \quad (3)$$

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

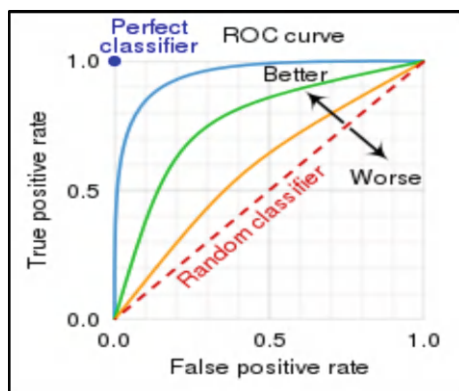


Fig. 4. ROC curve

$$Specificity = \frac{(TN)}{(TN) + (FP)} \quad (5)$$

Precision (Eq. 1), calculated as $TP/(TP + FP)$, measures the accuracy of positive predictions. Recall (Eq. 2), calculated as $TP/(TP + FN)$, indicates the model's ability to identify all positive instances. The F1 score (Eq. 3), which is the harmonic mean of precision and recall, is calculated as $2 * (Recall * Precision)/(Recall + Precision)$. Accuracy (Eq. 4), representing the overall correctness of the model, is computed as $(TP + TN)/(TN + FP + FN + TP)$. Specificity (Eq. 5), which measures the proportion of true negatives correctly identified, is given by $TN/(TN + FP)$.

4 Result and Discussion

The images in the collection were categorized as either benign or malignant. The dataset for the experiment was split up as follows:

Classes: ['Benign', 'Malignant'].

Table 1. Comparison between different models result

	Accuracy	Precision	Recall	F1 score	Jaccard Score	ROC_AUC Score	Specificity	Training Time (mins.)
AlexNet	89%	0.9017	0.8994	0.8959	0.8555	0.8555	0.7379	25.6478
VGG19	88%	0.8850	0.8817	0.8765	0.7853	0.8290	0.6875	38.3050
ResNet152	90%	0.9104	0.9082	0.9054	0.8302	0.8680	0.7600	43.4959
DenseNet201	89%	0.8972	0.8956	0.8922	0.8092	0.8522	0.7358	37.7008

The dataset comprises a total of 2 classes. For the training phase, 80% of the dataset is allocated, while 20% is reserved for testing. The dataset is further divided into batches

for efficient processing: there are 99 training batches, 7 testing batches, and 7 validation batches. The following accuracy levels were reached following the completion of the training, validation, and testing process:

Table 1 clearly shows that the CNN model, ResNet152, demonstrated the highest overall accuracy of 90%. However, the cancerous cells were the focus of our attention, and all four CNN models demonstrated cutting-edge performance in completing the classification assignment.

Table 2. ResNet152 Result

	Test Loss	Test Accuracy of Benign (%)	Test Accuracy of Malignant (%)	Test Accuracy – Overall (%)
AlexNet	0.2484	73	97	89
VGG19	0.3056	68	97	88
ResNet152	0.2155	76	97	90
DenseNet201	0.2609	73	96	89

Table 2 indicates that ResNet152 outperforms the other three CNN models employed in the experiment, with higher values for all of the analysis metrics. The interconnected nature of all the layers in ResNet152 results in a lengthy computing procedure, which accounts for the higher value of training time. Additionally, despite AlexNet’s simpler architecture, it performs on par with other models that are significantly denser. AlexNet offers 89% accuracy and significant values for other performance analysis criteria while requiring less training time. As a result, researchers can use this approach to address related issues quickly.

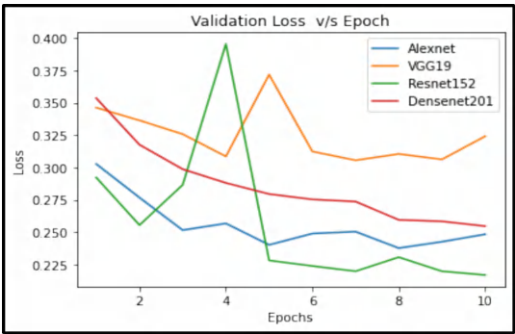


Fig. 5. Validation loss v/s epochs plot

Figure 5 shows that results provide insight into the differences in test loss between four machine learning models (AlexNet, VGG19, ResNet152, and DenseNet201), suggesting that ResNet152 is the most accurate based on test loss. However, other factors

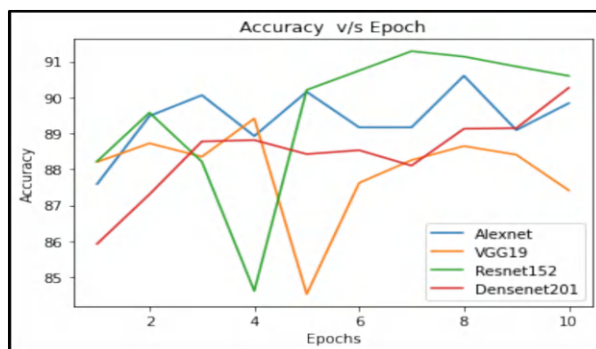


Fig. 6. Accuracy v/s epochs plot

such as model size, training data, and hyperparameters could also affect accuracy and speed and should be considered when choosing a model. Figure 6 provides test accuracy (%) for four machine learning models for various tasks, showing that ResNet152 has the highest accuracy (90%), followed by AlexNet and DenseNet201 (both at 89%), and VGG19 (88%).

5 Conclusion

Assessed several convolutional neural network models' abilities to distinguish between benign and malignant cells in pictures of breast biopsies stained with hematoxylin and eosin. A large collection of over 9,000 RGB image patches with various magnification factors served as the basis for all structures. The CNN model ResNet152 beat all other models in our testing, with an accuracy of 92%. AlexNet and DenseNet201, both of which produced accuracy of 89%, came next. The performance of VGG19 was impressive, with an accuracy of 88%. Additionally, ResNet152 clearly outperformed all other models as seen by its higher scores on many medical image analysis measures, including precision, recall, specificity, and others. When compared to the other three models, the ResNet152 model took the longest to train—roughly 44 min. On outdated hardware, CNN's learning process still requires a lot of work. As a result, developing a currently difficult, but the use of such neural networks in hardware can help doctors detect breast cancer early. Intend to create a better enhanced and special interface for the program designed and made the help for modern medicinal technologies and also, cardiac arrhythmia, and diabetic retinopathy problem. That will cover the most delicate parts of human body like heart, lungs, brain etc. The training process of CNNs still demands significant computational power even on older hardware. Nonetheless, utilizing such neural networks on specialized hardware can aid medical practitioners in detecting breast cancer at an early stage. Our team aims to develop an optimized and sophisticated user interface for our program that can also address other critical healthcare issues such as cardiac arrhythmia and diabetic retinopathy, which affect vital organs such as the heart, lungs, and brain. By leveraging the latest advancements in medical technology, aspire to revolutionize the healthcare industry and improve the quality of life for people.

References

1. Kowal, M., Filipczuk, P.: Nuclei segmentation for computer-aided diagnosis of breast cancer. *Int. J. Appl. Math. Comput. Sci.* **24**(1), 19–31 (2014)
2. George, Y.M., Zayed, H.H., Roushdy, M.I., Elbagoury, B.M.: Remote computer-aided breast cancer detection and diagnosis system based on cytological images. *IEEE Syst. J.* **8**(3), 949–964 (2013)
3. Belsare, A.D., Mushrif, M.M., Pangarkar, M.A., Meshram, N.: Classification of breast cancer histopathology images using texture feature analysis. In: *Tencon 2015–2015 IEEE Region 10 Conference*, pp. 1–5. IEEE (2015)
4. Brook, A., El-Yaniv, R., Isler, E., Kimmel, R., Meir, R., Peleg, D.: Breast cancer diagnosis from biopsy images using generic features and SVMs. In: *IEEE Transactions on Information Technology in Biomedicine* (2006)
5. Yamini, B., et al.: Predictive modelling for lung cancer detection using machine learning techniques. In: *Proceedings of the 8th International Conference on Communication and Electronics Systems, ICCES*, pp. 1220–1226 (2023). <https://doi.org/10.1109/ICCES57224.2023.10192648>
6. Zhang, B.: Breast cancer diagnosis from biopsy images by serial fusion of Random Subspace ensembles. In: *2011 4th International Conference on Biomedical Engineering and Informatics (BMEI)*, vol. 1, pp. 180–186. IEEE (2011)
7. Geetha, R., Sivasubramanian, S., Kaliappan, M., Vimal, S., Annamalai, S.: Cervical cancer identification with synthetic minority oversampling technique and PCA analysis using random forest classifier. *J. Med. Syst.* **43**, 1–19 (2019)
8. Josephine, M.S., Lakshmanan, L., Visu, P., Ganesan, R., Jothikumar, R.: Monitoring and sensing COVID-19 symptoms as a precaution using electronic wearable devices. *Int. J. Pervasive Comput. Commun.* **16**(4), 679–694 (2020). <https://doi.org/10.1108/IJPC-06-2020-0067>
9. Russakovsky, O., et al.: Imagenet large scale visual recognition challenge. *Int. J. Comput. Vision* **115**, 211–252 (2015)
10. Neeba, E.A., Koteeswaran, S., Malarvizhi, N.: Swarm-based clustering algorithm for efficient web blog and data classification. *J. Supercomput.* **76**(6), 3949–3962 (2017). <https://doi.org/10.1007/s11227-017-2162-z>
11. Spanhol, F.A., Oliveira, L.S., Petitjean, C., Heutte, L.: Breast cancer histopathological image classification using convolutional neural networks. In: *2016 International Joint Conference on Neural Networks (IJCNN)*, pp. 2560–2567. IEEE (2016)
12. Harin Fernandez, F.M., et al.: Personalized ontology and deep training tree-based optimal gated recurrent unit-recurrent neural network for prediction of students' behavior. *Concurrent. Comput.: Pract. Experience* (2023). <https://doi.org/10.1002/cpe.7420>
13. Cireşan, D.C., Giusti, A., Gambardella, L.M., Schmidhuber, J.: Mitosis detection in breast cancer histology images with deep neural networks. In: Mori, K., Sakuma, I., Sato, Y., Barillot, C., Navab, N. (eds.) *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2013: 16th International Conference, Nagoya, Japan, September 22–26, 2013, Proceedings, Part II*, pp. 411–418. Springer Berlin Heidelberg, Berlin, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40763-5_51
14. Nair, R.R., David, E., Rajagopal, S.: A robust anisotropic diffusion filter with low arithmetic complexity for images. *EURASIP J. Image Video Proc.* **2019**(1), 679–694 (2019). <https://doi.org/10.1186/s13640-019-0444-5>



Social Media and Stress Detection: A Machine Learning Approach

Rutuja Kamble^(✉), Piyush Borhade, Urjita Garud, and S. M. Mali

Dr. Vishwanath Karad University, MIT-WPU, Kothrud, Pune, India
{rutuja.kamble,piyush.borhade,urjita.garud,
shankar.mali}@mitwpu.edu.in

Abstract. Social media has now become a focal point in many lives, and with it, so do the aspects that machine learning is focused on. One such is the detection of stress, which has been practiced using natural language and machine learning models processing techniques on Reddit posts. Researchers have applied machine learning to classify posts as stressful or non-stressful to identify inferred stress indicators in text found in social media. Indeed, machine learning algorithms effectively identify stress levels via analyzing the language and sentiment expressed in these posts. Beyond just stress detection, machine learning has also been in use to identify extremist ideation, filter radicalized content, and detect hate speech on social media platforms. The detection of rumors and fake news on social media has been further enhanced by deep learning models, in general, integrating machine learning techniques with social media analysis has opened doors for automated detection and classification of various psychological and behavioral aspects, contributing to a deeper understanding of user interactions and content dissemination on these platforms.

Keywords: Sentimental Analysis · NLP · Random Forest · Machine Learning · Mental Health

1 Introduction

Indeed, social media platforms have proven to be a fertile ground as a data source for various applications, including mental health analysis. Machine learning techniques have been used to find out stress and other mental health conditions from social media content. One such aspect is the detection of stress, which has been achieved through the application of natural language processing and machine learning techniques on Reddit posts [1]. Research has even studied ways of detecting at the individual level the psychological stress through the application of deep neural networks to social media data. These methods aim at automating the process of stress detection through the analysis of users' online behavior and language. In the past few years, there is a growing interest in applying social media data to a lot of different purposes other than just social networking, such as healthcare and mental health research. An area that has been particularly targeted in this direction is the detection and comprehension of the stress levels across social media

users. Stress, a pervasive aspect of modern life, has the power to disrupt individuals' well-being, productivity, and quality of life. Early identification and dealing with stress are critical to prevent adverse health outcomes and promote overall mental wellness. In the contemporary digital landscape, the social media platforms have risen to powerful mediums for communication, dissemination of information, and social interaction. It is the large user population of billions, active on sites such as Facebook, Twitter, Instagram, and Reddit, which means these virtual spaces have become integral parts of everyday life for many of the people worldwide, with the tremendous amounts of information which reflect the thoughts, emotions, and behaviors of the users posted on these platforms, therein lies a treasure trove. Also, some researchers have also applied machine learning to other areas such as rumors identification and extremist ideation detection. Researchers have developed supervised machine learning algorithms to identify rumors and extremist content on social media platforms, where computational linguistic analysis combined with expert input is important for accurate detection. Additionally, high-performance techniques in handling large volumes of data have been emphasized as how the skills of these advanced algorithms and techniques prove fruitful in these modern times of full-scale social media. This research will bring about a paradigm change for which various aspects of mental health in the digital age can be understood and combated. By tapping into the vast quantities of data from social media across the globe, we will get further into the intricate symbiotic relationship between social interactions, emotional states, and mental well-being. Besides, with automated tools to detect stress, early intervention strategies, personalized support systems, and community-based interventions aimed at promoting mental wellness in online environments can be facilitated.

2 Literature Survey

The research addresses the use of machine learning to detect stress from social media data. Previous studies have tested different methods like sentiment analysis and deep learning to find signs of stress in social media posts. While these methods show potential, challenges like cleaning up the data and keeping user information private remain. This survey aims to give a summary of what has been done so far and what exactly is achieved in this field.

Mochamad Rizky, Febriansyah, et al. developed a Stress Detection System for Social Media Users [1], employing SVM, Naïve Bayes, Decision Tree, Bag of Words, and TF-IDF on Dreddit (Reddit posts). Their system achieved an 80.00% F1 Score and 75.00% Accuracy [1], demonstrating effective stress detection capabilities.

ORYNGOZHA NAZZERE, et al. implemented a Stress-Related Posts Detection and Analysis system [3] in Reddit's Academic Communities, employing Support Vector Machine (SVM), Logistic Regression, Naive Bayes, and Long Short-Term Memory (LSTM) on Dreddit (labelled Reddit). They achieved a notable 79% F1 score and 77.78% accuracy, showcasing effective stress-related post detection capabilities.

Tadesse Michael M., et al. developed a Depression-Related Posts Detection system for Reddit, utilizing SVM, Random Forest, Adaptive Boosting, and MLP. Their model achieved 80% accuracy and 80% F1 score with the best single feature, and 91% accuracy and 93% F1 score with combined features, showcasing robust detection capabilities for depression-related posts.

G Aishwarya, et al. implemented a Stress Detection system for Social Interactions, employing Logistic Regression, Decision Trees, and Random Forest. Their model achieved an impressive accuracy of 98.87% on Twitter tweets dataset, highlighting its efficacy in detecting stress from social interactions using ML and NLP techniques.

Rastogi, Aryan, et al. conducted a Stress Detection study from Social Media Articles [11], employing Support Vector Machine (SVM), Random Forest, and Multilayer Perceptron (MLP) on Reddit and Twitter datasets. They achieved an accuracy of 93.76% and F1 score of 96.52% for Reddit, and 94.72% accuracy and 96.52% F1 score for Twitter [11], establishing a robust benchmark for stress detection.

Jadhav, Sachin, et al. investigated Text-Based Stress Detection Techniques using Social Media [6], employing Bidirectional Long Short-Term Memory (BLSTM) and Support Vector Machine (SVM) [6] on various social media sources like tweets, chats, and comments. Their study achieved an impressive accuracy of 93% [6], showcasing effective stress detection capabilities.

Jain, Amit, and Muskan Kumari explored stress prediction utilizing Machine Learning and IoT techniques [7], employing SVM, KNN, Random Forest, and CNN on EDR, ECG, and WESAD data. Their study achieved an accuracy ranging from 85.71% to 98.6% [7], demonstrating promising results in stress predictions.

3 Methodology and Algorithm

3.1 Implemented Algorithms

Decision Trees Classifier

This is one commonly used algorithm in Machine Learning. It's a supervised algorithm and a tree-based model represented by if-then-else statements. It can solve both regression and classification problems. It solves complex decision-making problems. The name "decision tree" is rather tone-explicit, still, because the tree-suchlike structure is basically comprised of bumps. Those internal bumps denote a particularity (decision the branches denote the issues of that particularity, and the leaf bumps represent a class marker (in type) or a numerical value (in regression)).

Random Forest

Random Forest is a ML method that can be implemented to both classification and regression problems. It creates multiple decision trees during training, which are used to provide solutions for both types of problems like classification and regression. Every tree within the random forest is constructed using a randomly selected subset of the training data points and a randomly selected subset of the features. This noise can make a difference to decrease overfitting and make strides generalization execution. During expectation, it just totals the expectations of all trees to create the last result. It is known for vigor, adaptability, and capacity to handle high-dimensional information. They are commonly utilized over diverse spaces for their flexibility and their viability when managing with complex.

Logistic Regression

Logistic regression is widely used when the dependent variable is binary, such as yes/no,

success/failure, or true/false outcomes. It provides a probabilistic framework for binary classification, making it invaluable in fields like medicine (disease diagnosis), marketing, and finance (loan approval) (Fig. 1).

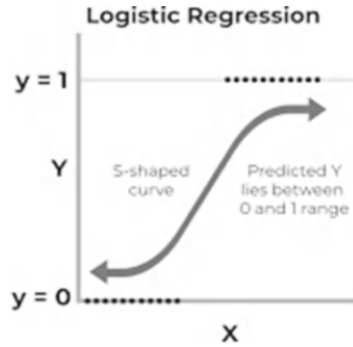


Fig. 1. Logistic Regression graph

Coefficients are estimated for each feature, representing their impact on the predicted probability. Training can be done with methods such as maximum probable estimation or gradient descent during the optimization of the coefficients. The model which is trained is interpretable, providing insights into the effect of each feature on the outcome. This regression method is used widely in diverse fields for its simplicity and efficiency in handling linearly separable datasets.

Bernoulli Naïve Bayes

This is very useful naïve Bayes model. The theory behind this algorithm is that the feature binary (0s and 1s) in nature. One important practice in text classification using the ‘bag of words’ model would be applying a Bernoulli Naïve Bayes classifier to a text document, essentially classifying the text according to its features. The ‘sklearn.naive_bayes.BernoulliNB’ class can be used in the Scikit-learn package to provide the implementation of the Gaussian Naïve Bayes algorithm. It operates under Bayes’ theorem, which It models the possibility of each feature occurring in each class using the Bernoulli distribution, which represents binary outcomes (presence or absence). It estimates probabilities of each trait given the class label during training, as well as prior chances of each class (Fig. 2).

In prediction, it computes the posterior probability for each class of a given instance via Bayes’ theorem and then opts for the class with the higher chances. This classifier is esteemed for its straightforwardness, efficacy, and efficiency, particularly when dealing with binary feature sets, such as those encountered in text classification [11].

Multinomial Naive Bayes (MNB)

It is most dynamic ML algorithm that is also based on Bayes’ theorem. It is frequently used in text analysis scenarios involving discrete data such as word frequencies in documents, Multinomial Naive Bayes stands out as a widely employed probabilistic classification algorithm and other contexts wherein features represent counts (Fig. 3).

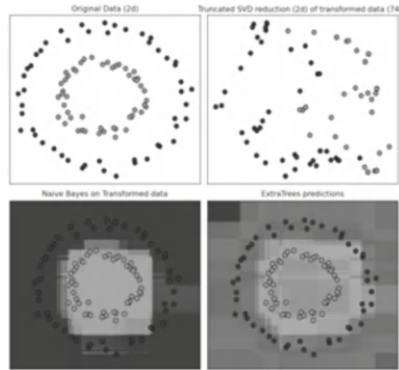


Fig. 2. Bernoulli Naïve Bayes

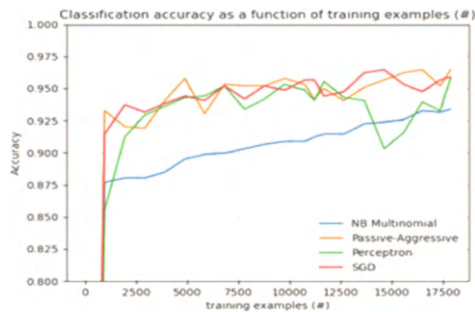


Fig. 3. Multinomial Naive Bayes

It is based on the Naive Bayes principle, assuming that traits are tentatively independent given the class label. Unlike Bernoulli Naive Bayes, which models binary features, it is designed for features that represent counts or frequencies of events. This makes it particularly suitable for tasks like document classification, where features might include word counts or term frequencies. During training, it estimates the probability of each feature occurring in each class, usually using maximum likelihood estimation. In prediction, it computes the posterior possibility for a given instance using Bayes' theorem and chooses the class with the highest probability. Multinomial Naive Bayes is esteemed for its straightforwardness, efficiency particularly in contexts where text data is involved.

Artificial Neural Network

ANN imitates the human nervous system to solve problems in a parallel manner. Artificial Neural Networks (ANN) are computational models in which the model's complexity is modelled on the complexity of brain neural networks (Fig. 4).

Outputs and predicted values using methods such as backpropagation. ANNs are versatile and can handle relationships that are nondifferentiable, meaning that they can be used in different kinds of applications, such as NLP, Image/Speech recognition. The model consists of nodes connected in layers, with information flowing through the

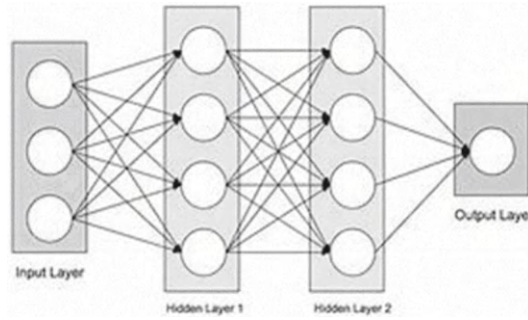


Fig. 4. Diagrammatic representation ANN [2]

network and processed with each node in each layer. ANNs are main used in supervised learning tasks, such as classification, pattern recognition and regression.

Support Vector Classification (SVC)

This is a supervised learning algorithm used for classification tasks. The main idea behind it is finding the best hyperplane that can be used to separate classes in feature space. It does this by maximizing the margin between them, which helps improve the algorithm’s generalization ability (Fig. 5).

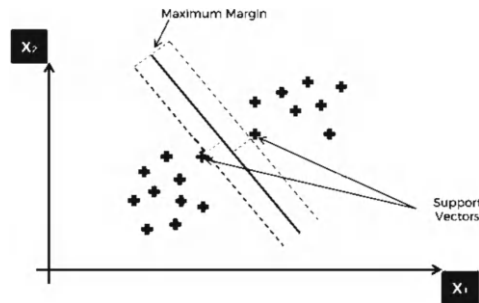


Fig. 5. Support Vector Classification representation

Dealing with Non-Linear Data is done by SVC to resolve nonlinear classification problems by using kernels and other methods allowing higher-dimensional representation where a separating plane should be easier to find.

3.2 Proposed Architecture

The proposed architecture presents a comprehensive framework for stress detection in social media via machine learning and functionality requirements. It encompasses various components, modules, and their interactions to achieve the desired algorithm behavior and performance We’re focusing on making it able to handle a lot of data and adapt to changes over time. The system will clean up the messy social media data

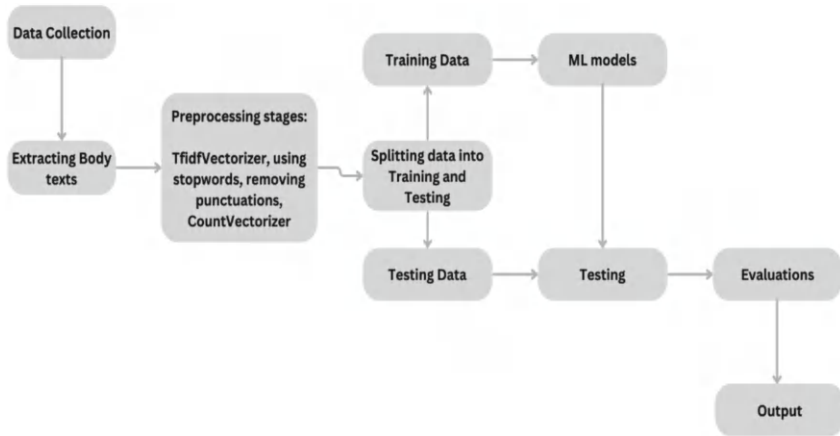


Fig. 6. Sentiment Analysis Methodology

so the algorithm can understand it better. We will test it thoroughly to make sure, it works reliably with the goal of helping improve mental health support online. Sentiment analysis, is implemented in the further architecture. It is a technique in context mining, aims to comprehend emotions and opinions conveyed in text, typically categorizing them as positive, neutral, or negative. Sophisticated applications extend sentiment analysis to interpret intentions, emotions, and urgency conveyed in the content (Fig. 6).

Step 1. Data collection

We extracted data from twitter, which consists of two features i.e. texts and labels. Texts consists of posts created by users on twitter. We collected our data from “Stress Detection from Social Media Articles” with 8900 in size and it was from Kaggle.com.

Step 2. Data Preprocessing

Before proceeding for further evaluation, we refined the data by using some data preprocessing steps. The steps applied are as follows:

- We checked if there are any null values and duplicate values present in our data. We then removed the data containing above values. Then it was part to perform NLP on our text feature.
- As we know that English is widely used in social media platforms, to reduce the unwanted data, data cleaning is performed. Used stop words alteration, such as ‘the’, ‘an’, ‘so’, ‘of’, ‘has’, ‘and’, which are present in abundance in every text but they don’t provide much useful information to the model, so by removing these words, we can focus on the more important information in the text.
- Then we removed the special characters like ‘# \$ @ ! % ^ & * () _ + - ~ ? >’, then we converted all the texts into lowercase.
- After required preprocessing is done, then we moved to our next step in which used Count Vectorizer which converted our text data into the format of numbers then this converted data is given as X and the value of y is given as labels.

Step 3. Model Training and Evaluation

As for model training and evaluation, we followed the below mentioned steps:

- Then we split our data into training and testing data with random state were 11 and test size was 0.2, thus out training data takes 80% of data and remaining 20% to testing data.
- Afterwards we did model training in which we used seven algorithms. Later trained our data with few models like logistic regression, Random Forest classifier, Decision Tree classifier, SVC, Multinomial Naïve Bayes, Bernoulli Naïve Bayes.
- We used optimizer as Adam, loss as binary Corr entropy in Artificial Neural Network. We trained 10 epochs of it with batch size 32 and validation split is equal to 0.1. All the approaches employed in this study will maintain the capability to classify text across all the models, enabling the identification of whether a person is experiencing stress or not.

4 Result Analysis

In this section, we present a comprehensive analysis of the results obtained from our study. The data collected has been meticulously examined and interpreted to uncover meaningful insights and trends. Through careful scrutiny and statistical analysis, we aim to shed light on the key findings, drawing attention to significant observations and correlations. This analysis serves to provide a better understanding of the phenomenon under investigation and offers valuable implications for further research or practical application (Table 1).

Table 1. Result Comparison

No.	Research Paper	Model	Datasets	Performance
1	“Stress detection system for social media users”	SVM, Naïve Bayes, Decision Tree, Bag of words, TF-IDF(Term-Frequency-Inverse Document Frequency) [1].	Dreaddit (posts from Reddit)	80.00% F-1 Score and 75.00% Accuracy [1].
2	“Detection and Analysis of Stress-Related Posts in Reddit’s Academic Communities”	Support Vector Machine (SVM), Logistic Regression, Naive Bayes Long Short-Term Memory (LSTM) [3]	Dreaddit (labelled Reddit)	79% F1 score and 77.78% accuracy [3].
3	“Detection of Depression-Related Posts in Reddit Social Media Forum”	Logistic Regression, SVM, Random Forest, Adaptive Boosting, Multilayer Perceptron (MLP) [8].	Reddit	Best Single feature-80% accuracy score and combined features-91%accuracy

(continued)

Table 1. (continued)

No.	Research Paper	Model	Datasets	Performance
4	“Stress detection using natural language processing and machine learning over social interactions”	Logistic Regression, Decision Trees, Random Forest [4].	Twitter-tweets dataset	98.87%, Max accuracy
5	“Stress Detection from Social Media Articles: New Dataset Benchmark and Analytical Study”	SVM, Random Forest, Multilayer Perceptron (MLP) [11].	Reddit and twitter	For reddit -accuracy 93.76%, For Twitter-Max accuracy 94.72% [11].
6	“Text Based Stress Detection Techniques Analysis Using Social Media”	Bidirectional Long Short-Term Memory (BLSTM), SVM [6]	Social media (tweets, chats, comments)	Max 93% accuracy [6].
7	“Prediction of Stress using Machine Learning and IoT”	SVM, KNN, Random Forest, CNN [7]	EDR, ECG, and WESAD data	Max Accuracy 85.71%-98.6% [7].
8	Our Predicted Model- Social Media and Stress Detection	Logistic regression, Multinomial Naïve Bayes, Bernouli naïve Bayes, Random Forest, Decision Tree, ANN, SVC	Twitter data (Stress Detection dataset)	Max Accuracy 83%, (For Logistic Regression)

5 Experiment Results

Our motive is to give the reader a good idea about which model does well and where they fall short while undertaking this task by means of showing their different metrics comparison such as accuracy, precision-recall, and error rate, below is the formula for calculation of the same (Table 2).

$$\textbf{Precision} = \text{True Positive} / (\text{True Positive} + \text{False Positive}) \quad (1)$$

$$\textbf{Recall} = \text{True Positive} / (\text{True Positive} + \text{False Negative}) \quad (2)$$

$$\textbf{Accuracy} = (\text{True Positive} + \text{True Negative}) / \text{Total} \quad (3)$$

$$\textbf{Error rate} = 1 - \text{Accuracy} \quad (4)$$

Table 2. Experimental analysis of metrices

Sr no.	Algorithms used	Accuracy	Precision	Recall	Error rate
1	Multinomial naïve Bayes	80.45%	77.90	85.28	19.54
2	Bernoulli naïve Bayes	80.69%	77.24	87.26	19.30
3	Logistic Regression	83.56%	81.78	86.56	16.43
4	Random Forest	81.98%	76.83	91.82	17.25
5	Decision Tree	77.57%	77.26	80.37	20.77
6	ANN	81.63%	0.77	0.85	19.27
7	SVC	83.33%	77.44	91.47	17.66

We want people not only understand which models worked better than others but also why so that they can make better decisions themselves when choosing an appropriate model for their own work. Below shown is the experimental analysis based on the above metrices. we examined the performance metrics of several MLA’s including Logistic Regression, Multinomial Naïve Bayes, Bernoulli Naïve Bayes, Random Forest, Decision Tree, ANN, and SVC.

In the below mentioned (Table 3), we checked whether the trained a model is performing well or not while giving the expected results below are few examples of the same.

Table 3. Examples- Sentiment Analysis dataset (training)

id	Text	Label
1	‘Simple things make me happy smiling face with smiling eyes happy simplicity	0
2	‘And now we have been given the walkthru book by and to base our whole school PD on!! grinning face instructionalcoaching excited’	0
3	‘Walked round the whole of notts today feeling tired’	1
4	‘26 Tips to Lose Weight in A Healthy Way tagfire lifestyle healthcare health life LOL healthy Yoga food diet love fitness RT swag stress anxiety happiness nutrition funny hot cool smile’	1
5	‘Wishing YOU Peace Joy & Love! JoyTrain MentalHealth Quote Mindfulness GoldenHearts IAM ChooseLove IAMChoosingLove TuesdayMorning TuesdayThoughts TuesdayMotivation RT’	0

Where ‘0’ means the given text is stress negative i.e. No stress and ‘1’ mean the given text is stress positive i.e. distressing statement. Which stated that the given model is working fine over the selected dataset and giving the results as estimated as we further proceed with the evaluation process.

6 Conclusion

Multiple models evaluated: Logistic Regression, Multinomial, SVC, Random Forest, Decision Tree, and ANN were evaluated for classification. Logistic and SVC top performers: Logistic Regression and SVC achieved the highest accuracy, both exceeding 83%. Similar performance: The accuracy difference between Logistic Regression (83.56%) and SVC (83.33%) is minimal.

Further exploration: Consider hyperparameter tuning for Logistic Regression and SVC to potentially improve performance. Our evaluations gained balanced scores, but they are yet to be scaled up. Our future scope will be changing the hyper parameters to get more accuracy.

Through systematic evaluation and validation, this architecture aims to establish a reliable and efficient tool for real-time stress detection, contributing to the advancement of stress level monitoring and intervention in the virtual era in our coming future.

References

1. Mochamad Rizky, F., et al.: Stress Detection System for Social Media Users (2022). <https://doi.org/10.1016/j.procs.2022.12.183>
2. Rishimohan, et al.: Stress detection based on social media blogs. Stress Detection Based On Social Media Blogs Section A, vol. 2023 (2019). <https://doi.org/10.48047/ecb/2023.12.si4.735>
3. Oryngozha, N., et al.: Detection and analysis of stress-related posts in Reddit's academic communities. IEEE Access **12**, 14932 (2024)
4. Aishwarya, G., et al.: Detection of stress over social interactions using machine learning and natural language processing. In: 2023 6th International Conference on Contemporary Computing and Informatics (IC3I) (2023). <https://doi.org/10.1109/ic3i59117.2023.10397916>
5. Jadhav, S., et al.: Text based stress detection techniques analysis using social media. In: 5th International Conference on Computing Communication Control and Automation (ICCUBEA, 2019)
6. Jain, A., Kumari, M.: Prediction of stress using machine learning and IoT. In: 2022 11th International Conference on System Modeling & Advancement in Research Trends (SMART), no. 978-1-6654-8734-4 (2022). <https://doi.org/10.1109/smart55829.2022.10046907>
7. Tadesse, M.M., Lin, H., Xu, B., Yang, L.: Detection of depression-related posts in Reddit social media forum. IEEE Access **7**, 44883–44893 (2019). <https://doi.org/10.1109/access.2019.2909180>
8. Lin, H., et al.: Psychological Stress Detection From Cross-Media Microblog Data Using Deep Sparse Neural Network
9. Kene, A., Thakare, S.: Smart Technologies, Communication & Robotics (STCR) | 09 -10. No. 978-1-6654-1806-5/21 (2021). <https://doi.org/10.1109/STCR51658.2021.9588803>
10. Rastogi, A., et al.: Stress Detection from Social Media Articles: New Dataset Benchmark and Analytical Study
11. Zhang, L., et al.: Deep learning for sentiment analysis: a survey. WIREs Data Min. Knowl. Discov. (2018). <https://doi.org/10.1002/widm.1253>

AI for Business and Finance



Computational Optimization of Image Feature Extraction Algorithms on PySpark and Hadoop

Arshi Jamal^(✉)  and K. Ramesh 

Karnataka State Akkamahadevi Women's University, State Highway 12,
Athani Road, Jnanashakti Campus, Torvi, Vijayapura 586108, Karnataka, India
arshi.jamal2020@gmail.com, rameshk@kswu.ac.in

Abstract. This paper addresses the computational challenges inherent in image feature extraction, a process that is computationally expensive due to the high dimensionality and large volume of data. Traditional stand-alone or GPU-based machines often face limitations in processing capacity, memory constraints, and scalability issues when dealing with batch feature extraction tasks. These challenges necessitate the exploration of distributed computing platforms for enhanced efficiency and scalability.

We propose a novel approach to image feature extraction utilizing PySpark, a distributed computing system, to harness the power of cluster computing. Focusing on the Local Binary Pattern (LBP) technique, known for its effectiveness in texture classification, we adapt this algorithm to operate within a distributed environment. By leveraging PySpark's ability to parallelize tasks across multiple nodes, we significantly reduce computation time and resource constraints, enabling the processing of large-scale image datasets more efficiently.

Our experiments, conducted on the LFW image datasets, demonstrate a substantial improvement in speed, achieving an N-fold increase in processing speed as the number of nodes is scaled from 1 to N. This scalability showcases the potential of distributed computing platforms like PySpark in overcoming the computational hurdles of image feature extraction, offering a scalable and efficient solution for large-scale image analysis.

1 Introduction

Feature extraction is the major step in pattern recognition applications where input images are represented by discriminative, meaningful, relevant and compact feature descriptors. Improving the computational speed of feature extraction task plays a crucial role as it accounts for significant portion of pattern recognition algorithms time due to its time intensive computational and I/O bound operations. Further, due to unprecedented increase in the volume, velocity and variety of images stored and processed by a large group of data centers

located at geographically diverse locations, incorporating fault tolerance, scalability and availability has become a major problem.

To overcome the above issues, several researchers resorted to big data platforms, broadly categorized into vertical and horizontal tools [1]. Vertical scaling platforms include multi-core systems, GPU and High Performance computing clusters [2–4]. Multi-core processors and GPUs provide excellent support for optimizing computational power of image feature extraction algorithms however, the shared memory space of these systems is not sufficient to store and process large volume of high dimensional images [5]. In contrast to GPU and multi-core based systems, HPC clusters comprises a group of tightly coupled computing nodes interconnected by high throughput networks provide relatively higher computational performance and scalability, and are widely used in processing remote sensing data and medical images [6]. In [7, 8], researchers attempted to improve computational power by combining GPUs with cluster nodes to form a hybrid framework to test remote sensing image datasets. However, hybrid system of GPUs combined with HPC cluster nodes is not a feasible solution when the dataset is exponentially increasing in volume and distributed geographically across various data centers.

Beside vertical scaling platforms, grid computing and Apache Hadoop framework are the two popular horizontal platform tools used for solving big data problems. Grid computing systems are more scalable and inexpensive as compared to vertical scaling platform tools [9, 10]. However, grid computing systems do not guarantee recovery of data or graceful completion of jobs during run time failures. The mechanism of recovering data or restarting of failed jobs during run time failures need to be implemented and incorporated manually using low level programming APIs. In contrast to Grid Computing systems, Hadoop provides distributed storage, computation, automatic data recovery and fault tolerance. Hadoop processes all its input/output and intermediate data reading directly from the disk and more suitable for batch processing applications [11, 12]. Therefore, it is computationally more expensive to run jobs involving iterative tasks. In contrast to Hadoop, Apache Spark is an alternative distributed cluster computing framework invented at the Berkley’s AMPLab, designed to process its intermediate results by utilizing RAM [13, 14]. Due to its in-memory computing capability, it is more suitable for the jobs involving iterative or multiple stages of execution. Therefore, in this paper we propose RDD based distributed frameworks for optimizing LBP feature extraction algorithm and evaluate its performance by comparing with MapReduce based LBP (MRLBP) design implementation.

The main contributions of this work are as follows:

- We have built multi-node Hadoop and PySpark clusters.
- Designed and implemented MapReduce version (MRLBP) and PySpark version (RDDLPB).
- Evaluated the performance of MRLBP and RDDLPB for LFW face image datasets.

- Analyzed computational optimization and scalability by varying the number of nodes.

In this paper, we present a comprehensive study on Local Binary Pattern (LBP) features and their extraction using distributed computing frameworks. The following sections are structured to provide a detailed explanation of our work.

In Sect. 2, we introduce the concept of Local Binary Pattern (LBP) features, explaining their significance and how they are computed. Section 3 describes the design and implementation of LBP feature extraction using the MapReduce framework, highlighting the key components and processes involved. In Sect. 4, we cover the design and implementation of LBP feature extraction using Resilient Distributed Datasets (RDDs) in PySpark, focusing on the differences and advantages over the MapReduce approach. Section 5 presents the datasets used in our experiments, followed by a detailed analysis of the experimental results, comparing the performance of MapReduce and RDD based LBP feature extraction methods. Finally, Sect. 6 provides a summary of our findings, discussing the implications of our work and potential directions for future research.

2 Local Binary Pattern (LBP) Features

LBP is the local visual texture descriptor and more widely used for representing texture features of the image [15, 16]. In LBP feature extraction technique, first image is divided into $m \times n$ grids and LBP binary feature vectors are computed for each pixel of the grid by comparing with the surrounding 8 neighborhood pixels. Histograms of LBP values are computed for $m \times n$ grids to form a final LBP feature vector of size $m \times n \times 255$.

Let $v_c(x, y)$ represents central pixel at position (x, y) and $v_0, v_1, v_2, \dots, v_7$ are the 8 neighborhood pixels then the LBP at any pixel point (x, y) of the image is

$$LBP(x, y) = \sum_{p=0}^7 f(v_p - v_n)2^p \quad (1)$$

where $f(x)$ is a threshold equals to zero when $x \geq 0$ and 1, when $x < 0$.

In the next two sections, we present designing MapReduce (MRLBP) and Spark RDD based (RDDLBP) designs of LBP algorithm on Hadoop Yarn cluster. In the experimental section, we evaluate the performance of RDDLBP algorithm for the large volume of LFW [17] by comparing with MRLBP implementation.

3 Designing MapReduce Based LBP Feature Extraction

The MapReduce design of LBP feature extraction algorithm involves mapping tasks to multiple slave nodes followed by reduction operations [18]. The number of map daemons created during MapReduce execution depends on size of

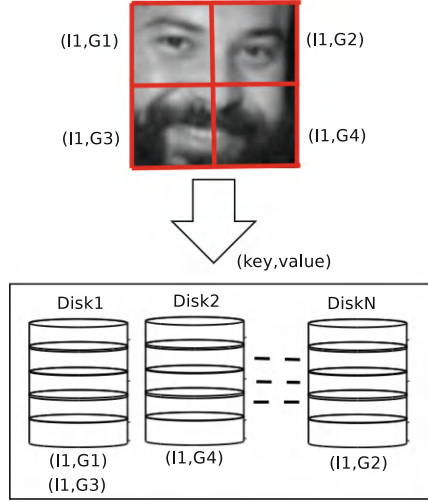


Fig. 1. Partitioning of face image into grids and storing the partitions in the form of a sequence file.

images and the input split conditions. The input dataset is prepared by dividing images into $p \times q$ grids and stored into HDFS (Hadoop Distributed File System) [19] in the form Hadoop binary sequence file with key representing $(file\_name, grid\_id)$ and the value representing array of bytes as shown in Fig. 1.

The MapReduce workflow of designing LBP feature extraction for face images is shown in Fig. 2. Let $i_1, i_2, i_3 \dots i_n$ be the n number of images stored in HDFS. Let M be the number of Mappers and m be the number of containers allotted to this job. As $M \gg m$ there will be M/m number of rounds of map operations executed for each container. In each iteration of map operation input images are divided into $p \times q$ grids and for each grid LBP histograms are computed using Eq. (1). Let $\{\mu_1, \mu_2, \dots, \mu_M\}$ denotes a set of Mappers, the output of Mapper function is

$$O_m = \bigcup_{(k_i, v_i) \in i_1, i_2 \dots i_n} \mu_m(k_i, v_i) \quad (2)$$

where μ_m is the user defined mapper function operates on (k_i, v_i) pairs and generates output results O_m . In this case μ_m is defined to perform computation of LBP on matrix of pixel values. The output of Mapper is again a (k_i, v_i) pair, with k_i is the file name and value v_i is array of bytes representing LBP values computed for each pixel.

The LBP values generated for each grid are stored with $(imageID, gridID, LBP\_value)$ as the key and 1 as the value into HDFS. At the Reducer stage, the instances having common keys are grouped together and added to produce final output as LBP histogram. The final LBP histogram output gener-

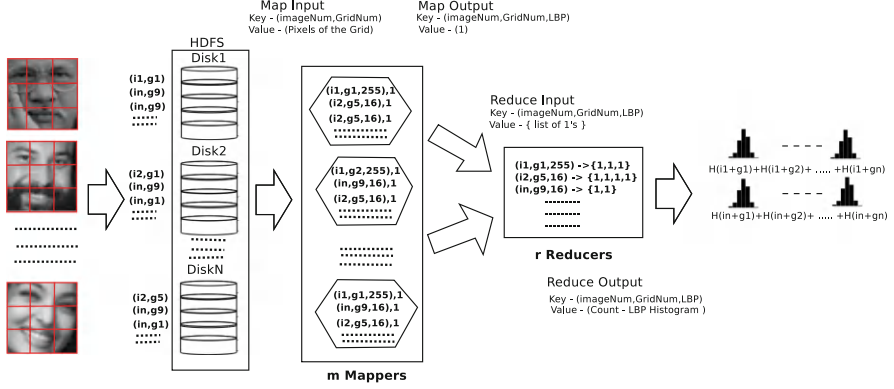


Fig. 2. The algorithm framework of computing MapReduce based LBP by using Mapper and a Reducer. The splitting of image into grids and gridwise computation of LBP takes place in the Mapper stage and reductions in the form of computing histograms is performed at the reducer stage.

ated from Reducer is stored into HDFS in the form of $(key, value)$ pairs with $(imageID, gridID, LBP\_value)$ as key and LBP_count as value.

Let (I, G) denotes the collection of input image grids with image id i and grid id g

$$(I, G) = \{(i_1, g_1), \dots, (i_2, g_5), \dots, (i_n, g_9)\} \quad (3)$$

The MapReduce workflow involves Mapper stage followed by Reducer, the intermediate output O_r from the first stage is given by

$$O_r = \bigcup_{(k_i, v_i) \in (I, G)} \mu_r(k_i, v_i) \quad (4)$$

In this case, μ_r is defined to read array of bytes for each image grid from HDFS and compute LBP for each pixel of the grid.

The intermediate output generated from mapper O_r will be shuffled and sorted based on keys $\{k_i\}$. The Reducer stage consisting of r reducers receive key as $k_i = (i_n, g_n, LBP\_value)$ and value as list of $\{1\}$ belonging to the key k_i . The algorithmic details of computing LBP feature extraction using MapReduce programming model is depicted in Algorithm 1.

With M Map processes, m containers, Mapper stage may require M/m rounds of map operations. Similarly with R Reduce processes and r allotted containers, reduce stage may require R/r rounds of reduce operations for completing all the reduce tasks. The computational cost for MRLBP using Mapper and Reducer stages is given by

$$T = \phi_{map} + \phi_{red} + \delta(map, red) \quad (5)$$

where δ is the computational cost for scheduling and management of jobs and, ϕ_{map} and ϕ_{red} are the computational costs for Mapper and Reducer phases.

4 Designing RDD Based LBP Feature Extraction

The basic building units for Spark programming model are RDDs where all input data, intermediate results and outputs are represented by RDDs. Each transformation or reduction operation on RDDs results into a new immutable RDDs. Spark employs lineage-graphs for recovering missing partitions during run-time failures. The lineage graphs are capable of keeping track of sequence of transformations applied on RDDs. Typically, in a production environment input images are stored in the HDFS or any persistent distributed file system. Reading input images from HDFS creates an RDD which is distributed to multiple worker nodes. The series of transformation functions applied on RDDs are mapped to these worker nodes and performs execution on their local partitions.

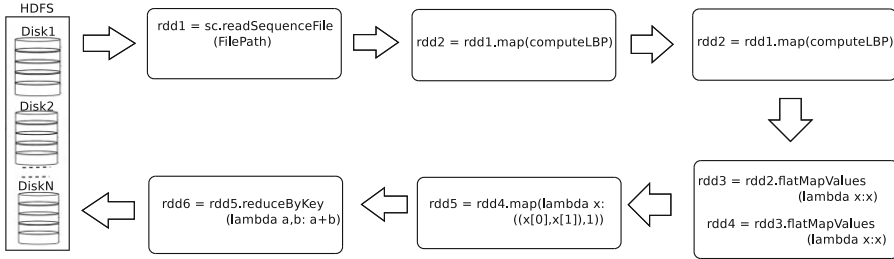


Fig. 3. LBP feature extraction on Apache spark platform using RDDs where intermediate results are stored in the RAM.

Figure 3 shows the Spark RDD based design flow of computing LBP histograms. Spark based RDDLBP algorithm involves a series of transformation operations followed by reduction operations. In each of the transformation or reduction operations, the intermediate results generated in the form of RDDs are stored and retrieved directly from RAM. In the below paragraphs we explain the sequence of steps used in computing LBP histograms on Apache Spark framework.

step 1: `rdd1 = sc.readSequenceFile(FilePath)`. It is the first step in Spark based applications where context is established with the master and worker nodes using `sc` object. The spark context object `sc` is used to read images from Hadoop binary sequence file stored in HDFS. Reading images from HDFS creates a new RDD comprising all image grids in the form of $(key, value)$ pairs.

Algorithm 1. MRLBP using Mapper and Reducer

Map - Compute LBP for each grid

```

1: function MAP(ImageID + GridID, ByteArray)
2:   Read input grids
3:   Scan each pixel
4:    $LBP = 0$ 
5:   for  $x = 0 \rightarrow \text{rows}$  do
6:     for  $y = 0 \rightarrow \text{columns}$  do
7:       for  $p = 0 \rightarrow 7$  do
8:          $\text{diff} = \text{grid}(x, y) - \text{grid}(p)$ 
9:         if  $\text{diff} \geq 0$  then
10:           $\text{flag} = 0$ 
11:        else
12:           $\text{flag} = 1$ 
13:        end if
14:         $LBP = LBP + \text{flag} \times 2^p$ 
15:      end for
16:    end for
17:  end for
18:   $\text{emits} \rightarrow (\text{ImageID} + \text{GridID} + LBP, 1)$ 
19: end function

Reduce - Compute Histogram of LBP
20: function REDUCE(ImageID + GridID + LBP,  $\text{list}\{1, 1, \dots\}$ )
21:    $\text{sum} = 0$ 
22:   for  $k \in \{1, 1, 1, \dots\}$  do
23:      $\text{sum} = \text{sum} + k$ 
24:   end for
25:    $\text{emits} \rightarrow (\text{ImageID} + \text{GridID} + LBP, \text{sum})$ 
26: end function

```

step 2: $\text{rdd2} = \text{rdd1.map}(\text{computeLBP})$. The image partitions are distributed across multiple nodes of the cluster. A new function *computeLBP* that computes LBP values for each pixel of the partitions is applied on *rdd1*. The *computeLBP* function reads image partitions with key as *imageID* and value as *ByteArray*. The *computeLBP()* transformation on *rdd1* creates a new RDD *rdd2* comprising LBP values associated with respective *keys*. The following steps are used to compute LBP histograms by applying a sequence of new transformations followed by reduction operations.

step 3: $\text{rdd3} = \text{rdd2.flatMapValues}(\text{lambda } x:x)$. The new RDD *rdd2* consists of partitions representing $[(key, 2D - array)]$ with key as partition id and value as 2D LBP array. In order to compute LBP histogram, we first split each partition having $[(key, 2D - array)]$ into $[(key, 1D - array)]$ by applying a transformation *flatMapValues* on *rdd2* creating a new RDD *rdd3*.

step 4: `rdd4 = rdd3.flatMapValues(lambda x:x)`. We split RDD `rdd3` comprising $[(key, 1D - array)]$ in to $[(key, PixelWiseLBP)]$ by applying one more `flatMapValues` on `rdd3` and creating a new RDD object `rdd4`. The new RDD `rdd4` comprises `key` as $(imageID, gridID)$ and `value` as pixel level LBP values `PixelWiseLBP`.

step 5: `rdd5 = rdd4.map(lambda x:((x[0], x[1]), 1))`. We assign 1 for each partition having $[(key, PixelWiseLBP)]$ using `map` function and creating a new RDD having partitions $[(key, PixelWiseLBP), 1]$ with key representing $(key, PixelWiseLBP)$ and value as 1.

step 6: `rdd6 = rdd5.reduceByKey(lambda a, b: a + b)`. In this final step, we compute LBP histogram by summing the values of all the partitions having similar keys using `reduceByKey` function. The reduction operations applied on `rdd5` creates a new RDD with partitions representing $[(key, PixelWiseLBP), LBPValueCount]$. The final RDD `rdd6` created may be sorted based on keys and stored back in to HDFS in the desired file format.

In RDDLBP algorithm, the sequence of transformations comprising `readSequenceFile`, `map` and `flatMap` functions operate on their local partitions independent of remaining data residing in other worker nodes. Unlike MRLBP implemented in Hadoop, shuffling of data in RDDLBP is needed only during `reduceByKey` or `aggregateByKey` operations. Further, RDDLBP operation is optimized by performing reductions on local data at worker nodes before final reduction at driver node. Finally, the reduced data from worker node is shuffled using Hash partitioner before it is pulled by the driver node. Since most of the reduction operations happen at the worker nodes before transferring it to the master, a significant reduction in network traffic time can be achieved in RDD based Spark applications.

5 Datasets and Experimental Results

5.1 LFW Dataset

The Labeled Faces in the Wild (LFW) dataset is a prominent resource in the field of facial recognition research, comprising 13,233 images of 5,749 distinct individuals. Among these, 1,680 individuals have multiple images, facilitating a comprehensive evaluation of facial recognition systems under various conditions. The dataset was designed to address the challenges associated with unconstrained face recognition, as the images exhibit a wide range of variations in pose, lighting, and facial expressions.

Collected from the internet, each image in the LFW dataset is annotated with the corresponding individual's name. The faces were initially detected using the Viola-Jones face detector, ensuring a consistent approach to face localization. This diversity in image characteristics makes the LFW dataset a robust benchmark for testing the accuracy and generalization capabilities of facial recognition algorithms.



Fig. 4. Sample LFW images

Sample LFW images are shown in Fig. 4. For the analysis of the computational capabilities of a PySpark and Hadoop systems, we have created a larger set of images by duplicating the original images in the LFW dataset. This augmented dataset allows for extensive testing and validation of the model's performance and scalability in handling large-scale facial recognition tasks. The LBP feature extraction and its histogram creation is depicted in Fig. 5.

5.2 Experimental Set Up

The Hadoop-Yarn cluster environment is set up on Amazon EC2 cloud with m4.4 xlarge instances having 16 vCPUs, 128 GB RAM, EBS SSD storage with Ubuntu 16.04 LTS as the operating system. Further, we have built our own in-house Hadoop cluster with a size of 100 nodes. We have installed Hadoop with version 2.7.3 and Apache Spark with version 2.0 on top of Hadoop. We used Java programming for MapReduce based LBP and Python 3.x for RDDLBP implementations.

We conducted experiments of running MRLBP and RDDLBP by varying number of nodes and size of dataset. We have used replication factor of 3 and optimization parameters are configured for the YARN cluster according to [20]. Table 1 shows comparison between computational costs of MRLBP and RDDLBP algorithms. Since MapReduce based programs involve frequent access to HDFS, MRLBP requires more execution time as compared to Apache Spark based RDDLBP algorithms. Due to in-memory computations and optimizations made in the map and reduce stages, RDD based LBP yields better computational performance. The major bottleneck for deploying MRLBP and RDDLBP algorithms on distributed platforms is the factors contributed by network traffic and scheduling overhead. Further, we analyzed the effect of increase in volume of intermediate results in case of RDDLBP, by increasing dataset size and increasing image partitioning scale. We noticed the fact that, when the volume of intermediate results increases RDDLBP often fails to improve computational speed.

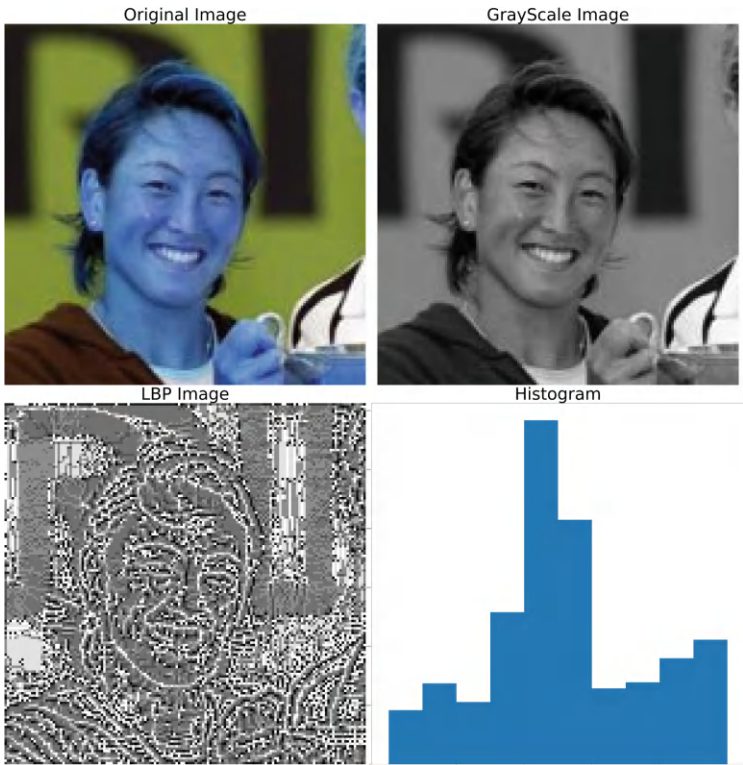


Fig. 5. LBP feature extraction for a sample LFW face image

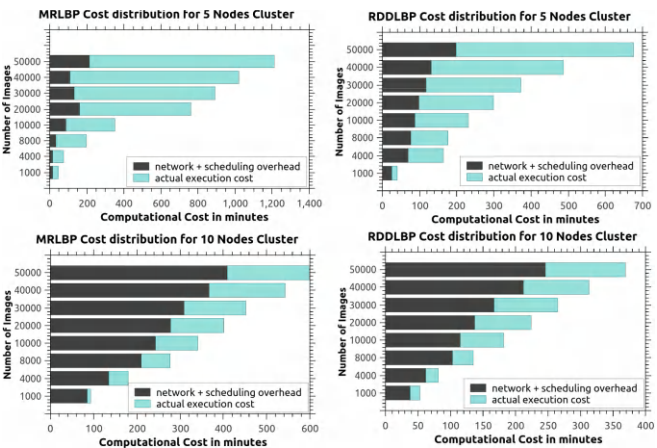


Fig. 6. Graphs shows influence of (*network + scheduling*) overhead in increasing overall computational cost of MRLBP and RDDLP algorithms.

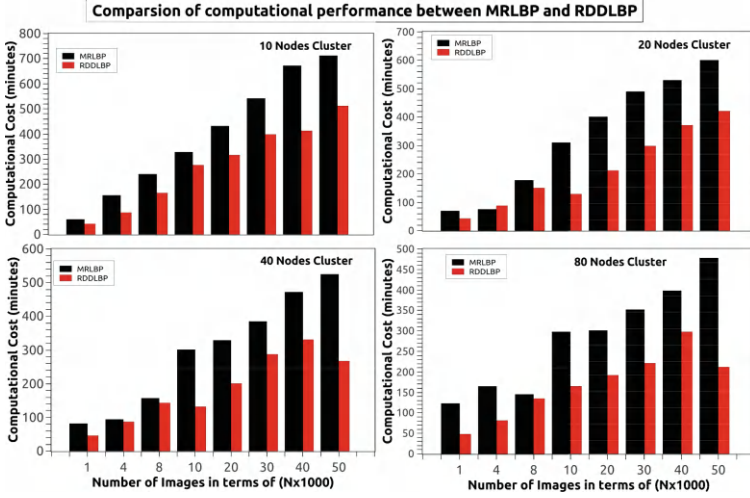


Fig. 7. Comparison of computational costs of running MRLBP and RDDLP on Hadoop Yarn cluster. The two charts either in the left part or right part have common X-axis.

Table 2 gives the comparison between computational performance of RDDLP and MRLBP when the volume of intermediate results is very large.

Figure 7 shows comparison of computational performance between MRLBP and RDDLP algorithms for the in-house Hadoop cluster. Due to in-memory computing and RDD based distributed computation, implementing LBP algorithms using Spark outperforms MapReduce based designs. An improvement of 58.28% speed up is achieved by using RDDLP algorithm in comparison with MRLBP algorithms for 80 nodes cluster. The influence of network and scheduling cost on overall cost of MRLBP and RDDLP is illustrated in Fig. 6.

6 Conclusion

In this work, we presented designing RDDLP algorithm for extracting texture features from LFW images on distributed platforms. We compared RDD based LBP feature extraction with MapReduce implementation on Yarn-Hadoop cluster. The experimental results prove that Spark based RDDLP yields around 58.28% times speed up compared to MRLBP implemented on Hadoop framework. However, avoiding overflow of large volume of intermediate results and spilling it to HDFS in Apache Spark still remained as challenging problem. In future, the novel idea of designing RDD based algorithms can be extended to implement scalable machine learning and deep learning applications.

Table 1. Computational Performance of MRLBP and RDDLBP when volume of intermediate results exceeds size of RAM on a 5-nodes cluster. The evaluations are conducted on cloud environment.

Image Size	MRLBP cost	RDDLBP cost
30 GB	52311 s	51397 s
40 GB	87423 s	86152 s
50 GB	130921 s	129878 s

References

1. Singh, D., Reddy, C.K.: A survey on platforms for big data analytics. *J. Big Data* **2**(8) (2015)
2. Christophe, E., Michel, J., Inglada, J.: Remote sensing processing: from multicore to GPU. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **4**(3) (2011)
3. Park, I.K., Singhal, N., Lee, M.H., Cho, S., Kim, C.: Design and performance evaluation of image processing algorithms on GPUs. *IEEE Trans. Parallel Distrib. Syst.* **22**(1) (2011)
4. Lopez, M.B.: Evaluation of real-time LBP computing in multiple architectures. *J. Real-Time Image Process.* **13**(2) (2017)
5. Song, C., Li, Y., Huang, B.: A GPU-accelerated wavelet decompression system with SPIHT and Reed-Solomon decoding for satellite images. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **4**(3) (2011)
6. Lee, C.A., Gasster, S.D.: Recent developments in high performance computing for remote sensing: a review. *Recent Dev. High Perform. Comput. Remote Sens. Rev.* **4**(3) (2011)
7. Cresson, R., Hauteux, G.: A generic framework for the development of geospatial processing pipelines on clusters. *IEEE Geosci. Remote Sens. Lett.* **13**(11) (2016)
8. Plaza, A., Du, Q., Chang, Y.-L., King, R.L.: High performance computing for hyperspectral remote sensing. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **4**(3) (2011)
9. Buyya, R., Venugopal, S.: A gentle introduction to grid computing and technologies. *CSI Commun.* **29**(1), 9–19 (2005)
10. Beaumont, O., Boudet, V., Petitet, A., Rastello, F., Robert, Y.: A proposal for a heterogeneous cluster scalapack (dense linear solvers). *IEEE Trans. Comput.* **50**(10), 1052–1070 (2001)
11. Happ, P.N., da Costa, G.A.O.P., Bentes, C., Feitosa, R.Q., da Silva Ferreira, R., Farias, R.: A cloud computing strategy for region-growing segmentation. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **9**(12) (2016)
12. Huang, J., Zhang, R., Buyya, R., Chen, J., Wu, Y.: Heads-join: efficient earth mover’s distance similarity joins on hadoop. *IEEE Trans. Parallel Distrib. Syst.* **27**(6) (2016)
13. Huang, W.: *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **10**(1) (2016)
14. Huang, W., Meng, L., Zhang, D., Zhang, W.: Cloud computing for earth surface deformation analysis via spaceborne radar imaging: a case study. *IEEE Trans. Cloud Comput.* **4**(1) (2016)
15. Wang, L., He, D.: Texture classification using texture spectrum. *Pattern Recogn.* **23**(8) (1990)

16. Ojala, M.P.T., Harwood, D.: Performance evaluation of texture measures with classification based on Kullback discrimination of distributions. In: Proceedings of the 12th IAPR International Conference on Pattern Recognition (ICPR 1994), vol. 1, no. 582 (1994)
17. Huang, G.B., Ramesh, M., Berg, T., Learned-Miller, E.: Labeled faces in the wild: a database for studying face recognition in unconstrained environments. University of Massachusetts, Amherst, Technical report 07-49 (2007)
18. Dean, J., Ghemawat, S.: Mapreduce: simplified data processing on large clusters. *Commun. ACM* **51**(1), 107–113 (2008)
19. Hadoop distributed file system. <http://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-hdfs/HdfsUserGuide.html>. Accessed 05 Mar 2017
20. Yet another resource negotiator. https://docs.hortonworks.com/HDPDocuments/HDP2/HDP-2.6.1/bk_command-line-installation/content/ch_setting_up_hadoop_configuration_chapter.html. Accessed 10 Jan 2024



Sentiment Analysis of Political Party Unveiling Insights from Big Data

Mayuri Bapat¹(✉), Shreya Karade¹, and Aniket S. Nagane²

¹ Department of Computer Science, Dr. Vishwanath Karad MIT World Peace University, Pune, India

mmbapat@mitacsc.ac.in

² Symbiosis Institute of Computer Studies and Research (SICSR), Symbiosis International (Deemed University), Pune, India

Abstract. Sentiment analysis is critical for analyzing public sentiment, especially in the context of political discourse on social media platforms such as Twitter. This study delves into the intricacies of sentiment analysis methodologies within political conversations, employing diverse techniques to draw insights from large datasets. The study covers data preprocessing, encompassing handling missing values, text preprocessing techniques, and data splitting for robust model training and evaluation. Key phases include TF-IDF vectorization for numerical feature representation, label encoding for categorical variables, tokenization for text normalization, and sentiment classification using sentiment analysis tools like VADER and SentimentIntensityAnalyzer from NLTK. The approach combines Min-Max scaling for data standardization, logistic regression for sentiment prediction, and rigorous correlation analysis to determine the links between variables. The results show that the approach is successful, with logistic regression obtaining an impressive 92% accuracy in sentiment identification. The sentiment split displays the percentage of positive tweets (23.20%), negative tweets (7.33%), and neutral tweets (69.46%). Furthermore, a comparison with an alternate dataset using logistic regression revealed a 59% difference in accuracy. The sentiment distribution in this dataset reveals 59.03% of positive tweets, 18.66% of negative tweets, and 22.32% of neutral tweets. This demonstrates whether model performance varies depending on the dataset used. These findings highlight the significance of dataset selection and model assessment in sentiment analysis, demonstrating how various datasets can result in variable accuracy levels and sentiment distributions. Our research focuses on sentiment analysis of huge data from political parties to discover positive, negative, and neutral tweets.

Keywords: Sentiment Analysis · Twitter · Sentiment Analysis Tools (VADER · SentimentIntensityAnalyzer) · TF-IDF Vectorization · Sentiment Distribution · SentimentIntensityAnalyzer · Sentiment Prediction

1 Introduction

The integration of social media and politics in the current era has become a critical platform for evaluating public sentiment and understanding political discourse. With the advent of big data technology and advanced analytics, researchers can delve deeper into the sentiments expressed in political conversations on platforms such as Twitter. This study focuses on sentiment analysis applied to political party speech, aiming to uncover significant insights from large datasets using advanced analytical techniques. The evolution of social media platforms has revolutionized political communication, enabling real-time interactions among politicians, parties, and voters. In this context, sentiment analysis emerges as an effective method for discerning the underlying emotions, attitudes, and perceptions embedded in textual data such as tweets and posts. By combining sentiment analysis with big data analytics, researchers can identify patterns, trends, and sentiment shifts that are crucial for evaluating public opinion and understanding the public's perception of political entities. Our research encompasses comprehensive data preprocessing, starting with handling missing values and null entries, followed by text processing and cleaning. The data is then partitioned into training, testing, and evaluation sets to ensure reliable model training. TF-IDF vectorization is employed to convert textual input into numerical representations, a critical step in preparing data for machine learning (ML) algorithms. We utilize the VADER sentiment analysis tool, specifically designed for social media language, to calculate sentiment scores for each comment. Min-Max scaling normalization ensures a standardized scale for analysis. Correlation analysis reveals the relationships between variables, providing essential insights into the dataset's structure. Utilizing logistic regression as our ML technique, we achieve an impressive 92% accuracy, demonstrating the model's efficacy in sentiment prediction. Visual representations using best-fit line plots facilitate the assessment of the model's performance across various conditions. The sentiment analysis results offer valuable information on the distribution of tweets across positive, negative, and neutral sentiment classes. Additionally, the use of the `SentimentIntensityAnalyzer` from the NLTK library enhances our analysis, leading to a more comprehensive understanding of sentiment classes within the dataset. Through this research, we aim to provide valuable insights into sentiment analysis applied to political discourse, elucidating the complex nature of public sentiment, opinion formation, and sentiment expression in the context of political party debates on social media.

2 Database

We perform sentiment analysis on two datasets: one obtained from Kaggle with 66,664 records from Twitter related to a political party, containing columns `Datetime`, `Tweet Id`, `Text`, `Username`, and `like Count`; and another dataset we created with 6,663 records, we compute the accuracy of the sentiment analysis, compare the percentage of sentiments between the two datasets, and ascertain

technological era, which is a crucial shift. Political parties use big data analytics to learn about their people's vote preferences based on socioeconomic position, caste, local concerns, and other factors. Data analysis from multiple apps provides insights into MPs' and MLAs' political experiences while also evaluating their performance. We suggest collecting the data using Twitter APIs and then using data mining and sentiment analysis algorithms to the collected data [3]. Using the lexicon-based sentiment analysis with Python and Natural Language Processing (NLP). Real-time data is gathered using the streaming application programming interface (API). Twitter data is based on certain keywords. The retrieved tweet data set is then transferred to text pre-processing, followed by text categorization into subjectivity and polarity as part of sentiment analysis. Before the final analysis, the data was cleaned to eliminate any unnecessary information such as URLs, @ users, digits, punctuation, stop words, white spaces, and so on, resulting in more accurate and relevant results from the dataset. Accurately anticipating these signs might help allocate limited resources, improving the efficiency of the Political Campaign on "Twitter [4]. According to the Election Commission, 1.9 crore voters are eligible to vote in August 2016 in Punjab. The collection of data acquired via Twython was evaluated to identify trivial and interesting patterns in the data. The essential purpose of this investigation project is to conduct sentiment analysis" for political parties including the BJP, INC, and AAP on the results of the 2017 Punjab Legislative Assembly election. We evaluated the whole election process and derived major conclusions. Punjabi Non-resident Indians (NRIs) have played an important part in elections. Opinion mining is an alternative term for sentiment analysis since it extracts the main features from the opinions of others. Sentimental Analysis uses many ML algorithms NLP, and statistical "models to discover the properties of huge quantities of data. We carried out a sentiment analysis of numerous political parties and studied the results, discovering that AAP had higher polarity, meaning that" tweets conveyed more emotions (either positive or negative). More polarity can be observed in positive evaluations than in negative ones. The findings show that in comparison with the other 2 parties, AAP has more negative tweets [5]. In this study, the rise in popularity of social networks such as Facebook, Twitter, and Instagram altered the way politicians engage with voters and conduct elections. They also pushed for a machine learning strategy that used regression techniques like Linear Regression. Because of SM's inherent characteristics, like the huge quantity of available data that can be accessed in real-time, the focus is on employing SM data to forecast election outcomes. The main conclusions are that while other approaches, like regression models developed from traditional polls, yield better results, the most popular methodology—volume as well as sentiment analysis on Twitter—is ineffective. Modelling, as well as evaluation, are also covered, as is the necessity for additional research into the use of machine learning technologies. As a result, SM data can forecast election outcomes [6]. In this research sentiment analysis after a careful process of data collection, pre-processing, text cleaning, tokenization, label encoding, missing value handling, and correlation analysis are used. Using lexicon-based approaches and advanced fea-

ture engineering techniques. Past investigations suggest “that social media could be utilized to fore-cast political elections. Personal traits have also been” cited as key elements affecting election results, and candidate attributes have been used to forecast elections. To fill a research vacuum, this study examines the function of 2 separate social media sites Twitter as well as Facebook in election gathering and preprocessing. This study contributes to the issue of how social media might forecast election outcomes [7]. Natural language processing is used in this research to analyze political attitudes. These evaluations may be utilized for a variety of objectives, including understanding public opinion and trends, particularly around election time. This study investigates sentiment analysis as it relates to political party discourse. Context sentiment analysis is a useful tool for understanding the underlying emotions, attitudes, and views buried in textual data like tweets and postings. Sentiment analysis. Analysis of sentiment is a strong approach to understanding the overall Feeling of political groupings. This study aims to predict outcomes based on percentages [8]. This research focuses on the real-time and open personality of political conversation, especially during the election season. on documenting current social patterns and projecting vote outcomes in a given location. These social networking sites produce a huge volume of raw data which may be utilized to make electoral predictions. This massive amount of data is referred to as “Big data”. This paper attempts to cover the many methods, models, and algorithms for discussing, predicting, and observing campaigns and investigates the influence of social media on the country’s political scene [9]. Using particular information from a political investments dataset, the study applies Numerous machine learning techniques to anticipate developing countries and costs. An insight from big data analytics two of the four ensembles developed are upgraded. Raising and building ensembles beat bagging in terms of accuracy. It looks at how technologies like WhatsApp, Twitter, and Facebook are altering political communication. This efficiency might increase income for developing countries. ML models excel at managing huge datasets [10].

4 Proposed Approach

Our paper presented a method for sentiment analysis of comments from Twitter related to a political party. For this approach, we use a lexical analysis approach where Text Blob calculates the sentiment polarity of each cleaned text and classify it according to the sentiment polarity score, as favourable, unfavourable, or neutral. The text input is then converted into numerical features using TF-IDF vectorization. Following that, we perform sentiment analysis using VADER and NLTK’s SentimentIntensityAnalyzer to generate sentiment scores. Following that, we use the logistic regression machine learning technique, which is the best-fit model and achieves the best accuracy. Finally, we compute the count for each sentiment class. We use the NLP library to do a study of the cleaned text’s sentiment. Text Blob determines the feeling polarity for each text item, which is a sentiment score on a scale of 1 (most positive) to -1 (most negative). We

categorize sentiment polarity into three categories: neutral, adverse, and favorable. This is completed by applying conditions on the sentiment polarity score obtained by Text Blob. If the score exceeds zero, the sentiment is identified as positive; if it is less than zero, it is labeled as negative; otherwise it is classified as neutral. Algorithm for the same is given below:

- 1. Start
- 2. Load the dataset into DataFrame df with columns
- 3. Convert the Text Column to String Type to handle float values gracefully: It is equal to `df['Text'].astype(str)`
- 4. Preprocess the text data
for each text in `df['Text']`:
`cleaned_text=remove_non_alphanumeric(text)`
`df['Text_Cleaned'].append(cleaned_text)`
- 5. Perform Sentiment Analysis: 6. Classify Sentiment: if `sentiment_score>0`:
`sentiment_class='Positive'`
if `sentiment_score<0`:
`sentiment_class='Negative'`
else: `sentiment_class='Neutral'`
- 7. Output the updated DataFrame with Sentiment Analysis Result 8. Stop

5 Methodology

This section outlines the systematic approach undertaken to investigate the research questions and achieve the objectives of this study and shown in Fig. 2.

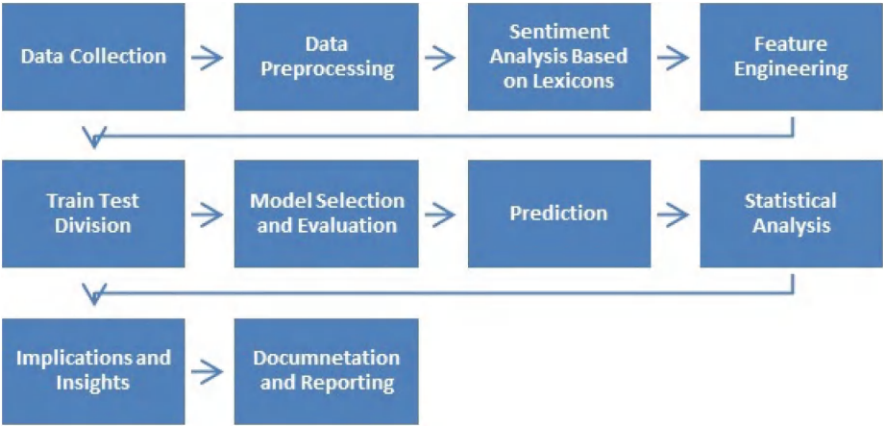


Fig. 2. Methodology.

5.1 Data Collection

We utilized Kaggle to collect a comprehensive dataset of tweets related to political parties, and additionally generated synthetic data to facilitate a comparative analysis of political sentiments.

5.2 Data Pre-processing

The pre-processing phase involved the removal of missing and null values from critical columns such as Tweet ID, Text, Username, and Like Count. We employed the Natural Language Toolkit (NLTK) for tokenization and stop word removal. The text pre-processing steps included converting text to lowercase, removing punctuation, tokenizing the text, eliminating stop words, and reconstructing the tokens into a clean string. Label Encoding was applied to transform categorical data into numerical values, ensuring compatibility with machine learning algorithms.

5.3 Sentiment Analysis Based on Lexicons

Lexicon-based sentiment analysis algorithms were utilized to classify tweets as positive, neutral, or negative based on predefined sentiment lexicons. This approach leveraged the semantic orientation of words to determine the overall sentiment of each tweet.

5.4 Feature Engineering

We explored interactions between features using a correlation matrix to identify significant relationships. Sentiment scores were developed to enhance the accuracy of the sentiment analysis process. This step was crucial for creating meaningful features that contribute to the predictive power of the models.

5.5 Dataset Division

The dataset was split into training and testing sets, with 80% allocated for training and 20% for evaluation. The training set included features extracted from the 'Datetime' column, such as year, month, day, hour, and minute, along with Tweet ID, represented as `X_train`. The 'Sentiment' column served as the target variable for training the models. The testing set, represented as `X_test` and `y_test`, contained similar features but from a different portion of the dataset, ensuring that the models were evaluated on unseen data.

5.6 Model Selection and Evaluation

We implemented machine learning algorithms, specifically Linear and Logistic Regression, for sentiment analysis. The models underwent rigorous training and cross-validation procedures to validate their performance. Logistic Regression emerged as the best-performing model, achieving an impressive accuracy of 92%.

5.7 Prediction

Following the training and validation phases, the models were employed to predict sentiment labels (positive, neutral, or negative) for new, previously unseen data. This prediction capability allows the application of sentiment analysis to real-time or future Twitter data, providing valuable insights into public sentiment dynamics.

5.8 Rationale for Choosing VADER and SentimentIntensityAnalyzer as Sentiment Analysis Tools

In this study, VADER (Valence Aware Dictionary and Sentiment Reasoner) and SentimentIntensityAnalyzer from the NLTK (Natural Language Toolkit) library were chosen as the primary sentiment analysis tools due to their effectiveness and reliability in handling social media text. VADER is specifically designed to analyze sentiments expressed in social media platforms, making it particularly well-suited for the analysis of political discourse on Twitter. It combines a sentiment lexicon with rules that account for the intensity of sentiments, enabling it to effectively parse the nuanced expressions often found in social media posts. VADER's lexicon includes common slang, emoticons, acronyms, and other informal language typically used on social media, which enhances its ability to accurately classify the sentiment of tweets.

The SentimentIntensityAnalyzer, which implements VADER, provides a comprehensive analysis by outputting scores for positive, negative, neutral, and compound sentiments. This allows for a more granular understanding of the sentiments within the text. Its rule-based model performs well with minimal computational resources, making it efficient for real-time applications and large-scale data processing. One of the key strengths of VADER is its ability to handle both polarity (positive/negative/neutral) and intensity of sentiment, making it highly sensitive to the subtleties of social media language. This includes the use of capitalization, punctuation, and degree modifiers, which can significantly alter the sentiment conveyed in a tweet. Additionally, VADER's ease of use and integration with the NLTK library makes it accessible for quick implementation and evaluation. However, VADER and SentimentIntensityAnalyzer also have limitations. Being a rule-based system, VADER may struggle with more complex sentences where contextual understanding is crucial. It may not perform as well on texts that require deep semantic understanding or cultural context beyond its pre-defined lexicon and rules. Moreover, while VADER can handle informal language, it might not accurately interpret all slang or emerging social media trends, which are constantly evolving.

Despite these limitations, VADER and SentimentIntensityAnalyzer offer a robust and efficient approach for sentiment analysis in social media contexts, providing a strong rationale for their use in this study. Their strengths in handling informal language, ease of implementation, and efficiency in processing large datasets make them invaluable tools for analyzing the sentiments expressed

in political discourse on Twitter. This comparison reveals significant differences in the sentiment distribution captured by these two sentiment analysis tools, showcasing the nuances and variations in sentiment interpretation.

5.9 Statistical Analysis

To assess the significance of sentiment differences between datasets, various statistical tests may be employed. These tests help determine whether observed differences are statistically significant and not due to random variation. Common tests include t-tests, chi-square tests, and ANOVA, depending on the data distribution and the nature of the comparisons. These analyses are crucial for validating the models' prediction capabilities and deriving meaningful conclusions about the underlying data patterns.

5.10 Implications and Insights

The results from sentiment analysis, prediction models, and dataset comparisons are carefully interpreted to extract actionable insights. These insights provide a deeper understanding of public sentiment toward the Political Party. The findings have significant implications for political perception and communication strategies, guiding how political messages are crafted and delivered to the public. By understanding sentiment trends, political entities can refine their approaches to resonate more effectively with their audience.

5.11 Documentation and Reporting

Comprehensive documentation of the entire process is essential. This includes recording the parameters used for model development, the evaluation metrics employed, and the data pre-processing steps undertaken. Summarizing the findings and insights derived from the analysis is crucial for transparency and reproducibility. Additionally, creating visualizations such as charts, graphs, and dashboards can significantly enhance the presentation of results, making the data more accessible and understandable for stakeholders. This thorough documentation ensures that the methodology and conclusions can be reviewed and validated by others in the field.

6 Result

The results of the sentiment analysis conducted using VADER and SentimentIntensityAnalyzer are illustrated in Fig. 3. This figure highlights significant disparities in the distribution of sentiment classes within political conversations on Twitter. Specifically, the analysis reveals notable differences in how positive, negative, and neutral sentiments are expressed across various political topics and discussions. These disparities indicate that public sentiment towards political parties and issues is not uniform and can vary widely depending on the context

of the conversation. This insight is crucial for understanding the dynamics of political discourse on social media and can inform strategies for political communication and engagement. This mismatch implies that the sentiment analysis technique used has a considerable influence on how sentiment is perceived in the dataset. The sentiment obtained for the specified row ID. We compare the sentiment obtained from your analysis with the sentiment data in your CSV file as a result both matches.

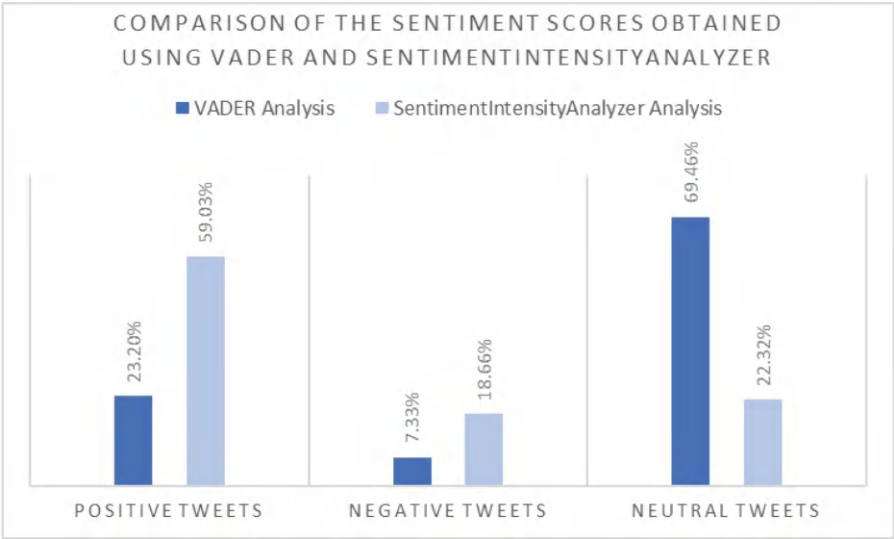


Fig. 3. comparison of the sentiment scores obtained using VADER and SentimentIntensityAnalyzer.

7 Conclusion and Future Enhancement

Social media are being utilized by more people to express their ideas. Twitter is an excellent resource for researching public opinion, especially around elections. Election results can be understood by examining the sentiment and data on Indian political Twitter. In this research, we developed a Lexical Analysis approach to detect positive, negative, and neutral comments using sentiment analysis, employing SentimentIntensityAnalyzer and VADER.

Future research in the area of sentiment analysis of political discourse on social media should focus on enhancing analytical techniques, expanding the scope of data sources, and addressing ethical considerations. By incorporating advanced NLP techniques, real-time processing, cross-platform analysis, and integrating external data sources, researchers can develop more accurate and comprehensive sentiment analysis models. These enhancements will provide

deeper insights into public sentiment, aiding political analysts, policymakers, and researchers in understanding and responding to the dynamics of political discourse.

References

1. Romanyshyn, M.: Rule-based sentiment analysis of Ukrainian reviews. *Int. J. Artif. Intell. Appl.* **4**(4), 103 (2013)
2. Alvi, Q., Farooq, S., Ahmed, B., Khan, N., Awan, M., Nobanee, H.: On the frontiers of Twitter data and sentiment analysis in election prediction: a review. *PeerJ Comput. Sci.* **9**, e1517 (2023). <https://doi.org/10.7717/peerj-cs.1517>
3. Ramteke, J., Shah, S., Godhia, D., Shaikh, A.: Election result prediction using Twitter sentiment analysis. In: 2016 International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India, pp. 1–5 (2016). <https://doi.org/10.1109/INVENTIVE.2016.7823280>
4. Kaur, M., Verma, R., Ranjan, S.: Location-Based Sentiment Analysis of Indian Political Campaign on Twitter (2022). <https://doi.org/10.9756/INT-JECSE/V14S3.3>
5. Singh, A., Gupta, D., Singh, R.: Sentiment analysis of Twitter user data on Punjab legislative assembly election. *Int. J. Mod. Educ. Comput. Sci.* **9**, 60–68 (2017). <https://doi.org/10.5815/ijmecs.2017.09.07>
6. Brito, K., Filho, R., Adeodato, P.: A systematic review of predicting elections based on social media data: research challenges and future directions. *IEEE Trans. Comput. Soc. Syst.* **8**, 819–843 (2021). <https://doi.org/10.1109/TCSS.2021.3063660>
7. Vepsäläinen, T., Li, H., Suomi, R.: The role of social media platforms in forecasting elections: a comparison of Twitter and Facebook. *Digit. Gov. Res. Pract.* (2024). <https://doi.org/10.1145/3651227>
8. Political Marketing. <https://politicalmarketer.com/political-sentiment-analysis/>. Accessed 28 July 2021
9. Koli, A., Ahmed, M.: Election prediction using big data analytics-a survey. *Int. J. Eng. Technol. (UAE)* **7**, 366–369 (2018). <https://doi.org/10.14419/ijet.v7i4.5.20108>
10. Ogbuokiri, B.: Social media impact on the politics of developing countries: an insight from big data analytics. *Transylvanian Rev.* **XXIV**(5), 431–448 (2016)

AI for Defense and Information Security



Securing Web Applications with Improved Multifactor Authentication Using Browser-Based Geolocation

A. Yadav¹, K. Kumar¹(✉), P. Pandey², M. Khari¹, and P. P. Singh¹

¹ School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi 110067, India

kapil.sharma0942211@gmail.com, piyushpsingh@mail.jnu.ac.in

² Graphic Era University, Dehradun, India

Abstract. In the current fast-paced technology-driven cyber world, security is one of the major key challenges. Unauthorized access and identity theft is one of the security concerns and a good authentication mechanism is required to protect from these vulnerabilities. Basic Single Factor Authentication (SFA), username and password, can be easily cracked by various attacks like Dictionary Attack, and Bruteforce Attack. Hence multiple factors are utilized for enhancing security known as MultiFactor Authentication (MFA). It comprises three factors such as Knowledge Factor (like PIN, Password), Possession Factor (like Card, Mobile Phone) and Inherence Factor (like Biometrics). Conjuncture of two or more factors constitutes the multifactor authentication, which increases the security. However, these methods have vulnerabilities, hence, some new factors like location factor added for enhancing the security. Proposed system employs three factors for authentication that are Knowledge Factor, Possession Factor, and Location Factor. A web application is developed by incorporating these factors. Wireshark and Burpsuite tools are used for testing purpose. A browser add on VPN (Virtual Private Network) is used for analyzing the Location factor. Data transmitted over network in encrypted form by enabling HTTPs (Hypertext Transfer Protocol Secure) through Secure Socket Layer (SSL)/Transport Layer Security (TLS) certificates which is verified using Wireshark and if IP address location is mocked by VPN (Virtual Private Network), then also proposed system is working as expected. Since three factors are used, hence, it lowers the risk of various attacks like Man in The Middle Attack (MITM), Social Engineering attacks etc.

Keywords: SFA · MFA · Authentication · Location Factor · HTTPs · SSL · TLS · VPN · MITM

1 Introduction

Internet applications security is an essential concern in the current scenario of the digital era. Everything is available on the Internet either through web applications or through mobile applications. Most things like internet banking, e-commerce, online fee submission, are widely used through internet. These applications have private information of users. Hence, security is major concern to protect the confidentiality of information.

One of the crucial aspects of maintaining security and privacy for the systems is authentication [1]. Authentication is the process where verification of user's credentials are checked, for proving his/her own identity [2]. It covers many different areas such as online payment, Communication systems and Access right management [3]. Most of the services a user accesses are through web applications like the internet banking, online ticket booking and e-commerce websites. Hence providing security to these web application services cannot be overlooked. Authentication is an elementary security mechanism to protect these application services against threats [10]. Evolution of Authentication from SFA to MFA is illustrated in Fig. 1.



Fig. 1. Authentication Evolution.

Use of multiple factors incorporated for better security and mitigating the attacks, is called as MFA. MFA consists of mainly three factors:

1. Knowledge Factor - Something You Know, like PIN, Password.
2. Possession Factor - Something You Have, like Card, Mobile Phone.
3. Inherence Factor - Something You Are, like Biometric Passwords.

Conjuncture of two or more factors constitutes the MFA, which increases the security. However, these methods have vulnerabilities [11]. Hence, some new factors are added like location factor (takes the real time location of user) for enhancing the security. The most widely used authentication mechanism is password based authentication, which is highly susceptible to attacks [12]. Most of users makes use of simple passwords which are easy to guess by brute force attack or dictionary attack. Currently, most of the web applications are using two factor authentication to provide security. Two-factor authentication (Knowledge factor and Possession factor) increases the security to some extent, despite, attacker finds new ways to exploit the security, like use of Sim swap to attack on the OTP (One Time Password) based authentication in Possession factor (Fig. 2).

Now most research focuses on multifactor authentication, where two or more than two factors are used to increase the level of security. Adding more factors does not mean to add various multiple factors, as it increases the complexity of the system, which is not comfortable for users. Hence, adding factors in which user involvement is less so that user's experience is also maintained. Use of multiple solution from same category of factors, for example use of PIN and Password of Knowledge factor, does not constitute MFA. This thesis covers the implementation of MFA.



Fig. 2. Authentication Factors.

1.1 Problem Statements

As security measures increase, cyber threats related to those measures also increase. SFA can easily be susceptible to various attacks like Social Engineering attacks, Shoulder Surfing attacks, Phishing etc.

System using 2-factor Authentication, where, OTP is the 2nd factor, can be susceptible to SIM Swap attack, and MITM attack. Similarly, IP Location based authentication is less secure as attackers can mock their location by VPNs to show as legitimate users and take access to the system.

1.2 Research Objectives

Various attacks are possible on SFA and 2-factor authentication, hence, objective is to provide enhanced security measures using multiple factors to lower the risk of attacks on web applications.

To Use three levels of authentication factors, Password based, OTP-based, and Browser based Geolocation as third level of authentication.

To add and verify self-signed certificates in the web application, enabling HTTPS, to transmit data over network in encrypted form.

Testing of web application and proposed solution with tools like Burpsuite and Wireshark.

2 Literature Review

The literature review aims to provide a comprehensive overview of existing research related to MFA. This section synthesizes key findings, theoretical frameworks, methodologies, and significant trends in the literature, identifying gaps and areas for further investigation.

Abdullah Alabdulatif et al. [4] proposed a novel approach developed for securing ATM transactions, this paper focuses on geo-location-based authentication, as traditional methods like PINs and OTPs have proven to be vulnerable to attacks. In this approach, a user's real-time geolocation is measured through a unique smartphone app, allowing for authentication of every ATM transaction.

Ali Hameed Yassir Mohammed et al. [5] discusses various classifications of authenticators, identity access management protocols, and authentication protocols that are

biometric-based authentication and non-biometric-based authentication and also discusses various security attacks and performance measures used in non-based biometric approaches, provides a security analysis of current OTP generation and distribution methods.

M. Maranco *et al.* [6] proposed end-user security in Cyber Physical System(CPS), contains an internet-enabled digital component. The proposed system presents enhancements to end-user security using MFA techniques. Three distinct works in this study use MFA technique. The MFA technique consists of three distinct layers. The end-user's geo-location is considered the first security gateway by the user distance within the range of CPS and validates the user's current position with predefined geo-location logic. One Time Password (OTP) send through e-mail is used for the second layer. Hash algorithms such as SHA256 or SALTING use the third layer implementation.

Hang Wu [7] proposed Information encryption and Data Storage Security, focuses on communication security and Secure storage through RSA (Asymmetric Encryption), AES (Symmetric Encryption), Digital Signatures, and MD5(Message Digest 5) algorithms, currently used for information security, however, Single Encryption Algorithm does not provide adequate security. This paper focuses on Combining of AES and RSA for encryption and use of RSA Supported Digital Signature for authenticity, transmitted through MD5.

Manju Khari *et al.* [8] describes the various vulnerabilities and attacks in web application and this proposes mechanism to prevent from these attacks, describes the Static and dynamic tainting for prevention. Diego Carrillo-Torres *et al.* [9] proposed that most of the authentication mechanism can be breached by credentials stolen or brute force attack, this proposes an approach of combining textual passwords with graphical image password which can enhance the security of multifactor authentication and this does not require any additional hardware like biometric authentication due to which cost increases.

3 Proposed Methodology

This Section presents the methodological plan that is used to achieve the goals of the proposed work. The methodology serves as the guiding journey of the proposed work. The flow of methodology used for the proposed work is illustrated in Fig. 3 in the form of a flow chart. Methodology describes the overall approach for the progress of research work, like here, to securing the web applications first process is to design and develop the web application with multifactor authentication mechanism and the next steps will go like as in the flow chart of the methodology.

After the literature review, Proposed work will follow a plan to achieve the goals, where, a web application needs to be designed and developed. A web browser is required for accessing web applications and accessing locations for authentication purposes.

Web application development has two main parts, **a) Frontend** which is also called as the client side, which contains the structure and design of web applications where users interact. **b) Backend** which is also called as server side, where the information is stored in the database and this information is used for logical processes and cannot be accessed by users.

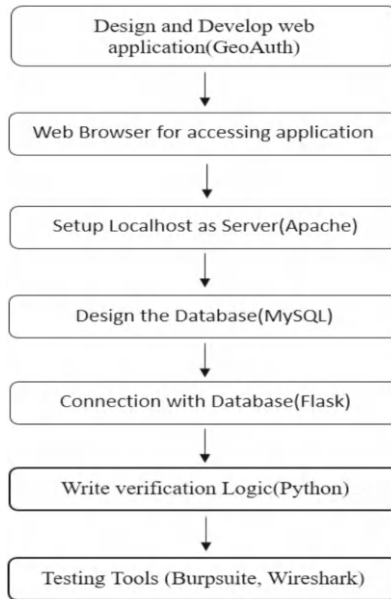


Fig. 3. Proposed Work Methodological Plan.

Frontend of web application is developed by **HTML (Hypertext Markup Language) and CSS (Cascade Style Sheet) and JS (Javascript)**. HTML gives the structure of the application page, CSS gives the presentable and appealing look to application page and JavaScript is used for the client side click events in application page.

Backend of application is developed by Python language framework named Flask, because of its lightweight and minimalistic nature.

The proposed work will take geolocation from the browser. Geolocation from the browser can be taken by the Geolocation API of the browser which is illustrated in Fig. 4.

4 Proposed Work

This section includes the proposed work and includes the flow of the proposed web application.

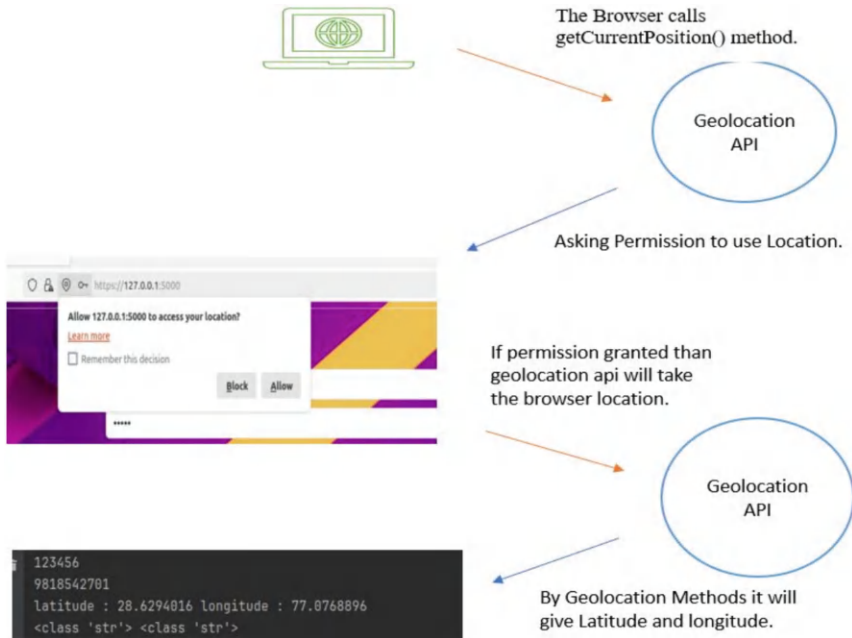


Fig. 4. Geolocation API Working in Browser.

Step 1: The user accesses the web application and fills the authentication details like username/email and password.

Step 2: Authentication details go to the server database where authentication logic is already given to verify the details given by username is authentic or not.

Step 3: If authentication details given by the user are verified?
then Goto Step 4.
Else unsuccessful authentication or Goto Step 1.

Step 4: The second authentication factor OTP is sent to the user's mobile phone, which is given at the registration time and the user needs to submit this OTP to verify the second authentication factor.

Step 5: if OTP matches?
then System will automatically take the Browser based Location
IF this location matches with database stored?
Then third factor is successfully passed.
Else Goto Step 1.
Else Goto Step 1.

Step 6: When all the authentication factors are successfully passed the user is successfully authenticated.

These above steps, are utilized for the proposed work and used for authentication purposes in the developed web application.

4.1 Registration of Authentication Factors

In the Proposed system, different authentication factors must be provided at the time of first registration, username, password, mobile number, and browser based geolocation.

The registration process flow chart is illustrated below in Fig. 5.

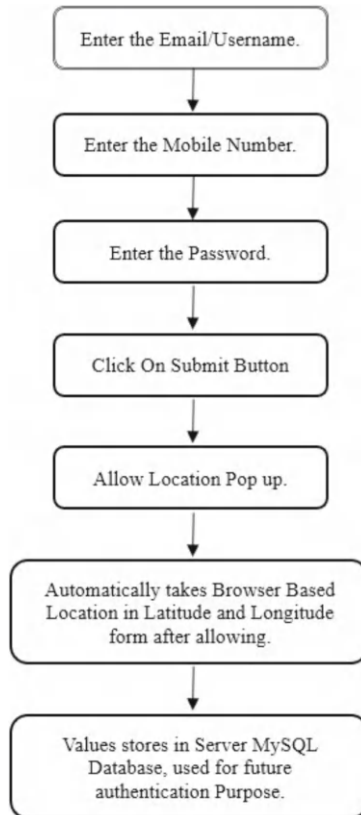


Fig. 5. Registration Phase Flow Chart.

4.2 Authentication of User

For authentication purposes user needs to pass three stages of authentication which are illustrated in Fig. 6 in the form of a flow chart. First Stage is the knowledge based factor where password need to enter, second stage is the Possession factor where OTP send to Mobile number has to enter and third stage is location factor.

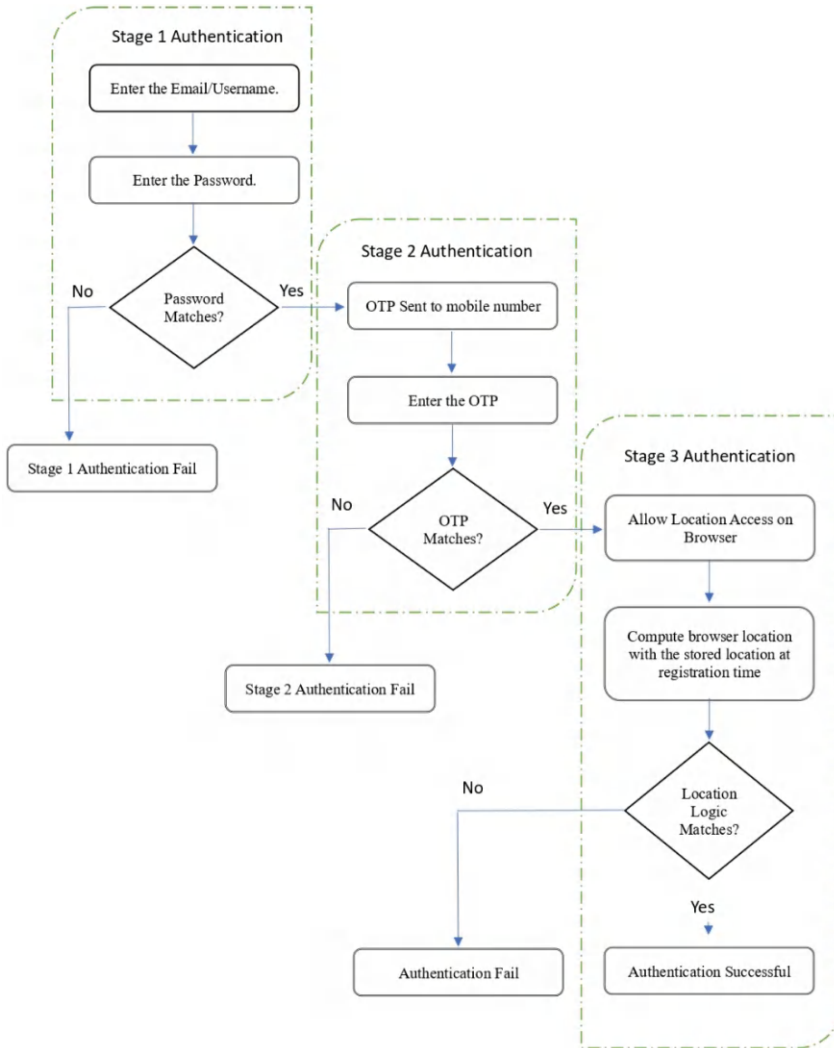


Fig. 6. Authentication Process Flow Chart.

5 Result and Analysis

This section presents the implementation and results of the proposed work. Evaluation is based on the security of the web application against various vulnerabilities and testing done through various tools.

5.1 Methods Investigated

Three factors are used for authentication: Knowledge Factor: Email/Username and Password is used for the first factor of authentication. Ownership factor(Possession factor):

where the user needs their phone to get the OTP on mobile phone and this is the second factor. Location Factor: where the user's browser location is used for authentication purposes and this is the third factor of authentication.

5.2 Experimental Setup

There are a bunch of setups that are used like PyCharm: a Python Development Kit, XAMPP for MySQL and Apache Server, Python language for programming, and Flask Python's framework used for backend logics. HTML, CSS, and Bootstrap is used for the frontend of web applications. Use of localhost as a server for testing and analysis, and use of SSL/TLS self-signed certificate. Wireshark is used for analyzing the HTTP/HTTPS packet. OpenSSL is required for creating self-signed SSL/TLS certificates for enabling HTTPS in web application.

5.3 Implementation of Web Application

This sub-section includes the user interface and implementation of web applications.

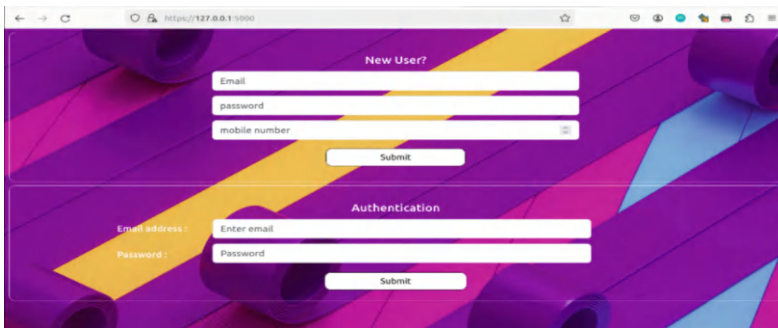


Fig. 7. Application Main Screen

Main screen of the application is shown in Fig. 7, where two sections, Registration and login is present for authentication.

Registration form contains three fields Email, Password, and Mobile number. These three values with location latitude and longitude values are stored in database through HTTPS request. Location values in the form of latitude and longitude captured when user allows the location permission as shown in Fig. 8. The second section of the application is the authentication form where user need to login to confirm the identity.

Login form contains two input fields Email and Password. These two inputs are matched with already registered values in the database, if details are matched then OTP window opens and ensuring the success of first stage of authentication.

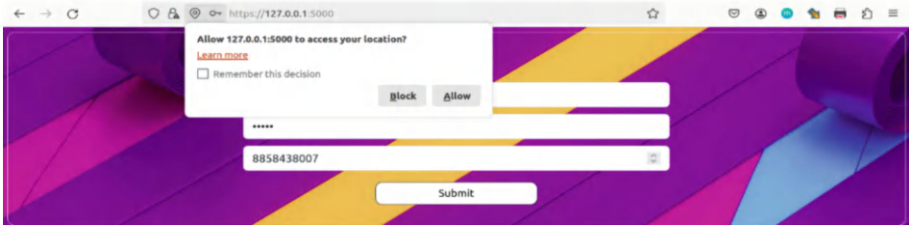


Fig. 8. Location Access Permission.



Fig. 9. OTP window.

After successfully passing of first stage, Server will send an OTP to the user's registered mobile number. OTP received on mobile number will be used as input for the OTP window page as shown in Fig. 9. This OTP needs to be correct, wrong OTP leads to start the process again and second stage authentication fails otherwise.

If Second stage passes successfully, then browser will ask location permission from browser and server will match this allowed location with stored location in the database, if the location matches or found in the radius of pre-determined range through haversine algorithm then authentication is approved otherwise it fails.

When all three stages are successfully passed then authentication gets successful otherwise not, if any one stage is not successfully passed then user needs to repeat all the steps.

5.4 Secure Transmission of Data

SSL/TLS certificates are needed for secure and encrypted communication over the network. In the proposed system self-signed certificates are used that generated by OpenSSL.

Figure 10 shows the Wireshark tool window, showing unencrypted data transmitted over network through HTTP, confirming that HTTP is not secure.

When SSL/TLS certificate is attached in the application then HTTPS functionality enabled and the information sent over the network is in encrypted form.

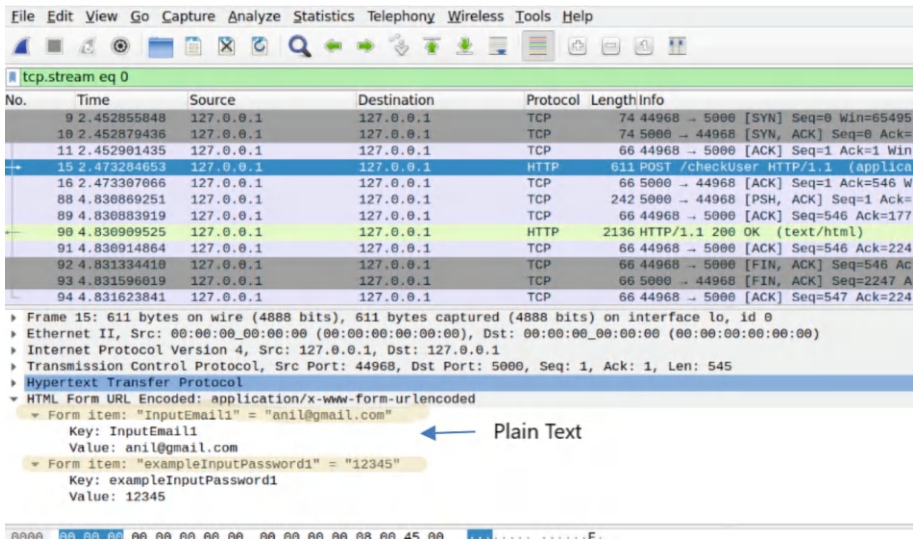


Fig. 10. Data sent without Encryption (HTTP).

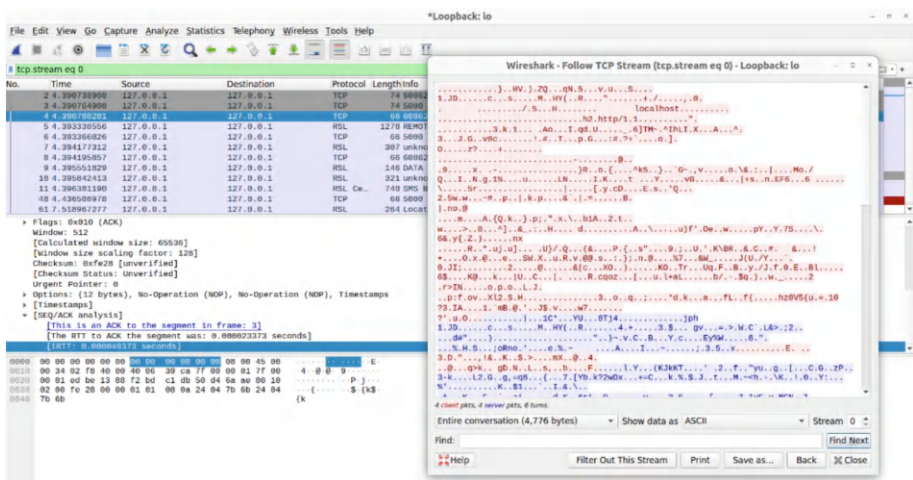


Fig. 11. HTTPs Encrypted Data.

Above Fig. 11 is showing the Wireshark window, showing Encrypted data transmitted over the network through HTTPs.

5.5 VPN Effect

In the proposed work Browser Based Geolocation is used as an authentication factor which is different from the user's IP address location, IP address location can be easily mocked by various tactics like VPN, but in this solution, location mocked by VPN does

not help to give entry into the system. For testing and analysis purpose, third party add on VPN is used in browser for mocking the location but still the system works fine.

6 Conclusion and Future Scope

In today's rapidly growing cyber world, most of the daily tasks are done on the internet and these tasks are done through any application either Mobile applications or Web application. In this proposed approach three factors are used for increasing the security level and these three factors are Knowledge Factor where a Password need to be entered, Second is OTP sent to user's registered Mobile Number which comes into the Possession Factor and Third is Browser Based Geolocation which is Location Factor. HTTPs is used for sending the data over network in encrypted form. The proposed solution can be used as the attendance system where user need to present in the same location where registration happens, or in the radius of some distance for authentication purpose. It may be used with some more implicit authentication factor to enhance the security of application. Most of the things of objective is fulfilled here like username and password are sending across network in encrypted form because of SSL/TLS certificates, if sim swap happens to get the OTP there is still one more location-based factor which lowers the risk of attack. Proposed solution has location factor as the third factor and it is also giving effective results on VPN mocking so this third factor is good as authentication factor and increases the security level.

New implicit authentication factors need to consider so that lesser user involvement makes system less complex and this may also reduce the attacks. Potential Challenges that are faced in implementation is, this system is most effective for authentication for static locations but not for dynamic locations, so it can be improved in future for dynamic location changes also. Mobile Applications for iOS and android environment can be developed using above authentication factors for authentication purpose. Use of some encryption methods to store the data in encrypted form make use of prepared statement with parametrized arguments and input validation help to prevent the SQL injection attacks.

References

1. Khari, M., Garg, A.K., Gandomi, A.H., Gupta, R., Patan, R., Balusamy, B.: Securing data in Internet of Things (IoT) using cryptography and steganography techniques. *IEEE Trans. Syst. Man Cybernet. Syst.* **50**(1), 73–80 (2019)
2. Suleski, T., Ahmed, M., Yang, W., Wang, E.: A review of multi-factor authentication in the Internet of Healthcare Things. *Digit. Health* **9**, 205520762311771 (2023)
3. Ometov, A., Bezzateev, S., Mäkitalo, N., Andreev, S., Mikkonen, T., Koucheryavy, Y.: Multi-factor authentication: a survey. *Cryptography* **2**(1), 1 (2018)
4. Alabdulatif, A., Samarasinghe, R., Thilakarathne, N.N.: A novel robust geolocation-based multi-factor authentication method for securing ATM payment transactions. *Appl. Sci.* **13**(19), 10743 (2023)
5. Mohammed, A.H.Y., Dziyauddin, R.A., Latiff, L.A.: Current multi-factor of authentication: approaches, requirements, attacks and challenges. *Int. J. Adv. Comput. Sci. Appl.* **14**(1), 166 (2023)

6. Maranco, M., Logeshwari, R., Sivakumar, M., Manikandan, V.: Improvised multi-factor authentication for end-user security in cyber physical system. *Int. J. Intell. Syst. Appl. Eng.* **12**(15s), 416–424 (2024)
7. Wu, H.: Information encryption and data storage security. In: *Highlights in Science, Engineering and Technology 2023 International Conference on Artificial Intelligence, Database and Machine Learning (AIDML 2023)*, vol. 57, pp. 189–194 (2023)
8. Khari, M., Sangwan, P.: Web-application attacks: a survey. In: *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, pp. 2187–2191. IEEE (2016)
9. Carrillo-Torres, D., Pérez-Díaz, J.A., Cantoral-Ceballos, J.A., Vargas-Rosales, C.: A novel multi-factor authentication algorithm based on image recognition and user established relations. *Appl. Sci.* **13**(3), 1374 (2023)
10. Khari, M., Shrivastava, G., Gupta, S., Gupta, R.: Role of cyber security in today's scenario. In: Kumar, R., Pattnaik, P.K., Pandey, P. (eds.) *Detecting and Mitigating Robotic Cyber Security Risks*, pp. 177–191. IGI Global (2017)
11. Wazid, M., Das, A.K., Odelu, V., Kumar, N., Conti, M., Jo, M.: Design of secure user authenticated key management protocol for generic IoT networks. *IEEE Internet Things J.* **5**(1), 269–282 (2017)
12. Wazid, M., Das, A.K., Kumar, N., Vasilakos, A.V.: Design of secure key management and user authentication scheme for fog computing services. *Futur. Gener. Comput. Syst. Comput. Syst.* **91**, 475–492 (2019)



Human Activity Recognition Using Machine Learning

Meena Tiwari^(✉)  and Prateek Gupta

Department of CSE SRIST College, Jabalpur, India
phmeenatiwari@gmail.com

Abstract. Human AR (HAR) use of IAs has generated much attention on their applications on various campuses, including health services, deportations, security, and security situations. This research focuses on creating powerful computations and models for data mining and collecting accurate human information as a source of information obtained from sensors, accelerometers, sensors, and wearable devices. Artificial intelligence techniques including deep learning, support vector machines (SVMs), and irregular plots are used to process and analyze sensor information, extraction, and meaningful aspects to separate the results, such as a fireplace, adjustment, senders, parse, and movimientos complejos. The goal is to create HAR brands trusted for preparation to realize accurate information about constant and precise actions, carry out mass medical surveillance, provide Bienstar personalized customization, and create the highest association between people and computers in an intelligent environment.

Keywords: Human Activity Recognition (HAR) · Machine Learning · Sensor Data · Accelerometers · Support Vector Machines (SVM) · Random Forests · Activity Classification · Healthcare Monitoring · Fitness Tracking · Pattern Recognition

1 Introduction

Human activity recognition (HAR) has become a critical area in the artificial intelligence and artificial consciousness space. It involves the development of computational models and calculations aimed at perceiving, organizing and deciphering human activities and behaviors in light of the information received from various sensors. The use of data from accelerometers, spinners, and wearable devices has fundamentally stimulated research in this area, allowing the creation of systems suitable for distinguishing between exercises such as walking, running, sitting, standing, and compound exercise events.

Artificial intelligence strategies, including deep learning techniques such as brain organization, and traditional computing such as Backing Vector Machines (SVM) and Irregular Woods, play an important role in processing and deciphering large amounts of sensory information. These procedures aim to extract

important examples and highlights necessary for the precise organization and continuous recognition of various human activities [1].

The meaning of HAR extends to a variety of areas, including healthcare, sports and wellness testing, security, and living conditions. In the healthcare industry, HAR systems can support remote patient detection and monitoring by performing exercises and detecting oddities. In sports and wellness, these systems allow individual training regimens and performance tests to be created. Furthermore, HAR adds to creating flexible environments that improve human-PC connectivity and work with more efficient and intuitive user interfaces. This article plans to explore and analyze the systems, progressions, challenges, and possible applications of human action recognition using AI. Exploring different strategies, information processing techniques and certified applications means deepening understanding and developing effective HAR frameworks that can impact various aspects of our daily lives.

2 Related Work

Meena Tiwari et al. [1] in this work, the review utilized four divisions the Bases association, the multistakeholder association, the nearest neighbors, and extra vector hardware. What's more, the arrangement was tried in four noticeable examinations. These are cases that are addressed with 70% remedy, 30% test, 60% planning, 20% test, at least 60% readiness, 40% test, and 10 cases. From the outcomes, it tends to be finished up very well that all the development of a typical division finishes portrayal of over 90%. Notwithstanding, SVM is as yet the best divider proposed to be counted. Fingerprints serious areas of strength for are of authenticity in court. Concerning the astounding power as a finger acknowledgment framework, this post attempts to investigate connections between ladies. Perusing and standing are the consequence of our insights and part of the essential significance, of seeing the right qualities for sex. This plan requires a demonstrated test structure for special items to lessen mathematical time and further develop execution.

Suchita Tarare et al. [2] additionally make sense of the entire course of the plan above. The DWT change gives different finger impression structures from an informational index (picture perusing) to make a data set of capabilities utilized as a survey table to recognize obscure fingerprints from different fingerprints (fingerprints) utilized for testing. The divider KNN assigns one of the two sets for testing fingerprints.

S. Sudha Ponnarasi [3] depends on the recognizable uniqueness test system of the toothpaste. The subject of the overview is more than 500 individuals, of which 250 are men and 250 ladies somewhere in the range of 1 and 90. Number of white lines, number of chains, and type conditions. The vector machine was utilized mainly and came with serious outcomes.

H. Yang et al. [4] in this paper The Inertial Estimation Units (IMUs) have been generally utilized in human movement acknowledgment (HAR) for information procurement. Be that as it may, most openly accessible datasets in light of

IMUs just include information from a few body parts and are somewhat homogeneous. Utilizing information from a few body parts can be restricted in specific explicit or complex movement acknowledgment assignments. Consequently, we made another HAR dataset named the Multi-Position Joint Point Human Action Dataset (MPJA-HAD). Other freely accessible datasets just hold back the crude inertial sensor readings. The MPJA-HAD dataset likewise gives joint point changes from every one of the 15 body positions. Joint points straightforwardly connect with human exercises performed and trial results show the seriousness of the made dataset in HAR undertakings.

Sheharyar Khan et al. [5] Wearable sensor-based Human development Affirmation (HAR) is getting a reputation for its moderateness and low computational solicitations. These sensors are extensively used in clinical consideration and perception. Nevertheless, using PDA sensors for HAR can be mixed up in light of their non-fixed position. This study uses the transparently open Human Activity Affirmation Trondheim dataset (HARTH) to encourage a HAR model. The proposed HAR model is fit to see different human standard routine practices in the free-living environment. Separating individual customary lead in the wild (free-living) stays testing since individuals take part in unscripted regular activities. While working with controlled conditions (prearranged information) can deliver ideal outcomes, this framework frequently battles when applied, in actuality, situations. Numerous AI classifiers are tried on time-space highlights extricated from sensor information, with the Multi-facet Perceptron (MLP) classifier accomplishing a great 92.92% precision, making a critical commitment to the HAR field.

N. A. Choudhury et al. [6] in this paper, human activity assertion is the most well-known approach to perceive an individual's daily life activities using sensor credits and accurate learning estimates. Perceiving complex human activities is depressing as it is difficult to overcome prolonged circumstances and remove capable components from the approximate sensor data. This letter proposes a skilled and lightweight lost canine significant learning model for review complex human exercises utilizing physiological electromyography (EMG) sensors and changed highlight learning. The proposed Convolutional Mind Affiliations - Long Blaze Memory (CNN-LSTM) consolidates different 1-D convolution layers for spatial part extraction and later arrangements with include guides made for unpredictable layers to see passing conditions of significant length. Consolidating an inexact EMG informational index in light of physiological sensors and unimportant preprocessing, we set up and tried our proposed model and accomplished the greatest exactness of 84.12% and a run of the mill precision of 83%. The proposed model beat the reference models with ideal execution benefits and summed up the models essentially quicker than other significant learning models.

Trung Q. Duong et al. [7] Quantum processing is imagined as a developing worldview for tackling computationally complex enhancement issues with an enormous number factorization and thorough hunt. Lately, there has been an increasing improvement of the size of multi-layered datasets, the data yield

space dimensionality, and data structures. Thus, the customary artificial intelligence approaches in data planning and taking care of have shown their limited abilities to enlist to help the sixth time (6G) networks with significantly special applications and organizations. In such manner, the quick creating quantum registering with AI for 6G organizations is researched. Quantum AI calculation can altogether improve the handling productivity and dramatically computational accelerate for powerful quantum information portrayal and superposition system, profoundly equipped for ensuring high information stockpiling and got interchanges. We present the cutting edge in quantum figuring and give a far-reaching outline of its true capacity, by means of AI draws near. Moreover, we present quantum-propelled AI applications for 6G organizations regarding asset distribution and organization security, taking into account their empowering innovations and likely difficulties.

R. Roscher et al. [8] AI techniques have been strikingly fruitful for an extensive variety of use regions in the extraction of fundamental data from information. A thrilling and generally late advancement is the take-up of AI in the innate sciences, where the significant objective is to acquire novel logical bits of knowledge and disclosures from observational or reproduced information. An essential for getting a logical result is space information, which is expected to acquire make sense of capacity, yet in addition to improving the logical consistency. In this article, we survey logical AI considering applications in the innate sciences and talk about three center components that we recognized as significant in this specific circumstance: straightforwardness, interpretability, and making sense of capacity. Regarding these center components, we give an overview of late logical works that consolidate AI and the way that reasonable AI is utilized in blend with space information from the application regions.

A. T. Prihatno et al. [9] The huge advancement of Industry 4.0 is indistinguishable from the progress of AI. To shield the producing area from undesirable occasions, for example, electrical disappointments because of the elevated degree of dampness, the prescient upkeep in view of AI ought to be grown precisely. This paper depicts the execution work of foreseeing Relative moisture (RH) in the savvy manufacturing plant's current circumstance by involving the Irregular Woods technique as a piece of AI. To help information unwavering quality and interoperability in a savvy production line climate, IIoT gadgets-based oneM2M standard stage was utilized to gather the information. The consequence of this Arbitrary Backwoods strategy for foresee relative dampness shows 82.49% which is considered as a phenomenal exactness. This examination objective might add to the assembling fields to have the option to bring down the expense and increment productivity in upkeep.

K Sai Dhanush et al. [10] The dimensionality revile happens because of the great dimensionality of information, which is significant in quality articulation information examination. The marking system is perplexing with the accessible dataset and tedious. The preparation interaction is likewise perplexing with AI draws near. Different existing methodologies expect to upgrade interpretability and address the smoothing issue and marking issues. Here, two assorted clas-

sifiers, k-NN and SVM are utilized to order the information and inspect the highlights expected for characterization. The presentation is examined utilizing five assorted datasets: BRCA, GSE412, lung, Leukemia and DLBCL to offer commonsense help. In light of the examination, it is seen that k-NN gives better expectation results contrasted with SVM and upgrades execution. Both these models lay out better compromises contrasted with different methodologies.

Yang J. et al. [11] in this paper, propose an efficient technique to highlight learning in HAR space. This strategy uses deep convolutional brain organizations (CNN) to obtain highlights from the raw information accordingly. The deep design enables the extraction of unique higher-level representations from low-level raw time series signals. By using managed learning with marked information, the elements proposed by CNN become more discriminative. Combining learning and component characterization within a single model results in common improvement. These unmistakable benefits of CNN result in better performance than other HAR calculations, as demonstrated by tests from the Open Door Action Acknowledgement Challenge and other benchmark data sets.

Guan et al. [12] in this paper created a lean training methodology for LSTM organizations and consolidated different LSTM learners into aggregate classifiers. Our results show that deep LSTM learner groups outperform individual LSTM organizations, thereby driving best-in-class human motion recognition using wearable devices. Extensive exploratory evaluations on three standard benchmarks (Opportunity, PAMAP2, Skoda) show the fantastic recognition capabilities of our methodology and its true capability for genuine HAR applications.

Chen et al. [13] in this paper applied ensemble methods to HAR, demonstrating improved accuracy and robustness over single-model approaches, ensemble methods techniques like boosting and bagging have been employed to aggregate the predictions of multiple models, reducing variance and bias.

Wang et al. [14] in this review presents an overview of state-of-the-art deep learning strategies for sensor-based human motion recognition. We initially introduce the multi-methodology of haptic feedback and provide data to public datasets that can be used for evaluation in various testing companies. Then, at that point, we propose another scientific categorization to structure deep techniques by challenges. Difficulties and deep techniques related to challenges are summarized and examined to shape an outline of flow research progress. Towards the end of this work, we discuss open problems and provide some insights for future bearings.

Kwapisz et al. [15] in this overview presents the latest proposed arrangements for each task in the human motion clustering process, i.e., acquisition, preprocessing, information partitioning, highlight extraction, and characterization. The agreements are analyzed highlighting their advantages and disadvantages. To conclude, the study also presents the measurements commonly used to evaluate the decency of a classifier and the mobile phone inertial signal data sets that are generally used in the evaluation stage.

3 Proposed Method

This paper presents a model that utilizes information gathered from sensors to foresee human exercises like strolling, strolling uphill, strolling downhill, sitting, standing, or resting. In this paper, information is gathered by utilizing two significant sensors like Gyration and an accelerometer. The dataset comprises of sensor information recorded from thirty unique individuals with appended sensors to their body. By utilizing sensors like Gyration and accelerometer which are essentially additionally connected in our advanced mobile phones, have caught '3-pivotal straight acceleration' (named Acc-XYZ in the dataset) from the accelerometer and '3-hub rakish speed' (named Gyro-XYZ in the dataset) from Whirligig with a few varieties.

In the wake of gathering the information, they are cleaned utilizing different channels. The sensor signals are pre-handled utilizing commotion channels and then inspected to fixed width window with half of cross-over. From every window, an element vector is determined by working out factors from time and recurrence space. The speed increase signal got from the sensors is isolated into body sign and gravity signal utilizing some low pass channels. After that precise speed is determined so as to get the Jerk signals. Jerk signals are determined for Body Speed increase readings. The size of these signs was determined utilizing Euclidian standards. The sizes are addressed as highlights. The Design of the proposed model is given beneath (Fig. 1):

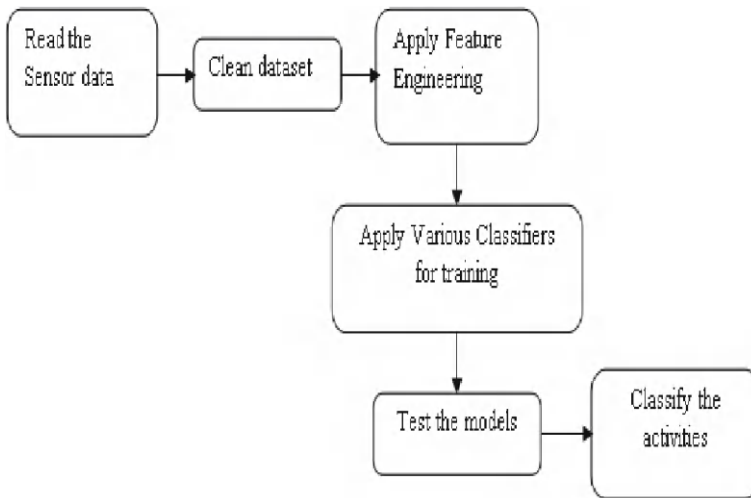


Fig. 1. Proposed Model architecture.

3.1 Proposed Model Steps

The following data cleaning steps are applied:

1. Check for duplicate and Null Values.
2. Check for data imbalance.
3. Apply t-SNE feature selection on the data.

The following feature engineering steps are applied:

1. Select static activities like sitting, standing and lie down.
2. Select moving activities like Walking, Walking-Upstairs, Walking-Downstairs.
3. Apply EDA on Static and Dynamic Activities.
4. Apply Acceleration magnitude to differentiate between static and moving activities. For separation following observations are applied.
 - If Acceleration Mean is < -0.8 then the Activities are either Standing or Sitting or Laying.
 - If the Acceleration Mean is > -0.6 then the Activities are either Walking or Walking Downstairs or Walking Upstairs.
 - If Acceleration Mean > 0.0 then the Activity is Walking Downstairs.
 - We can classify 75% of the Activity labels with some errors.

Position of Gravity Acceleration also matters for activity. The following observations are applied.

- If the angle of the X-axis, and gravity Mean > 0 then Activity is Lying.
5. Again save the data to train.csv and test.csv.
 6. Read train data and test data from the CSV files.
 7. Create the proposed model using train data.
 8. Evaluate the model by printing the confusion matrix and various results
 9. Apply the DBSCAN clustering model.
 10. Remove the Outliers using LOF, Isolation Forest and ABOD.
 11. Apply PCA and Gaussian Random Projection. 12 Again apply t-SNE.
 13. Apply Random search and Grid Search.
 14. Train the model using the classifiers.
 15. Evaluate the model by printing the confusion matrix and various results (Fig. 2).

Step 7: Remove outliers by using LOF, ABOD and Isolation Forest.

Step 8: Detect and classify the activities using Base classifiers and Proposed MLP classifier.

Step 9: Evaluate the results.

Step 10: apply PCA on cleaned data.

Step 11: Detect and classify the activities using Base classifiers and Proposed MLP classifier.

Step 12: Compare the results. Step 13: End of algorithm.

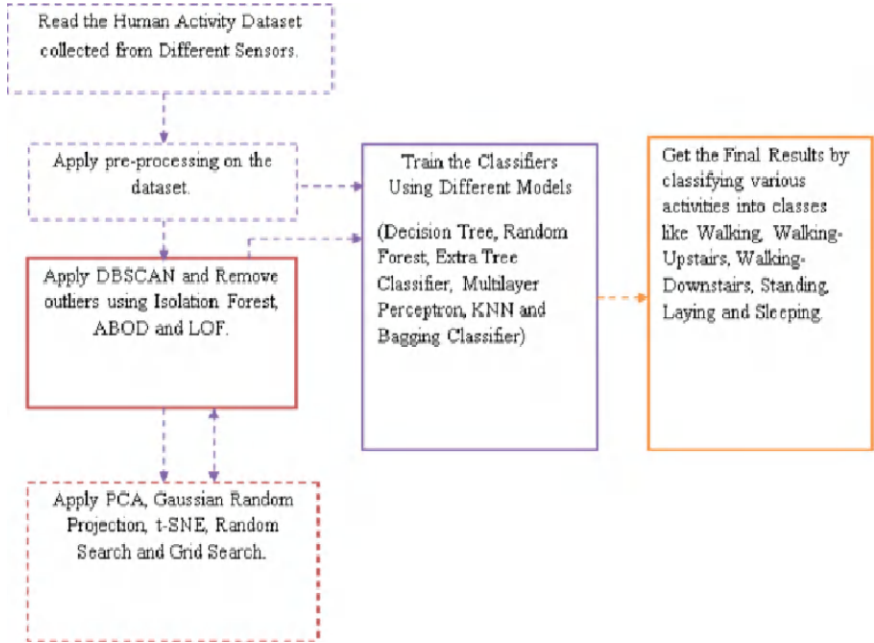


Fig. 2. Proposed training steps.

3.2 Results and Comparison

(See Figs. 3 and 4)

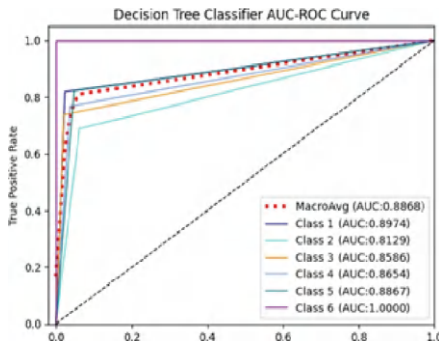


Fig. 3. Result of Decision Tree without pre-processing.

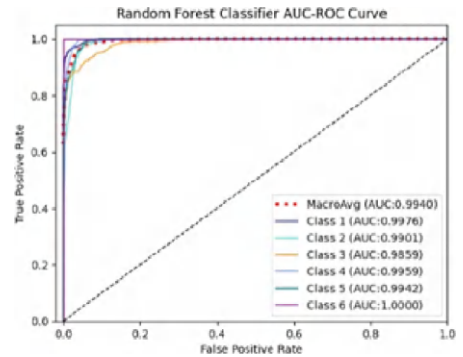


Fig. 4. Accuracy RF Classifier without pre-processing

3.3 Proposed Algorithm

- Step 1: Read the dataset from various sensors.
- Step 2: Apply preprocessing on the dataset.
- Step 3: Split the dataset into train and test data.
- Step 4: Detect and classify the activities using Base classifiers and Proposed MLP classifier.
- Step 5: Evaluate the results.
- Step 6: Apply Clustering on train and test data (Figs. 5, 6, 7 and 8).

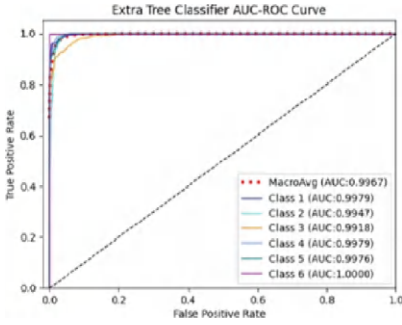


Fig. 5. Accuracy of MLP Classifier

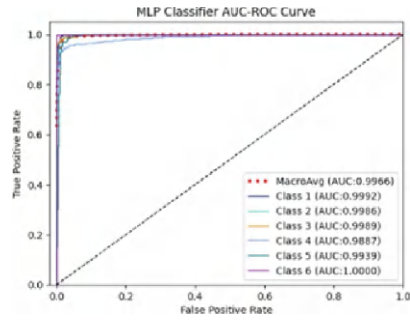


Fig. 6. Accuracy of Extra Tree Classifier

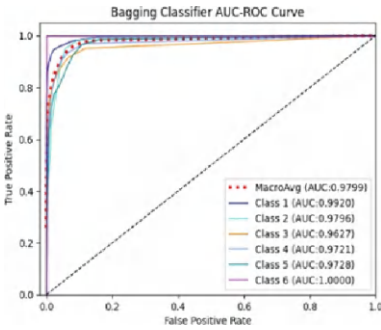


Fig. 7. Accuracy of Bagging Classifier without pre-processing.

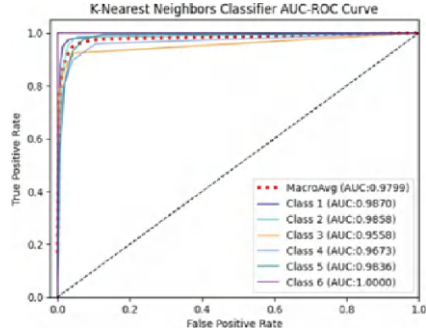


Fig. 8. Accuracy of KNN Classifier without pre-processing.

3.4 Comparison of Results

In this part, we will look at the aftereffects of existing strategies as well as proposed techniques. The boundaries for contrasting the outcomes are exactness, accuracy, review, and F1-score. The outcomes are determined on the first dataset acquired from the Kaggle sites. The outcomes are looked at in three changed

tables. Table 1 is the precision correlation without applying preprocessing techniques. Table 2 is the precision correlation with applying anomaly’s evacuation methods and Table 3 is the examination for the outcomes with PCA and expulsion of exceptions. In every one of the three cases the proposed MLP classifier beats different classifiers.

Table 1. Accuracy Comparison without preprocessing

Implemented Algorithm (Without Preprocessing)	Accuracy in %
Decision Tree	85.95
Random forest	92.09
Extra Tree Classifier	94.02
Bagging classifier	89.17
KNN	90.43
MLP Classifier (Proposed)	94.74

Table 1 presents the accuracy of various algorithms implemented without preprocessing. The algorithms evaluated include Decision Tree, Random Forest, Extra Tree Classifier, Bagging Classifier, K-Nearest Neighbors (KNN), and the proposed Multi-Layer Perceptron (MLP) Classifier. The results are as follows (Fig. 9):

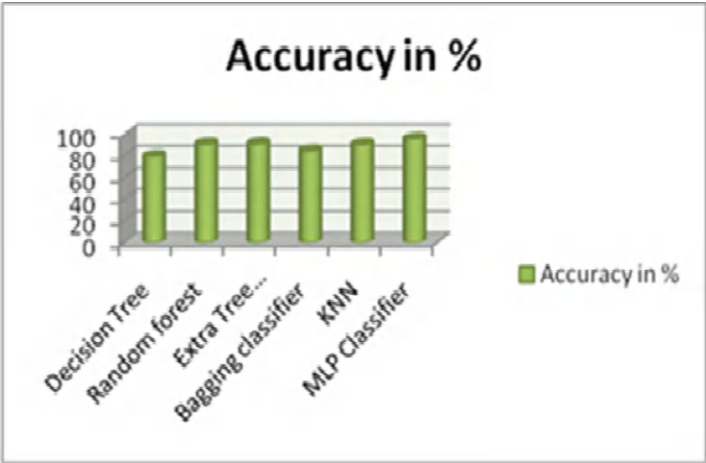


Fig. 9. Graph Chart for Accuracy Comparison.

These results demonstrate that while preprocessing generally improves algorithm performance, certain advanced models such as the MLP Classifier, Ran-

dom Forest, and Extra Tree Classifier still achieve high accuracy without it. The MLP Classifier stands out with the highest accuracy, indicating its robust learning capabilities. Ensemble methods like Random Forest and Extra Tree Classifiers also show strong performance, highlighting their effectiveness in handling raw data. Simpler models like the Decision Tree benefit less from the absence of preprocessing, emphasizing the importance of preprocessing for enhancing the performance of less complex algorithms.

Table 2. Accuracy Comparison without preprocessing

Implemented Algorithm (With Preprocessing)	Accuracy in %
Decision Tree	85.57
Random forest	92.98
Extra Tree Classifier	94.19
Bagging classifier	88.90
KNN	90.43
MLP Classifier (Proposed)	95.96

Table 2 presents the accuracy of various algorithms implemented with preprocessing. The algorithms tested include Decision Tree, Random Forest, Extra

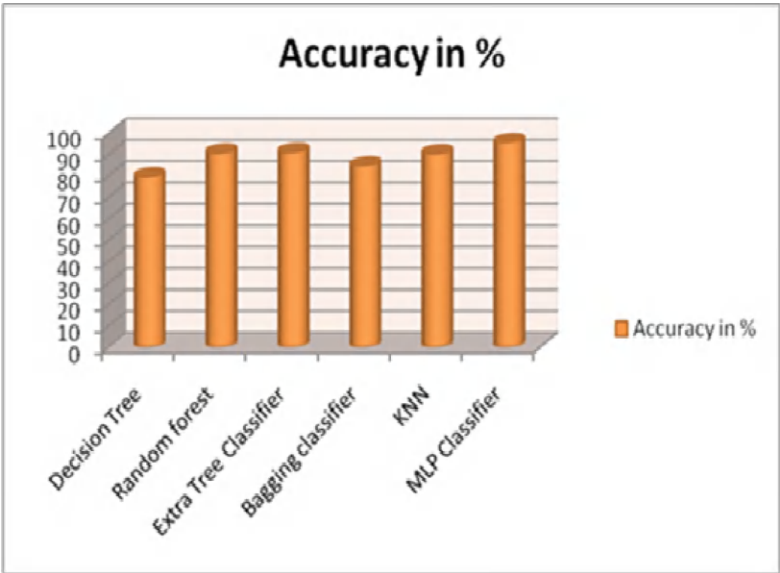


Fig. 10. Graph Chart for accuracy Comparison with preprocessing.

Tree Classifier, Bagging Classifier, K-Nearest Neighbors (KNN), and the proposed Multi-Layer Perceptron (MLP) Classifier (Fig. 10).

These results underscore the importance of preprocessing in enhancing algorithm performance. The MLP Classifier stands out with the highest accuracy, indicating its effectiveness in leveraging the benefits of preprocessing to achieve superior results. Ensemble methods like Random Forest and Extra Tree Classifier also show significant improvements, highlighting their robustness and adaptability to enhanced data. Simpler models like Decision Trees and KNN, while benefiting from preprocessing, do not perform as well as the more complex algorithms, emphasizing the advantage of using advanced methods for achieving higher accuracy.

Table 3. Accuracy Comparison with preprocessing and PCA

Implemented Algorithm (With Preprocessing after PCA)	Accuracy in %
Decision Tree	79.57
Random forest	90.77
Extra Tree Classifier	91.11
Bagging classifier	84.79
KNN	90.43
MLP Classifier (Proposed)	95.69

Table 3 presents the accuracy of various algorithms implemented with preprocessing followed by Principal Component Analysis (PCA). The evaluated algorithms include Decision Tree, Random Forest, Extra Tree Classifier, Bagging Classifier, K-Nearest Neighbors (KNN), and the proposed Multi-Layer Perceptron (MLP) Classifier. The results highlight the following performance metrics (Fig. 11):

These results highlight that preprocessing and PCA can significantly enhance the performance of complex algorithms, especially ensemble methods and neural networks. The MLP Classifier stands out with the highest accuracy, underscoring its effectiveness in handling preprocessed and dimensionally reduced data. Ensemble methods like Random Forest and Extra Tree Classifier also perform exceptionally well, indicating their robustness and adaptability to preprocessing techniques. Simple models like Decision Trees benefit less from these techniques, emphasizing the importance of model complexity and advanced algorithms in achieving high accuracy.

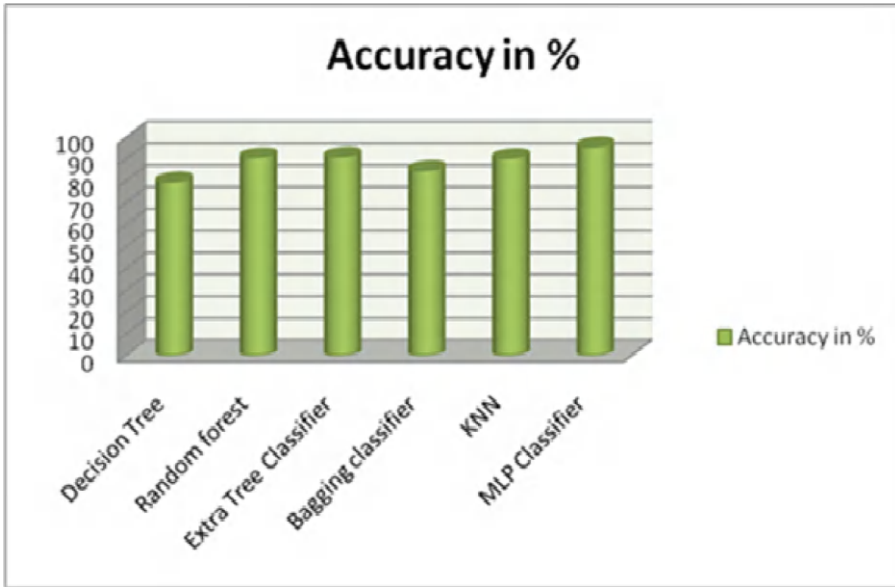


Fig. 11. Graph Chart for accuracy Comparison with preprocessing and PCA

4 Conclusion

The evaluation of various human activity recognition strategies has demonstrated the effectiveness of different preprocessing techniques on classification performance. By comparing the results based on accuracy, precision, recall, and F1-score, it was evident that preprocessing methods significantly influence the outcomes. In all three scenarios, the proposed Multilayer Perceptron classifier consistently outperformed other classifiers, indicating its robustness and superior ability to accurately recognize human activities. This highlights the importance of selecting appropriate preprocessing strategies and robust classifiers in achieving optimal performance in HAR tasks. The ongoing development in HAR using machine learning underscores the potential for creating intelligent systems that seamlessly integrate into our daily lives, improving efficiency, safety, and overall quality of life. The future of HAR promises even more innovative solutions, driven by the synergy of data science and human-centric design.

References

1. Tiwari, M., et al.: Development of association rule mining model for gender classification. In: 1st International Conference on Computational Research and Data Analytics (ICCRDA 2020) (2021)
2. Tarare, S., et al.: Fingerprint based gender classification using DWT transform. In: IEEE International Conference on Computing Communication Control and Automation (2015). 978-1-4799-6892-3/15

3. Sudha Ponnarasi, S., Rajaram, M.: Gender classification system derived from fingerprint minutiae extraction. In: International Conference on Recent Trends in Computational Methods, Communication and Controls Proceedings Published in International Journal of Computer Applications (IJCA) (2012)
4. Yang, H., Wen, X., Geng, Y., Wang, Y.: MPJA-HAD: a multi-position joint angle dataset for human activity recognition using wearable sensors. In: 2022 International Conference on Advanced Mechatronic Systems (ICAMechS) (2022). <https://doi.org/10.1109/ICAMechS57222.2022.10003441>
5. Khan, S., Noorani, S.H., Arsalan, A., Mahmood, A., Rauf, U., Ali, Z.: Classification of human physical activities and postures during everyday life. In: 2023 18th International Conference on Emerging Technologies (ICET) (2023). <https://doi.org/10.1109/ICET59753.2023.10374573>. <https://ieeexplore.ieee.org/author/37089815657>
6. Choudhury, N.A., Soni, B.: Enhanced complex human activity recognition system: a proficient deep learning framework exploiting physiological sensors and feature learning. *IEEE Sens. Lett.* **7**(11) (2023). <https://doi.org/10.1109/LSENS.2023.3326126>.
7. Duong, T.Q., Ansere, J.A., Narottama, B., Sharma, V., Dobre, O.A., Shin, H.: Quantum-inspired machine learning for 6G: fundamentals, security, resource allocations, challenges, and future research directions. *IEEE Open J. Veh. Technol.* **3**. <https://doi.org/10.1109/OJVT.2022.3202876>
8. Roscher, R., Bohn, B., Duarte, M.F., Garcke, J.: Explainable machine learning for scientific insights and discoveries. *IEEE Access* **8**, 42200–42216. <https://doi.org/10.1109/ACCESS.2020.2976199>. Electronic ISSN 2169- 3536
9. Prihatno, A.T., Nurcahyanto, H., Jang, Y.M.: Predictive maintenance of relative humidity using random forest method. In: 2021 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC). <https://doi.org/10.1109/ICAIIIC51459.2021.9415213>
10. Sai Dhanush, K., Sudha, S.: Gene expression analysis using SVM and KNN classifiers on various datasets. In: 2023 5th International Conference on Smart Systems and Inventive Technology (ICSSIT). <https://doi.org/10.1109/ICSSIT55814.2023.10061014>
11. Yang, J., Nguyen, M.N., San, P.P., Li, X., Krishnaswamy, S.: Deep convolutional neural networks on multichannel time series for human activity recognition. In: Proceedings of the 24th International Conference on Artificial Intelligence, pp. 3995–4001 (2015)
12. Guan, Y., Plötz, T.: Ensembles of deep LSTM learners for activity recognition using wearables. *Proc. ACM Interact. Mob. Wearable Ubiquit. Technol.* **1**(2), 1–28 (2017)
13. Chen, Z., Liu, L., Sun, H.: Human activity recognition using ensemble learning. In: 2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC), pp. 485–490 (2019)
14. Wang, Z., Wang, H., Liu, Q.: Deep learning for sensor-based human activity recognition: a comprehensive review. *J. Ambient. Intell. Humaniz. Comput.* **14**(1), 155–169 (2023)
15. Kwapisz, J.R., Weiss, G.M., Moore, S.A.: Activity recognition using cell phone accelerometers: a survey of recent advances. *ACM Comput. Surv.* **52**(5), Article no. 95 (2020)



Predictive Analytics of Injection Attacks in Web Applications

U. Farjana^(✉) and Gowrineni Harshitha Sai

Department of Computer Science and Engineering, Rajalakshmi Engineering College, Chennai,
Tamil Nadu 602105, India
farjana.u@rajalakshmi.edu.in

Abstract. Cybersecurity threats such as SQL injection and Cross-site Scripting (XSS) are examples of injection attacks that pose a substantial risk to both individuals and organizations. Through the use of software application vulnerabilities, these assaults give malevolent actors the ability to enter systems without authorization, steal confidential information, or jeopardize system integrity. The ever-changing nature of injection threats makes traditional rule-based security methods ineffective. The suggested model concentrates on using machine learning approach to successfully detect these injection assaults in order to address this difficulty. The main goal of this research is to create a reliable solution for identifying injection attacks in databases and online applications. The system will continuously analyze application input by using supervised machine learning method (Multinomial Naïve Bayes). It is capable of identifying patterns linked to injection attacks, such as peculiar input patterns and unexpected actions. The website's activity is monitored and a track of these instances is kept in a different database to help in detecting and preventing similar assaults. The necessary analysis is derived from this occurrence history. The use of this study can help in predicting and protecting the web applications from future possible attacks.

Keywords: SQL injection · Cross site Scripting (XSS), · Predictive analytics · Multinomial Naïve Bayes (MNB)

1 Introduction

One common and serious kind of cybersecurity hazard is injection assaults. By inserting unauthorized code or data into databases or applications, these attacks take advantage of the target software weaknesses. Often, the goal is to steal confidential data or jeopardize the integrity of the programs. Injection is listed as A03 (third most critical risk) in the OWASP Top 10 – 2021 report. The OWASP Top Ten is an open-source document that provides standard awareness for developers and web application security about the most critical security threats to which web applications are subject. SQL injection and Cross-Site Scripting (XSS) are common examples of these injection threats. A new addition to the OWASP Top Ten report of 2021 under the Injection category is Cross-Site Scripting. Attacks of this nature have the potential to undermine systems,

cause data leaks, breaches, and unauthorized access to private information. To preserve data security and protect system functionality, injection attack prevention and detection are essential. Sophisticated security mechanisms and input validation are two strong countermeasures that are necessary to reduce the hazards brought on by these sneaky attackers. These countermeasures can be implemented with the help of machine learning algorithms. There are different supervised and unsupervised learning algorithms which can be used to achieve these goals, but it is also important to remember that the efficiency and accuracy of algorithms is completely dependent on the data is collected to train and test these algorithms. One such algorithm is Multinomial Naïve Bayes (MNB), which is a very well-liked and effective machine learning technique. It is frequently applied to text categorization jobs that require handling discrete data, such as document word counts. This algorithm is used to detect the injections received from user-end. The possible injections are stored in an allocated database. The purpose of this stored injection incidences (historical data) is to understand intentions of the attackers and patterns followed by them. This stored data will help us to research and develop more effective counter measures to block such type of attacks.

2 Related Works

The following studies have been thoroughly summarized in this section, which served as inspiration for the creation of this application.

[1] “Using Data Mining Techniques for the Detection of SQL Injection Attacks on Database Systems” suggests the use of data mining techniques to identify SQL injection attacks in database systems. By inserting malicious SQL code into online application input fields, attackers can change or extract data, making SQL injection attacks a serious security risk. It’s common for traditional detection techniques to miss new assault patterns. The authors suggest using data mining algorithms to analyze web application traffic data, such as association rule mining, classification, and clustering, in order to address this. The outcomes of experiments show how well data mining methods work to find unusual patterns that point to SQL injection vulnerabilities.

[2] “SQL Injection Detection Using Machine Learning Techniques” outlines a method for utilizing machine learning techniques to identify SQL injection threats in online applications. The usefulness of different machine learning algorithms, such as decision trees, support vector machines, and neural networks, in detecting SQL injection vulnerabilities is examined by the writers. To train the machine learning models and categorize requests as benign or malicious, they examine features taken from HTTP request parameters and response headers. The performance of the suggested method is evaluated experimentally using real-world datasets to gauge its accuracy, precision, recall, and false positive rates. The study offers useful insights for cybersecurity academics and practitioners and emphasizes the potential of machine learning approaches for improving the security of online applications against SQL injection attacks.

[3] “Detection of SQL Injection Attack Using Machine Learning Techniques: A Systematic Literature Review” provides a through analysis of the literature on machine learning approaches for SQL injection attack detection. To give readers an understanding of the present status of this field’s research, the writers methodically examine and

summarize the results of a variety of investigations. They look at different datasets, evaluation metrics, machine learning algorithms, and SQL injection detection problems. The review provides practitioners and scholars in the subject of cybersecurity with significant insights by highlighting common pattern, strengths, and limits of various approaches. The work advances our understanding of SQL injection detection techniques by synthesizing and analyzing a wide range of literature it also offers recommendations for future research aimed at enhancing the efficacy and efficiency of detection mechanisms against this pervasive cyber threat.

[4] “Review of SQL injection attacks: Detection, to enhance the security of a website from client-side attacks” gives a thorough rundown of SQL injection threats and how to identify them in order to improve website security against client-side attacks. The authors provide a thorough analysis of the frequency, consequences, and several methods for detecting SQL injection attacks, such as hybrid, anomaly-based, and signature-based strategies. They talk about new developments in SQL injection detection, including evasion strategies and polymorphic attacks. The study provides insights into current research trends and recommendations for enhancing website security, while highlighting the significance of efficient detection techniques to reduce the risks caused by SQL injection attacks. All things considered, the study is a useful tool for academics, industry experts, and security specialists who want to comprehend and counter the risk of SQL injection attacks in online applications.

[5] “Ensemble Machine Learning Approaches for Detection of SQL Injection Attack” investigates the use of ensemble machine learning methods to web application SQL injection attack detection. In order to counteract changing assault patterns, Farooq talks about the shortcomings of conventional techniques and the requirement for stronger detection systems. Using real-world datasets, the research introduces many ensemble learning methods and assesses how well they detect SQL injection threats, such as Random Forest, AdaBoost, and Gradient Boosting. The findings show that ensemble methods perform better than single classifiers in terms of accuracy, precision, and recall when it comes to identifying fraudulent SQL queries. The study emphasizes the promise of ensemble machine learning approaches for practical deployment in real-world cybersecurity scenarios, while also highlighting their usefulness in strengthening web application security against SQL injection threats.

[6] “Cross Site Scripting (XSS) vulnerability detection using Machine Learning and Statistical Analysis” offers a cutting-edge method for identifying web application vulnerabilities that allow Cross-Site Scripting (XSS). The authors suggest using statistical analysis methods and machine learning algorithms to recognize and lessen XSS threats. The suggested method is examining input data from web applications to find patterns suggestive of XSS vulnerabilities. The approach’s efficacy in precisely identifying cross-site scripting vulnerabilities is demonstrated by the experimental findings. This study provides a strong and proactive method to improve web application security against XSS attacks by combining machine learning and statistical analysis. In order to reduce XSS vulnerabilities in online applications, researchers and cybersecurity experts can benefit greatly from the study’s insightful analysis and useful suggestions.

[7] “MFXSS: An effective XSS vulnerability detection method in JavaScript based on multi-feature model” offers MFXSS, a cutting-edge technique for finding JavaScript

code that has Cross-Site Scripting (XSS) vulnerabilities. In order to precisely detect XSS vulnerabilities, the authors present a multi-feature model that combines a variety of code aspects, such as static code features and dynamic executions information. MFXSS analyses JavaScript code snippets and classifies them as either not subject to XSS assaults or vulnerable by using machine learning techniques, more precisely a random forest classifier. The outcomes of the experiments show how well MFXSS works in precisely and recall rates in identifying XSS vulnerabilities. Through the integration of both static and dynamic code analysis approaches, MFXSS provides a strong defense against cross-site scripting (XSS) assaults for web applications. The paper provides insightful analysis and useful suggestions for researchers and cybersecurity experts looking to reduce XSS vulnerabilities in JavaScript-based web applications.

[8] “Cost-effective detection system of cross-site scripting attacks using hybrid learning approach” explains how to find Cross-Site Scripting (XSS) attacks in web applications at a reasonable cost. To effectively detect XSS vulnerabilities, the author suggests a hybrid learning strategy that blends rule-based methods and machine learning algorithms. To identify any XSS attacks, the system makes use of both dynamic analysis of user inputs and static analysis of web application source code. The suggested solution minimizes false positives while achieving high accuracy in finding XSS vulnerabilities through the integration of different detection techniques. The hybrid learning approach is appropriate for use in real-world scenarios since it provides an affordable means of strengthening web application security against cross-site scripting (XSS) threats. The study benefits cybersecurity experts and academics trying to reduce XSS vulnerabilities by providing insightful information about how to create effective and efficient XSS detection systems.

[9] “Detection of Cross-Site Scripting (XSS) attacks using machine learning techniques: A Review” offers a thorough analysis of the literature on the detection of Cross-Site Scripting (XSS) threats in web applications through the use of machine learning techniques. The writers conduct a thorough analysis of the body of literature to pinpoint developments, obstacles, and patterns in XSS detection techniques. They go over several machine learning algorithms, methods for extracting features, and assessment measures that are frequently used in XSS detection studies. The study also examines the merits and drawbacks of various strategies, highlighting new developments and potential areas for further investigation. The document provides insightful analysis for cybersecurity researchers and practitioners, with the goal of enhancing the efficacy and efficiency of XSS detection methods in online applications, by synthesizing and summarizing a broad range of investigations.

[10] “Machine and Deep Learning-based XSS Detection Approaches: A Systematic Literature Review” offers a thorough analysis of the literature on deep learning and machine learning techniques for identifying Cross-Site Scripting (XSS) attacks. The writers carefully review the literature to pinpoint patterns, developments, and difficulties in XSS detection strategies that make use of deep learning and machine learning approaches. They go over numerous algorithms, feature extraction techniques, datasets, and assessment criteria that are frequently used in XSS detection studies. The usefulness, drawbacks, and potential avenues for further research are also covered in the report. The review provides helpful insights for cybersecurity researchers and practitioners by

combining and summarizing a wide range of studies, with the goal of improving the efficacy and efficiency of XSS detection methods in online applications (Table 1).

Table 1. Summary of Related Works

Name of the Paper	Advantages	Disadvantages
[1] Using Data Mining Techniques for the Detection of SQL Injection Attacks on Database Systems	In order to detect SQL injection threats, the article employs sophisticated data mining algorithms, which offers a unique and potent method of improving database security	The approach's efficacy could be restricted by the dataset's representativeness, which could have an impact on how well the model performs in actual situations
[2] SQL Injection Detection Using Machine Learning Techniques	In comparison to conventional techniques, the paper's machine learning-based dynamic and adaptive solution to SQL injection detection greatly improves accuracy and efficiency	The models might not have undergone enough real-world testing, which could restrict their efficacy and dependability in a range of real-world situations
[3] Detection of SQL Injection Attack Using Machine Learning Techniques: A Systematic Literature Review	The study offers a comprehensive and methodical evaluation of the machine learning methods currently in use for recognising SQL injection attacks, providing insightful analysis and pointing out areas in which further research is needed	The paper's immediate practical impact may be limited because it primarily reviews previous research rather than introducing brand-new techniques or experimental results
[4] Review of SQL injection attacks: Detection, to enhance the security of a website from client-side attacks	In order to provide useful insights to guard against client-side assaults, the article reviews SQL injection detection techniques with an emphasis on improving website security	The paper's main contribution to the area may be limited because it only analyses current detection approaches without offering any new methodologies or experimental data
[5] Ensemble Machine Learning Approaches for Detection of SQL Injection Attack	The use of ensemble machine learning techniques, which integrate several models to increase the precision and resilience of SQL injection attack detection, is highlighted in the research	The computing demands and resource consumption of ensemble methods may make them impractical for organisations with inadequate computational resources

(continued)

Table 1. (continued)

Name of the Paper	Advantages	Disadvantages
[6] Cross Site Scripting (XSS) vulnerability detection using Machine Learning and Statistical Analysis	The study offers a strong method for web security by enhancing the detection of Cross-Site Scripting (XSS) vulnerabilities through the use of machine learning and statistical analysis	Combining statistical analysis and machine learning could lead to a difficult implementation process that needs a lot of knowledge and resources
[7] MFXSS: An effective XSS vulnerability detection method in JavaScript based on multi-feature model	The multi-feature methodology offers a more dependable security solution by considerably increasing the accuracy of XSS vulnerability detection in JavaScript	The multi-feature model's implementation can be difficult and resource-intensive, requiring significant processing power and highly developed technical know-how
[8] Cost-effective detection system of cross-site scripting attacks using hybrid learning approach	For businesses on a tight budget, the hybrid learning methodology provides an affordable means of identifying cross-site scripting (XSS) assaults	In certain situations, the system's overall effectiveness may be compromised by trade-offs between detection speed and accuracy brought about by the emphasis on cost-efficiency
[9] Detection of Cross-Site Scripting (XSS) attacks using machine learning techniques: A Review	The study offers a comprehensive overview of machine learning approaches for XSS attack detection, highlighting areas for further research and providing insightful information about existing approaches	Its emphasis on evaluating the body of prior research rather than offering brand-new experimental findings or methodology may restrict its immediate applicability
[10] Machine and Deep Learning-based XSS Detection Approaches: A Systematic Literature Review	The paper provides an indepth analysis of deep learning and machine learning-based strategies for XSS detection, offering insights into the most recent developments and techniques in the field	Its novelty and practical application may be limited because it largely concentrates on synthesising previous research without bringing fresh empirical discoveries or creative detection techniques

3 Proposed System

The proposed system aims to enhance cybersecurity measures by developing analysis modules that leverage machine learning (ML) techniques to detect SQL and XSS intrusions in web applications. The development process involves several key steps: collecting SQL/XSS injection payloads from various online sources to create a comprehensive dataset, thoroughly pre-processing the data to ensure cleanliness by handling

missing values and removing duplicates, and training the Multinomial Naïve Bayes (MNB) model to detect SQL and XSS injections. The trained model will then undergo extensive testing on distinct datasets to evaluate its accuracy and performance. Additionally, a Restful API will be developed to facilitate the integration of the ML models into web applications for real-time injection attack detection. A web application will also be created to provide clients with statistics and trends on injection attacks, including information on the frequency and severity of attacks over time. Furthermore, the system will generate analytical reports based on injection attack detections, offering detailed insights and patterns for further analysis and decision-making. Overall, the proposed system aims to strengthen cybersecurity defenses and provide valuable intelligence for effective threat mitigation (Fig. 1).

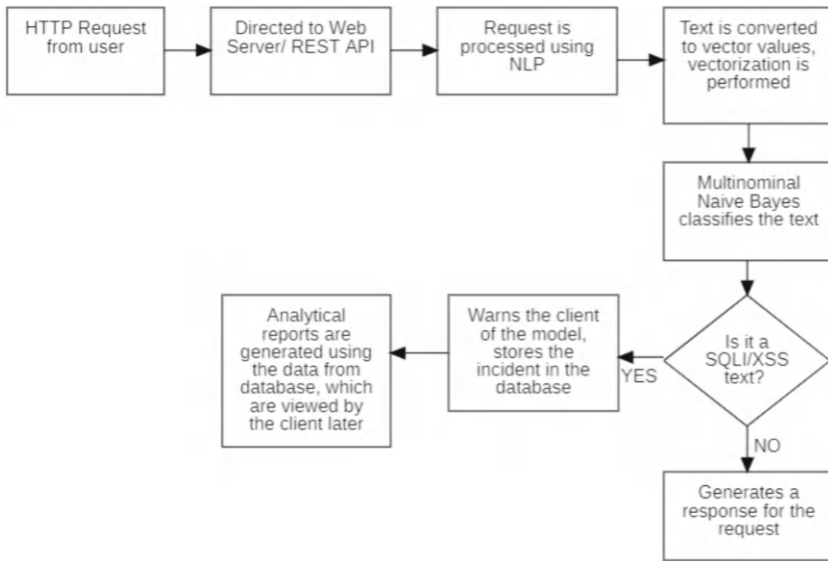


Fig. 1. Dataflow Diagram of the proposed system

4 Methodology

A project has been constructed on the proposed system, and the modules it contains are listed below:

4.1 Developing Analyzing Modules (ML)

The first step of the investigation is gathering SQLIA and XSS injection payloads from different websites; these datasets have 3242 and 7736 items, respectively. Preprocessing methods for Natural Language Processing (NLP) are applied to these datasets, including

tokenization, count vectorization, stop word removal, and n-gram synthesis. Unstructured data is transformed into structured formats by this preprocessing so that ML models can be trained on it. These datasets are used to train models using supervised learning methods like Support Vector Machines (SVMs) and Naïve Bayes. According to the classification report, evaluation using the scikit-learn metrics package shows good results with an accuracy score of 90% for SQLIA and 99% for XSS detection using Naïve Bayes.

4.2 Developing a REST API and Attaching ML Model

A Flask-based RESTful API is created, with endpoints for location information retrieval (“get_location/city” GET endpoint) and attack detection injection (“check_injection” POST endpoint). In order to enable the smooth integration of ML models for real-time threat detection and response, these endpoints manage JSON requests and answers.

4.3 Developing a Web Application to Provide Injection Attack Statistics and Trends

Clients can obtain reports on possible injection attacks arranged by industry and operational year through a web application designed specifically for the Indian context. Through access limitations, clients can read predicted reports and trend analyses pertinent to their particular industry, guaranteeing customised insights while protecting the confidentiality of their data.

4.4 Analytical Reports

The Prophet and Plotly libraries are used to build analytical reports that include visualisations like bar graphs for attack statistics and Prophet for prediction analytics. Important metrics, such as mean figures and Root Mean Square Error (RMSE), offer information about the predicted accuracy and performance of the model, assisting stakeholders in making well-informed decisions based on solid analytical results.

4.5 Web Page for Testing the REST API

The capability of the REST API for identifying invasive text statements can be tested and demonstrated on a dedicated webpage. Users may instantly tell if a text contains SQLIA or XSS vulnerabilities by entering their IP address, the text’s content, and the city that the text is related with. The database logs detected incidents in order to facilitate forensic analysis and incident response plans.

5 Implementation and Results

Multinomial Naive Bayes (MNB) has been used in the implementation of the aforementioned modules to identify potential injection threats. It has been noted that a given machine learning algorithm’s performance depends only on the data used to train the model. Thus, Multinomial Naïve Bayes’ performance in this instance turned out to be ideal (Fig. 2).

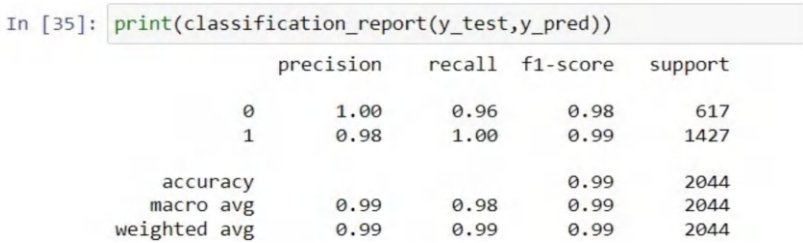


Fig. 2. Classification report on XSS detection model

Visual representations are included in analytical reports to improve understanding of the data. Plotly provides interactive and easily comprehensible tools, including scatter plots, box plots, heat maps, histograms, and bar graphs. To display analytical reports for this application, choropleth maps and bar graphs are used. Furthermore, time series graphs produced with the Prophet library are used in predictive analytics to predict future possible assaults. It is significant to remember that the graphs below were created with restricted access to real-time data.

6 Conclusion and Future Scope

The effectiveness of combining machine learning approaches with web technologies to improve SQLIA and XSS attack detection and mitigation in web applications is demonstrated by this study. Together, the built components, web application, API, and analytical tools make up a cybersecurity framework that is both responsive and proactive. Predictive analytics adds a proactive layer that makes it possible to take preventative action against dangers that are always changing. Going future, there are chances to improve the model even more. These include adding user behavior analysis to strengthen organizational security and broadening the model’s coverage of different injection threats. Using real-time reaction systems and integrating with threat intelligence streams are two further ways to improve proactive defense capabilities. Working together with industry stakeholders to collect a variety of information and investigate the use of blockchain technology for immutable logging could improve our defenses against sophisticated injection attacks. Our work adds to the continuing conversation about web application security by offering insightful analysis and practical tips for reducing cyberthreats in a constantly changing environment.

References

1. Añasco Loor, C., Morocho, K., Hallo, M.: Use of data mining techniques for detection of SQL injection attacks in database systems. *Polytech. Mag.* **51**(2), 19–28 (2023)

2. Gazala Akhtar, C., Harshini Reddy, M., Keerthana, P., Likitha, P., Nithya Sri, S.: SQL injection detection using machine learning techniques. *J. Crit. Rev.* **10**(4), 2394–5125 (2023)

3. Alghawazi, M., Alghazzawi, D., Alarifi, S.: Detection of SQL injection attack using machine learning techniques: a systematic literature review. *J. Cybersecur. Privacy* **2**(4), 764–777 (2022)

4. AL-Malikia, M.H.A., Jasima, M.N., AL-Maliki, M., Jasim, M.: Review of SQL injection attacks: detection, to enhance the security of the website from client-side attacks. *Int. J. Nonlinear Anal. Appl.* **13**(1), 3773–3782 (2022)
5. Umar, F.: Ensemble machine learning approaches for detection of SQL injection attack. *Tehnički Glasnik* **15**, 112–120 (2021)
6. Harish Kumar, J., Godwin Ponsam, J.J.: Cross site scripting (XSS) vulnerability detection using machine learning and statistical analysis. In: 2023 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, pp. 1–9 (2023)
7. Liu, Z., Fang, Y., Huang, C., Xu, Y.: “ MFXSS: an effective XSS vulnerability detection method in JavaScript based on multi-feature model. *Comput. Secur.* **124**, 103015 (2023)
8. Al-Haija, Q.A.: Cost-effective detection system of cross-site scripting attacks using hybrid learning approach. *Res. Eng.* **19**(2023), 101266 (2023)
9. Kaur, J., Garg, U., Bathla, G.: Detection of cross-site scripting (XSS) attacks using machine learning techniques: a review. *Artif. Intell. Rev.* **56**, 1–45 (2023)
10. Thajeel, I.K., Samsudin, K., Hashim, S.J., Hashim, F.: Machine and deep learning-based XSS detection approaches: a systematic literature review. *J. King Saud Univ. Comput. Inf. Sci.* **35**(7), 101628 (2023)
11. Thirumoorthy, P., Bhuvaneshwari, K.S., Kamalanathan, C., Sunita, P., Prabhu, E., Maheswaran, S.: Improved key agreement based Kerberos protocol for M-health security. *Comput. Syst. Sci. Eng.* **42**(2), 577 (2022)
12. Kumar, P., Kumar, S.V.: DDoS attack prediction system using machine learning algorithms. In: Tuba, M., Akashe, S., Joshi, A. (eds.) *ICT Systems and Sustainability. ICT4SD 2023. Lecture Notes in Networks and Systems*, vol. 765. Springer, Singapore (2023)
13. Ubayadhullah, F., Monish, G.S., Vassanth, S.S., Dhimantvanu, S.: Traceability of pharmaceutical products using blockchain. In: 2023 Eighth International Conference on Science Technology Engineering and Mathematics (2023)



Comparative Study of Vision-based Deep Learning Techniques for Human Action Recognition in Videos

R. Divya Rani^(✉) and C. J. Prabhakar

Department of Computer Science, Kuvempu University, Shimoga 577451, India
divyarsbg@gmail.com, psajjan@yahoo.com

Abstract. The main goal of Human Action Recognition systems is to automatically identify and analyze the specific actions performed by human beings in videos. HAR is extensively used in many application areas, such as video storage and retrieval, robotics, intelligent video surveillance, and healthcare. The most challenging part of the action recognition problem is to process videos that contain actions. Other challenges associated with action recognition are cluttered backgrounds, illumination variations, occlusion, and intra- and inter-class similarities. Recently, Deep learning-based techniques have given the highest accuracy for recognizing human actions. In this paper, we discuss many deep learning-based techniques and pre-trained models proposed for recognizing human actions present in videos. Further, the performance comparison of various deep learning-based techniques with respect to obtained recognition accuracy is carried out. We also presented a brief overview of popular benchmark action datasets available for human action recognition. It is evident from the results produced by deep learning-based techniques that these methods are delivering excellent results for human action recognition and are capable of handling significant challenges associated with human action recognition.

Keywords: Human Action Recognition · HAR · Deep Learning · CNN · 3D CNN

1 Introduction

Today, the Internet is the fastest communicating media and all individuals are aware of using the Internet. Due to the ease of using the Internet, people are crazy at uploading photos and reels on social networks resulting in a massive amount of video data. The process of organising and retrieving the video clips present on the Internet according to the content of the video is a tedious task. It can be automatized by the concept of Human Action Recognition (HAR) by analysing the specific actions performed by the actor present in the action video clip. With similar kinds of applications, HAR has drawn the interest of many

researchers and has enormous applicability in real-world problems such as abnormal detection, intelligent video surveillance, robotics, healthcare, video storage and retrieval, intelligent driving, and sports action analysis. The difficulty associated with action recognition in videos is the processing of videos of varying length. Rather than this action recognition problem also faces some challenges present in action videos such as illumination changes, actions with occlusion, presence of cluttered background, viewpoint variation, camera motion and intra-class and inter-class similarity.

Numerous techniques have been put forth by researchers for recognizing human actions present in images and videos. Traditional handcrafted features-based techniques for HAR extracted visual features from the image or video sequences, and extracted features are represented by standard representation models. A generic classifier model is employed to recognize human actions. In contrast, the techniques based on deep learning extract essential features automatically from raw image or video frames and classifies accordingly. Deep learning-based techniques are extremely employed these days due to their outstanding performance in the field of computer vision. A huge number of action datasets are required to train deep learning network models compared to traditional HAR techniques.

Deep learning models such as CNN, RNN, LSTM, DBN, DBMs, and RBMs are widely used models for HAR. A thorough review of the literature reveals that numerous computer vision-based techniques have been introduced for recognizing human actions present in videos using deep learning models. The popular CNN models are successful in recognizing human actions present in still images by collecting deep spatial features from the input image, but they fail to extract temporal features. To extract temporal features along with spatial features, many advanced deep architectures such as multi-stream techniques and hybrid models are introduced for recognizing human actions and to effectively overcome the challenges associated with action recognition.

For experimentation, the most frequently used popular standard action recognition datasets are the KTH dataset [1], UCF-101 dataset [2], and HMDB-51 dataset [3]. All these datasets consist of video clips corresponding to many action classes performed by many actors captured by static or motion cameras with different environmental conditions. In order to assess the efficiency of the action recognition model, different evaluation metrics are considered, such as recognition accuracy, precision, recall and F-measure, which are computed from the entries of the confusion matrix.

In this paper, we discuss various deep learning-based techniques proposed to recognize human actions in videos. The performance comparison of deep learning-based techniques for HAR based on the obtained recognition accuracy for the benchmark action datasets is carried out in this paper. The paper is organized as follows. Introduction to HAR is given in Sect. 1. In Sect. 2, three different categories of deep learning-based techniques and pre-trained deep learning models for HAR are discussed. The comparative analysis of deep learning-based techniques proposed for HAR is presented in Sect. 3. In Sect. 4, a discussion of

various action recognition techniques based on deep learning models is presented. Finally, the conclusion is specified in Sect. 5.

2 Deep Learning-Based Techniques for Human Action Recognition

With the rapid advancement of deep learning techniques in recent years, a notable change has been observed in computer vision research. Although hand-crafted features-based techniques have been quite successful, they still need a lot of human effort and domain expert knowledge for developing efficient methods for extracting features. Moreover, they usually exhibit poor performance on huge datasets. Over the past few years, Deep learning-based techniques have attained more consideration due to their capability of automatically extracting potential features from the raw input data and that can be generalized very effectively. This avoids the task of extracting traditional hand-crafted feature detectors and feature descriptors used in conventional techniques for representing human actions. These techniques employ multi-layered computational models and trainable feature extractors for action recognition. Techniques based on deep learning can be categorized into three groups: supervised (discriminative) models, unsupervised (generative) models, and hybrid models.

2.1 Supervised (Discriminative) Models for HAR

Supervised models classify the raw input data into numerous output categories using a hierarchical learning model with many hidden layers. The most popular supervised learning-based neural network models is the CNN. CNN models have shown excellent performance for various recognition tasks, such as image classification, handwritten digit recognition, and human action recognition. The significant benefit of CNN is its capability to learn knowledge directly from the input raw data. As a result, the recognition process progresses more quickly, making it easier to extract discriminative features that accurately represent the input. The CNN network architecture converts input data into a collection of classes using a hierarchical structure with more hidden layers. Discriminative features are extracted by the convolutional layer of the CNN model, and the dimensionality of the feature vector is reduced by the max-pooling layer by combining semantically related features into a single feature. The last layer of the CNN model is the softmax layer, which serves as a classifier. CNN model is prone to overfitting, hence caution should be taken while training. Regularization functions such as ReLUs and Dropout layers are also utilized to prevent overfitting problems and to speed up the learning process of CNNs.

In order to recognize human actions present in videos, Kim et al. [4] introduced an architecture in 2007 that combined a CNN model with modifications with the Weighted_Fuzzy_Min-Max_neural_network (WFMM). Yu et al. [5] introduced a pose based model called the Stratified Pooling-based CNN (SP-CNN). The crucial indicators for characterizing the type of action performed by the actor are Pose, poselet and other semantic-based features.

However, video is considered as a 3D spatio-temporal signal, whereas CNNs have mostly been used with two-dimensional data (2D-CNN), where these networks only use spatial dimensions to extract features, and thus making them unsuitable for video data due to the absence of temporal information. For video data, the temporal domain is considered as a third dimension. Thus, the primary challenge in expanding CNN from image data to video data is the use of sequential or motion information. The methods which considered temporal information for recognizing human actions are categorized into two categories: 1) 3D convolutional neural networks, and 2) Multi-stream networks.

1) 3D Convolution Neural Networks (3D CNN). To extract implicit motion features from input video sequences, 3D convolution networks have been used. Ji et al. [6] presented a deep learning model to leverage the temporal information of human motion, called a 3D CNN framework for HAR. The 3D CNN network architecture consists of five hardwired kernels, such as grey, x_gradient, y_gradient, x_optflow, and y_optflow, which together produce 33 feature maps. This model then repeatedly performs 3D convolution and sub-sampling operations and employs a fully connected layer to produce a feature vector of size 128-dimensions for the classification of actions. The convolution phases of this architecture make use of 3D kernels. Figure 1 illustrates the 3D CNN architecture for HAR introduced by Ji et al. [6].

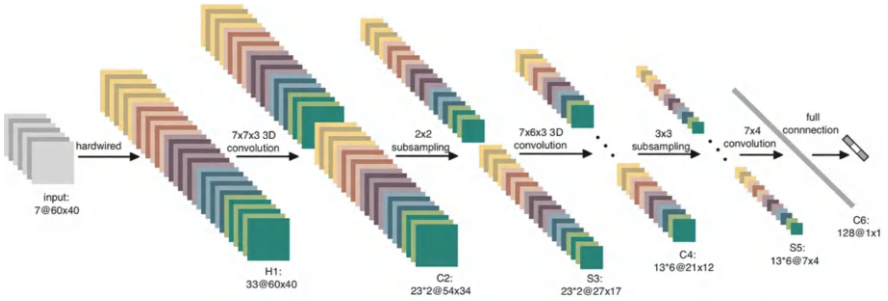


Fig. 1. The 3D CNN architecture for HAR proposed by Ji et al. [6].

A 3D kernel is convolved to an input video sequence in order to perform a 3D convolution. 3D convolutions are performed across successive frames to extract discriminative features from both spatial and temporal dimensions. The value of the 3D convolution operation is calculated at the position of the pixel (x, y, z) on the k^{th} feature map of the i^{th} layer using Eq. (1):

$$V_{ik}^{xyz} = \varphi(b_{ik} + \sum_p \sum_{h=0}^{H_i-1} \sum_{b=0}^{B_i-1} \sum_{t=0}^{T_i-1} W_{ikp}^{hbt} V_{(i-1)p}^{(x+h)(y+b)(z+t)}). \quad (1)$$

where, (x, y, z) is the pixel coordinates of the feature map, i means convolution layer, k and p means feature maps, φ is any non-linear activation function, W_{ikp}^{hbt} is the $(h, b, t)^{th}$ kernel weight connected to the p^{th} feature map of the previous layer. b_{ik} is the bias. H , B , and T are the kernel's height, width and temporal dimension [6].

Tran et al. [7] thoroughly evaluated the 3D CNN architecture and demonstrated that it performs better than the 2D CNN model in representing motion information of the human being. The authors found that $3 \times 3 \times 3$ size is the ideal kernel length for 3D CNN and was the optimum option for spatio-temporal features. However, 3D ConvNets for action recognition only have up to 8 layers, making them somewhat shallow. The extended version of 3D ConvNet is C3D, which learns from huge datasets. Depending on the available computing power and computer memory, the C3D network architecture consists of many layers compared to 3D CNN. Results demonstrated that the C3D approach with a linear classifier performed better than other video analysis benchmarks, including HAR.

Tran et al. [8] built Res3D, which is a ResNet18 style 3D architecture, it outperforms C3D and it is twice smaller than C3D and two times faster in run-time. Another approach was put forth by Qiu et al. [9] to construct a deep 3D_ConvNet, that employs a combination of one $1 \times 3 \times 3$ convolutional layer and one $3 \times 1 \times 1$ convolution to replace a standard 3D convolution. Very deep two-stream CNN networks for recognizing human actions were introduced based on the VGG-16 (Thirteen convolutional layers and three fully linked layers make up VGGNet CNN) and the GoogLeNet with a 22-layer network.

One drawback of 3D ConvNets is that they frequently take into account relatively very short time intervals, such as 16 frames, and as a result they are unable to capture long-sequence temporal information.

2) Multi-Stream Networks. To describe appearance as well as motion information in action videos, multi-stream networks are introduced which consider multiple convolutional networks. For HAR, Simonyan et al. [10] introduced a two-stream framework, as depicted in Fig. 2. A spatial ConvNet and a temporal ConvNet are two distinct streams in this two-stream model. From still images, the spatial stream learns actions, whereas the temporal stream learns motion information from the input video clip. CNN model is employed to build each stream in this architecture. The results obtained at the softmax layer of both streams are integrated by the late fusion method to obtain the final result. The experimental findings demonstrated that multi-stream networks outperformed handcrafted feature-based approaches.

However, this multi-stream network architecture requires extra hardware requirements. An enhancement to this architecture was suggested by Wang et al. [11], by using a two-stream network to produce multiscale convolutional feature maps. These feature maps were then combined with the detected trajectories to produce ConvNet responses. This technique encodes the extracted deep features into effective feature descriptors restricted by sampling trajectories, and these

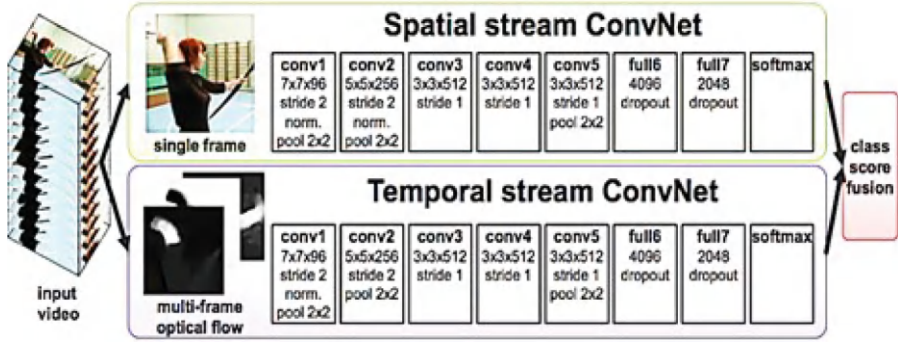


Fig. 2. Spatio-temporal CNN stream for HAR by Simonyan et al. [10].

descriptors are referred to as TDD (Trajectory_Pooled Deep_Convolutional Descriptors). For the spatio-temporal fusion of two-stream networks, Feichtenhofer et al. [12] introduced an architecture based on CNNs that made use of the deep CNN network VGG-M-2048 and the extremely deep neural network VGG-16. The adoption of deep neural models enhance the performance, as demonstrated by a comparison of performance between VGG-16 and VGG-M-2048 network architectures on the HMDB-51 and UCF-101 datasets.

Wu et al. [13] suggested a different approach to represent motion information in videos for recognizing actions using RNNs. For every input video frame, CNN extracts discriminative features and were subsequently passed into an RNN model. The primary benefit of the RNN network lies in its capacity to process sequential inputs, as it generates a single network copy for every sequence. The neurons in hidden layers of the RNN are interconnected between replicas and share weights with others. Using CNNs and optical flow, local motion information from video may be obtained. Alternatively, global motion information can be obtained using the RNN.

LSTM is a new kind of recurrent module that was proposed to handle sequential data. Ullah et al. [14] presented an architecture for action recognition that used a CNN and a deep bidirectional LSTM network model to interpret video input. Cob-Parro et al. [15] presented a model for HAR based on LSTM and edge computing. Initially, the people detection is performed by MobileNetV2-SSD, the feature vector generated from SSD is passed as input for LSTM. LSTM defines the temporal component of the action image. LSTMs are effective in addressing vision problems because they are easy to fine-tune from beginning to end. Additionally, LSTMs can process sequential data and are not limited to inputs or outputs of a particular length.

2.2 Unsupervised (Generative) Models for HAR

The unsupervised (generative) models are widely used by machine learning researchers and practitioners. These models can be used in the absence of labeled

data because they do not need target labels when training the model. Unsupervised models include DBNs, DBMs, RBMs, and regularized auto-encoders. The parameters were initially learned using an unsupervised training technique and then fine-tuned using back-propagation in a supervised approach. Le et al. [16] presented a model for recognizing human actions from video data based on unsupervised learning. The authors employed a separate subspace analysis approach to extract spatio-temporal features. Hasan et al. [17] introduced an unsupervised deep learning-based neural network model to learn continuously from unlabeled video streams. Foggia et al. [18] presented a framework to train DBNs using RBMs for the recognition of human actions. The results obtained by the experiments on two popular datasets showed that the method proposed by Foggia et al. outperformed hand-crafted feature-based approaches.

Furthermore, for recognizing human actions, DBNs-based techniques have been used to extract the features present in the unconstrained videos. Recent developments in Generative Adversarial Networks (GANs) were beneficial for both human action classification and prediction in videos. GAN networks typically have a generator and a discriminator component. The generator produces data similar to the training data, whereas the discriminator module maximizes the probability of the generative model's correct labels and training example samples.

Unsupervised or generative learning models have attracted the interest of many researchers and have been greatly influenced in the deep learning discipline. However, the unsupervised models have been somewhat overwhelmed by the rapid development of supervised learning techniques for object recognition and classification based on CNNs.

2.3 Hybrid Models for HAR

Hybrid models combine supervised and unsupervised models to acquire the benefits and performance of both models. To integrate temporal information into spatial models, recurrent layers, such as LSTMs, are placed on top of CNNs to produce hybrid networks. These hybrid models combine the strengths of CNNs with LSTMs, resulting best results for capturing spatial information, temporal orderings, and long-term relationships.

An hybrid deep learning model was presented by Donahue et al. [19] for HAR by employing a long-term recurrent CNN (LRCN). The raw input video clip and optical flow data are passed as input for LRCN. Experimentation was carried out on the popular UCF-101 dataset, and the results indicated improved performance compared to the base architectures. The architecture of the LRCN network is shown in Fig. 3. Gowda [20] introduced a hybrid deep learning model to extract features and identify interesting features. DBN is utilized to extract motion features from static images. The DBN's output is fed into KPCA, which then sends it to CNN for action classification.

The state-of-the-art deep learning-based models require a huge training dataset to train the model. It takes more time to train deep learning models from start, this can be overcome by employing pre-trained models. Large

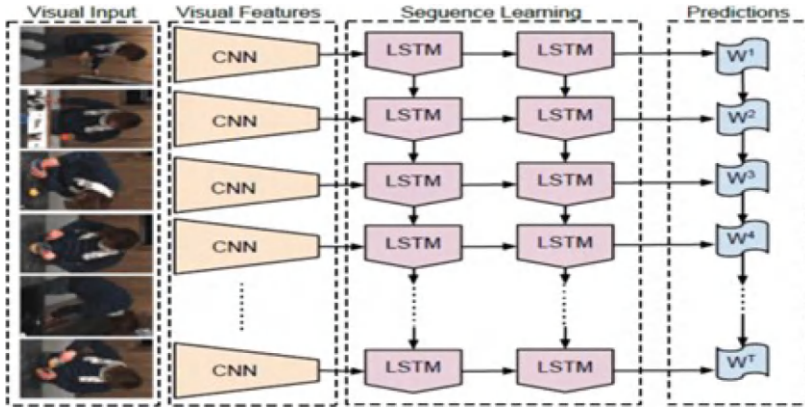


Fig. 3. LRCN network architecture proposed by Donahue et al. [19].

datasets are used to train pre-trained models, and these models are reused for different task thus by reducing the cost of training new deep learning models. Several pre-trained models are employed for HAR such as AlexNet, GoogLeNet, Inception v1/v2/v3/v4, ResNet 34/50/101/152, Inception-ResNet v1/v2, and EfficientNet B0/B7 pre-trained models. Abdelrazik et al. [21] proposed an algorithm for HAR that combines the Inception-ResNet-v2 and GoogLeNet models. The main architecture of the Inception-ResNet-v2 pre-trained model is shown in Fig. 4.

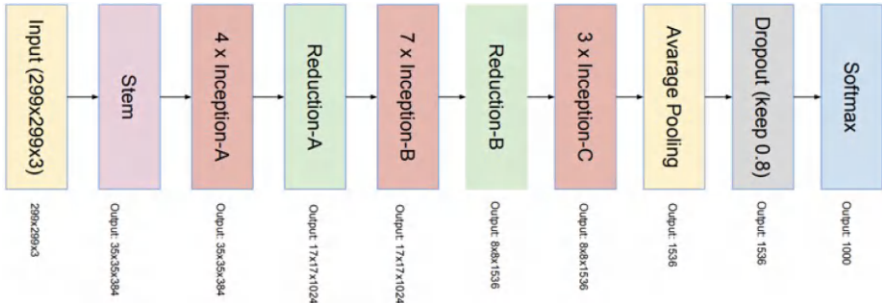


Fig. 4. The architecture of the Inception-ResNet-v2 pre-trained model [21].

3 Performance Comparison of Deep Learning-Based Techniques for HAR

We have conducted a brief performance comparison of deep learning-based techniques for recognizing human actions in videos carried out on the most frequently

used benchmark datasets such as the KTH, UCF-101, and HMDB-51 datasets. The videos in all these action datasets are captured in different environmental conditions, and each sample involves occlusion, cluttered background, view variations, motion camera, lightning conditions, etc. The sample action frames of the KTH, UCF-101, and HMDB-51 datasets are shown in Fig. 5.

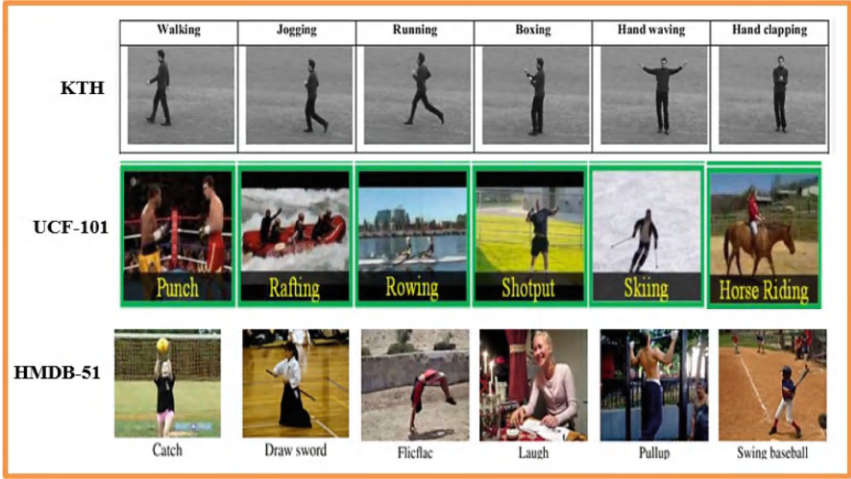


Fig. 5. Sample action frames of KTH, UCF-101, and HMDB-51 Datasets.

Initially, the input video frames of all datasets are converted to video frame sequences and resized to a standard size that is suitable for experimentation. For experimentation all these approaches employed a train test ratio of 70:30. The recognition results obtained for these techniques are described in Table 1. The recognition accuracies obtained for the deep learning-based techniques frequently surpass the currently used conventional techniques.

We can state that deep learning-based techniques are now the best option for action recognition tasks because they have the capability to learn straight from raw data and their ability to work with the wide variety of action classes. From Table 1, it is observed that deep learning-based human action recognition techniques achieved higher recognition accuracies in the KTH and UCF-101 datasets than on the HMDB-51 dataset, because this dataset is a huge dataset consisting of real-time videos and requires more training samples to obtain good recognition results.

Table 1. Performance comparison of the various deep learning-based techniques for HAR.

Authors	Year	Deep Learning Architecture	Dataset	Recognition Accuracy (%)
Ji et al. [6]	2013	Spatio-temporal 3D CNN	KTH	90.20
Hasan et al. [17]	2014	Sparse auto-encoder	KTH	96.60
Simonyan et al. [10]	2014	Two_stream_CNN	UCF-101 HMDB-51	88.00 59.40
Yue Hei N. et al. [22]	2015	CNN + LSTM	UCF-101	88.60
Sun L. et al. [23]	2015	Spatio-temporal CNN	UCF-101 HMDB-51	88.10 59.10
Tran et al. [7]	2015	Spatio-temporal_3D CNN	UCF-101	90.40
Tran et al. [8]	2015	C3D	UCF-101	82.30
Wang et al. [11]	2015	TDDs + iDTs	UCF-101	91.50
Mahasseni et al. [24]	2016	Spatio-temporal LSTM + CNN	UCF-101 HMDB-51	86.90 55.30
Park et al. [25]	2016	Deep_CNN	UCF-101	89.10
Shi et al. [26]	2016	RNN + Deep CNN	KTH UCF-101 HMDB-51	96.80 92.20 65.20
Feichtenhofer et al. [12]	2016	Two-stream + IDT	UCF-101 HMDB-51	92.50 65.40
Donahue et al. [19]	2016	LRCN (LSTM + CNN)	UCF-101	82.34
Yu S. et al. [5]	2017	SP-CNN	UCF-101 HMDB-51	91.60 74.70
Gowda [20]	2017	DBN + CNN	HMDB-51	80.48
Ullal et al. [14]	2019	Deep CNN	UCF-101 HMDB-51	94.33 70.33
Uddin et al. [27]	2019	Inception Resnet – v2	UCF-101	92.40
Dai C. et al. [28]	2020	Two-stream Attention based LSTM	UCF-101 HMDB-51	89.60 66.30
Abdelrazik et al. [21]	2023	GoogLeNet	UCF-101 HMDB-51	92.23 64.53
Abdelrazik et al. [21]	2023	Inception	UCF-101 HMDB-51	95.51 67.09

4 Discussion

Deep learning technology has been offered as a new avenue for improving action recognition abilities. The most significant property of deep learning techniques is their capability to learn features from unprocessed data. This has eliminated the

requirement for hand-crafted feature detectors and feature descriptors. In this paper, three categories of deep learning-based HAR techniques are discussed.

Supervised models are given higher priority for action recognition research due to their wide application areas and excellent implementation efficacy. It has attained competitive performance in several computer vision applications when compared to traditional techniques. CNN is the most popular supervised model, which is currently utilized in the majority of deep learning-based methods for directly extracting spatio-temporal features from videos without requiring any kind of pre-processing.

Almost all the present outstanding architectures have employed 3D convolutional filters in order to extract motion features. The underlying principle of CNN models allows them to operate directly on image data, integrating lower-level features to produce higher-level features. CNNs were used as a feature extractor as well as an integral component of other frameworks, in addition to their end-to-end functionality. However, there are still certain limitations to CNN-based approaches that must be taken into account. One of the drawbacks of CNN is the requirement of enormous volumes of data to train the models. In addition, high-performance computing equipment is required to facilitate calculations in a reasonable amount of time.

RNNs are combined with LSTM (LSTM-RNNs) to solve time-series problems. LSTM-RNNs can directly extract motion features from the input video and perform classification. Due to their ability to utilize all motion frames throughout history, LSTM-RNNs are successful at recognizing human action. However, most LSTM-RNN based models are unable to operate straight on raw input data. For instance, skeleton data must be pre-processed before being fed into LSTM-RNNs. For applications requiring action recognition, deep belief networks (DBNs) are another excellent option. Semi-supervised training is adopted for the DBNs using a few labeled data from a series of samples to categorize their inputs. Deep belief networks (DBNs) yield very promising results for action recognition tasks. The drawback of DBN networks is that they need handcrafted features for conversion of input data to an appropriate form.

Deep learning methods have outperformed for human action recognition. However, they require huge training datasets to train the model and high computational requirements. Training each model from scratch is extremely complex, hence different Transfer Learning-based methods are recommended pre-trained on massive amounts of data. These models reduced the training time to train the model from scratch. These pre-trained models can be employed to classify real time action and recognise complex actions.

5 Conclusion

In this paper, we have discussed deep learning-based techniques proposed for recognizing human actions in videos. With the detailed study and comparisons of the recognition accuracies obtained for the deep learning-based techniques, we find that deep learning is now the most effective method for classifying and

recognizing human actions in videos. The way we handle videos for HAR has been radically revolutionized by deep learning. Despite tremendous development in recent years, there exist numerous challenges with HAR in applying deep learning-based techniques for developing vision-based HAR systems and bringing their advantages to everyday life. We continue to look forward to develop new deep learning-based techniques that will boost the recognition performance of the system while reducing the computation costs and requirement of few labelled data by using more complicated and multi-view datasets.

Disclosure of Interests. The authors do not have any competing interests.

References

1. Schudt, C., Laptev, I., Caputo, B.: Recognizing human actions: a local SVM approach. In: 17th International Conference on Pattern Recognition, ICPR 2004, vol. 3, pp. 32–36 (2004)
2. Soomro, K., Zamir, A., Shah, M.: UCF101: a dataset of 101 human actions classes from videos in the wild (2012)
3. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: HMDB: a large video database for human motion recognition. In: 2011 International Conference on Computer Vision, pp. 2556–2563 (2011)
4. Kim, H.-J., Lee, J.S., Yang, H.-S.: Human action recognition using a modified convolutional neural network. In: Liu, D., Fei, S., Hou, Z., Zhang, H., Sun, C. (eds.) ISNN 2007. LNCS, vol. 4492, pp. 715–723. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72393-6_85
5. Yu, S., Cheng, Y., Su, S., Cai, G., Li, S.: Stratified pooling based deep convolutional neural networks for human action recognition. *Multimed. Tools Appl.* **76**(11), 13367–13382 (2016). <https://doi.org/10.1007/s11042-016-3768-5>
6. Ji, S., Xu, W., Yang, M., Yu, K.: 3D convolutional neural networks for human action recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(1), 221–231 (2013)
7. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3D convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 4489–4497. IEEE (2015)
8. Tran, D., Ray, J., Shou, Z., Chang, S., Paluri, M.: ConvNet architecture search for spatiotemporal feature learning. *arXiv preprint* (2017)
9. Qiu, Z., Yao, T., Mei, T.: Learning spatiotemporal representation with pseudo-3D residual network. In: Proceedings of the International Conference on Computer Vision (ICCV), pp. 5534–5542 (2017)
10. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: *Advances in Neural Information Processing Systems*, vol. 27 (2014)
11. Wang, L., Qiao, Y., Tang, X.: Action recognition with trajectory-pooled deep-convolutional descriptors. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4305–4314. IEEE (2015)
12. Feichtenhofer, C., Pinz, A., Zisserman, A.: Convolutional two-stream network fusion for video action recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1933–1941. IEEE (2016)

13. Wu, Z., Wang, X., Jiang, Y.G., Ye, H., Xue, X.: Modeling spatial-temporal clues in a hybrid deep learning framework for video classification. In: Proceedings of the 23rd ACM International Conference on Multimedia, pp. 461–470. ACM (2015)
14. Ullah, A., Ahmad, J., Muhammad, K., Sajjad, M., Baik, S.W.: Action recognition in video sequences using deep bi-directional LSTM with CNN features. *IEEE Access* **6**, 1155–1166 (2018)
15. Cob-Parro, A.C., Losada-Gutiérrez, C., Marrón-Romera, M., Gardel-Vicente, A., Bra-vo-Muñoz, I.: A new framework for deep learning video based Human Action Recognition on the edge. *Expert Syst. Appl.* **238**, 122–220 (2024)
16. Le, Q.V., Zou, W.Y., Yeung, S.Y., Ng, A.Y.: Learning hierarchical invariant spatio-temporal features for action recognition with in-dependent subspace analysis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3361–3368. IEEE (2011)
17. Hasan, M., Roy-Chowdhury, A.K.: Continuous learning of human activity models using deep nets. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8691, pp. 705–720. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10578-9_46
18. Foggia, P., Saggese, A., Strisciuglio, N., Vento, M.: Exploiting the deep learning paradigm for recognizing human actions. In: Proceedings of the 11th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), pp. 93–98. IEEE (2014)
19. Donahue, J., et al.: Long-term recurrent convolutional networks for visual recognition and description. *Clin. Orthop. Relat. Res.* abs/1411.4389 (2015)
20. Gowda, S.N.: Human activity recognition using combinatorial deep belief networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 1589–1594. IEEE (2017)
21. Abdelrazik, M.A., Zekry, A., Mohamed, W.A.: Efficient hybrid algorithm for human action recognition. *J. Image Graph.* **11**, 72–81 (2021)
22. Yue-Hei Ng, J., et al.: Beyond short snippets: deep networks for video classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4694–4702. IEEE (2015)
23. Sun, L., Jia, K., Yeung, D.Y., Shi, B.E.: Human action recognition using factorized spatio-temporal convolutional networks. *Santiago* 4597–4605 (2015)
24. Mahasseni, B., Todorovic, S.: Regularizing long short term memory with 3D human-skeleton sequences for action recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3054–3062. IEEE (2016)
25. Park, E., Han, X., Berg, T.L., Berg, A.C.: Combining multiple sources of knowledge in deep CNNs for action recognition. In: Proceedings of the of IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1–8. IEEE (2016)
26. Shi, Y., Tian, Y., Wang, Y., Huang, T.: Sequential deep trajectory descriptor for action recognition with three-stream CNN. *IEEE Trans. Multimed.* **19**(7), 1510–1520 (2017)
27. Uddin, Md.A., Lee, Y.-K.: Feature fusion of deep spatial features and handcrafted spatiotemporal features for human action recognition. *Sensors* **19**(7) (2019)
28. Dai, C., Liu, X., Lai, J.: Human action recognition using two-stream attention based LSTM networks. *Appl. Soft Comput.* **86** (2020)



Person Identification Using Improved Dilated Convolutional Network for Radar Signal Based Hand Gestures

Helen Victoria¹, Deepa Natesan¹(✉), Balamurugan Balusamy², Shilpa Bade-Gite³, Sushama⁴, and Alka Agnihotri⁵

¹ SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India
{helenvia, deepan3}@srmist.edu.in

² Associate Dean-Student Engagement, Shiv Nadar University, Greater Noida, Uttar Pradesh, India

³ CSE Department, Symbiosis Institute of Technology, Pune, India
shilpa.gite@sitpune.edu.in

⁴ Manipal University Jaipur, Jaipur, Rajasthan, India
sushama.tanwar@jaipur.manipal.edu

⁵ School of Business, Galgotias University, Greater Noida, Uttar Pradesh, India

Abstract. Hand Gesture based Person Identification has become a promising technology with a wide range of applications for Human Computer Interaction. Hand gestures are recognized by Kinect sensors, Gyro sensors and Inertial Sensors. We have used Frequency Modulated Continuous Wave Radar based hand gestures dataset of 6 people with four different gestures for identifying each Person. Previous works have identified hand gestures with Radar as its sensors, in this work we have proposed a dilated Convolution Neural Network Model for identifying the Person based on micro-Doppler hand gestures using a reduced number of parameters in contrast to existing state of art methods, that work with larger number of parameters for learning. We have observed that the proposed model has achieved an excellent recognition rate of 97.47% with the test data and 68% reduced parameters compared to benchmark and traditional CNN with minimal memory utilization and optimized number of computations of the Proposed Model.

Keywords: Human Computer Interaction · Radar Hand Gestures · Frequency Modulated Continuous Wave Radar · Dilated Convolutional Deep Neural Networks · Person Identification

1 Introduction

Human Computer Interaction has seen its meteoric rise since recent years across fields like Computer Science, Medical Science and Industrial Design [1]. This is mainly due to the improved interaction between the human and the user that resembles natural human to human interaction. Hand Gesture Recognition based human computer interaction

has flourished and emerged as an excellent solution for a broader range of applications like Device Control in consumer electronics, Smart Phones, Medical Devices and in Vehicle Control Mechanisms, rehabilitation [2]. Traditional Hand Gesture Based Image Acquisition and Recognition are mainly based on optical sensors, smart depth cameras [3, 4]. Yet these approaches are not environment friendly as varying climatic conditions has varied effects on the Gesture recognition rates [5]. Another factor that degrades hand-based gesture recognition is background noise which is inevitable [6]. Radar based smart sensors are said to be a better alternative to the existing optical sensors due to its distinct characteristics[7]. It is not affected by external environmental conditions and it also has evolved as a cost effective solution for Hand Gesture based recognition [8]. Google Soli, a ubiquitous device for Hand Gesture recognition using Multimeter Wave Radar for contactless interactions, was a breakthrough for Radar Based Hand Gesture recognition [9]. While Hand gesture related problems have been well approached by various researchers in recent times, a fascinating and yet less explored area is identifying the person based on radar hand gestures. Human Recognition based on micro doppler radar movements has evolved with applications like gait recognition, elderly monitoring, hospital monitoring systems, and defense [10, 11]. Ref. [12] proposed Natural Hand Gesture based Human identification and proved it as an effective solution. Machine.

Learning Algorithms like Support Vector Machine (SVM), K Nearest Neighbor (KNN), and Linear Discriminant Analysis (LDA) have been used and compared for their effectiveness in this work. Another work proposed an authentication model with common hand sign gestures obtained from different people as input in mobile devices [13].

1.1 State of Art Human Recognition Methods

Human recognition using ordinary cameras and depth cameras exhibits promising results in recent research. Depth cameras like Microsoft Kinect, Intel Real sense provide depth information besides the RGB information. Deep learning Approaches use CNN for obtaining discriminant features for human detection. Depth cameras can capture skeletal information of humans which is used for human movements and poses. There are some variants like pointNet processed by different representations of human movement in the cloud used for tasks like human recognition. Fusion of modalities is also done with different sensing modalities. Certain approaches use Biometric sensors to identify humans. Gait Methods uses human walk and patterns to identify the human. Another method to identify humans is by measuring the physical characteristics of the hand to identify the human. Medical-based modalities like electrocardiogram, Electroencephalogram, and iris can be used to identify the human under study. Even though different modalities were used, radar sensor has its unique characteristics for choosing it for human recognition.

The main aim of our work is to identify the person to whom the gesture is associated with. This is done by training and testing 4 different gestures of each person instead of a single gesture as in previous works. The person under study must perform one of the multiple gestures to get identified. The recognition rate is based on the correct classification of the person based on one of the chosen gesture images as input. Therefore, it can be formulated as an image classification problem. Here in this work, micro doppler signatures

obtained from radar signals are used as image inputs. Hand Gesture based Image Classification has been previously done with Machine Learning Algorithms like Decision Tree with 92% accuracy. Image classification tasks are handled well with the advancements in the field of Deep Learning. Radar based hand gesture recognition is done with Convolutional Neural Networks (CNN). Traditional Machine Learning Algorithms require features to be extracted for supervised learning. The classification accuracy depends on the effective extraction of features which needs a clear understanding of the problem domain as well. On the other hand, CNN does not require hand crafted features as the Network can extract features by itself. This is the main reason for many researchers to adopt this Network Model. CNN is a supervised training model where learning happens in the training phase using back propagation. Best Recognition is achieved when the CNN Architecture is deep with a large number of layers and parameters. Previous works have used optimization techniques for tuning the parameters in this kind of Black Box Approach. In specific, CNN computations have a lot of floating-point operations during each iteration. Hence CNN Architectures are getting computationally complex due to the depth and the number of parameters in the training data. To optimize the convolution operations and to minimize the number of parameters, in this work, we have used Dilated Convolutions. Dilated or atrous Convolutions were used predominantly in Image Segmentation, Object Detection tasks, Machine Translation. Dilated Convolutions inserts zeros in the traditional CNN. It helps in increasing the receptive area size i.e. field of view without increasing the number of parameters. This results in optimized consumption of computational resources compared to traditional CNN. The dilated convolutions significantly enhance the performance of image classification tasks. C. Liu et al. proposed an optimized dilated ResNet to extract the most important information from images for improved classification accuracy.

Much of the earlier research was based largely on radar- based hand gesture recognition. To the best of our knowledge, there have been, thus far, few works on person identification using radar based hand gestures. Moreover, the application of Dilation Convolutions for Radar Based Sensors is very minimal. The main aim of our work is to identify the person based on any one hand gesture among the multiple trained gestures like Waving, Clicking, Pinching, and Swiping. This is achieved effectively by building an optimized Model without compromising on its performance. In our work, we have identified 6 persons based on 4 of their hand gestures, including wave, swipe, pinch and click, accessed from the Dop Net Radar Database.

The main contributions of our research are as follows:

- Designing a novel dilated convolution model that identifies persons on the basis of radar-based hand gestures which, to the best of our knowledge, is the first of its kind.
- The Dilated Convolution Model with different dilation rates at different Convolutional layers, exhibit optimized resource utilization besides capturing the context of the image completely with good recognition rates is proposed, deployable for real life scenarios.
- We have made a comparative Analysis of the proposed Model with the existing Traditional CNN, Dilated Convolution Models.

The subsequent sections of the paper are organized in the following manner: Sect. 2 describes the background to the study of Radar, Convolutional Neural Networks and

Dilated Convolutional Neural Networks. Section 3 explains the preprocessed radar signals and the proposed Dilated Convolution Neural Network Model with detailed Architecture. Section 4 explains the Experimental Results based on classification and computational Analysis. Section 5 concludes the paper and offers insights for future directions.

2 Background Study

2.1 Radar Signal Representation

The backscattered signals from radar represent the movements from the individual body parts of humans. Radar signals are analyzed with time-frequency (T-F) representation for insights into such movements. The short time Fourier transform (STFT) cuts a radar signal into narrow, small, time slice windows and uses the Fourier transform to produce the spectral information of every window slice. This is done by having $y(t)$ as the backscattered signal and using a sliding window function $w(t)$ along with the signal $yw(t)$, that is given by the Eq. 1.

$$Fy(t, w) = \int_{-\infty}^{\infty} y(t + \tau)w(\tau)e^{-jw\tau} d\tau \quad (1)$$

A spectrogram is obtained from the individually cut window slices represented in the time frequency domain, which helps the classifier comprehend minute differences between gestures, as well as those in a whole range of gestures made by several people. Large portions of reflected echo signals from the radar represent the hands and palms, with their relatively large frequencies. Hand gestures have a spectral density below 100 Hz, while motions involving other body parts have a larger frequency, extending up to 200 Hz. Moreover, gestures used in this work, such as waving, swiping, pinching, and clicking, have flexion movements that contribute to the unique identification of each. Such finger motions have frequencies from 0 to 100 Hz, and have smaller radar cross sections compared to other parts of the body.

Micro-Doppler signatures are used to analyze flexion micro- movements of the fingers and macro movements of the palms and hands. All these gestures are captured using Frequency Modulated Continuous Wave Radar (FMCW). These low-cost radar sensors are popular for hand gesture acquisition, and give both the range and Doppler signature of the person within the sensor beam. These micro doppler signatures viewed as spectrograms of each gesture are given as the input to the Deep Neural Network Model.

2.2 Traditional CNN

The traditional convolutional neural network (CNN) has 3 layers namely input, hidden, and output layer that classify the output, based on the number of labels. The hidden layers have convolutional and pooling layers going hand in hand followed by the dense layer. CNN is capable of learning features on its own, unlike other machine learning models, and has shown remarkable results with image classification tasks. The convolution operation in traditional CNN is explained in Fig. 1. Sample kernel (K) size is 2×2 which convolves with 2×2 sized input image(I) that has a receptive field of 2×2 .

Given matrices	Calculation Steps	Result
	$(3*1)+(2*0)+(4*0)+(5*1)$ $=3+0+0+5=8(3*1) +$ $(2*0) + (4*0) + (5*1) = 3$ $+ 0 + 0 + 5 =$	8
Top-right value	$8(3*1)+(2*0)+(4*0)+(5*1)$ $=3+0+0+5=8$ $(2*1)+(4*0)+(3*0)+(1*1)$ $=2+0+0+1=3(2*1) +$ $(4*0) + (3*0) + (1*1) = 2$ $+ 0 + 0 + 1 =$	3
Bottom-left value	$3(2*1)+(4*0)+(3*0)+(1*1)$ $=2+0+0+1=3$ $(4*1)+(5*0)+(1*0)+(2*1)$ $=4+0+0+2=6(4*1) +$ $(5*0) + (1*0) + (2*1) = 4$ $+ 0 + 0 + 2 =$	6
Bottom-right value	$6(4*1)+(5*0)+(1*0)+(2*1)$ $=4+0+0+2=6$	
	Resulting Matrix	$\begin{bmatrix} 5 & 8 \\ 3 & 6 \end{bmatrix}$

Fig. 1. Convolution Operation

The area of the filter in the input image is said to be the receptive field. The stack of convolutional layers helps in learning the useful information from images as CNN learns the values of kernels or convolutional filters through back propagation. In this work, we have 2d convolutions with 2d kernels extending to a depth of 3 channels Red, Green, Blue (RGB). The kernels may also vary, for each channel depth. Increasing the number of filters and convolutional layers in the CNN helps it perform well in regard to the problem at hand, but its limitation is that it is computationally expensive [33]. While computations can be reduced by having smaller-sized filters and increasing the number of strides, it reduces the receptive field size, yielding little information from the input images. This is resolved by dilated CNN explained in the next section.

2.3 Dilated CNN

The dilated CNN offers the best solution to the problem of overcoming the intensive computation overhead of the convolutional layer. Dilation rate is introduced in the convolutional kernels. When the dilation rate is 1, it is similar to the traditional CNN. When the dilation rate increases, the number of spaces or gaps in the kernel increases correspondingly, as shown in Fig. 2. When the dilation rate is 2, one space is introduced in the dilated convolution kernels.

2.4 Dilation Rate = 3

It is clear that whenever the dilation rate increases the size of the receptive field also increases. This helps the network to learn more contextual insights about the images, besides keeping the kernel size and the number of parameters same. The red asterisk (*) in Fig. 2 indicates that the number of parameters is the same as the size of the kernel. The size of the receptive fields in Fig. 2A, B, and C is $3 * 3$, $7 * 7$, and $11 * 11$, respectively.

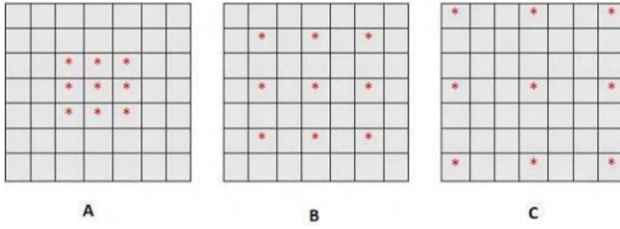


Fig. 2. A. Dilation Rate = 1, B. Dilation Rate = 2,

The advantage of dilated convolution is that the receptive field is increased to improve contextual information for better accuracy and performance at the least cost.

3 Proposed Work

3.1 Radar Data Collection

The Dopnet radar dataset has 4 gestures, namely, wave, pinch, click, and swipe, of 6 persons captured by the 24 GHz FMCW radar with a standalone GUI to control the data obtained. The FMCW system uses one transmit and two receiver antennas with a bandwidth size of 750 MHz and a chirp bandwidth of 1 ms. The subject was made to stand in front of the radar at a distance of approximately 30 cm, and 30 s of each gesture captured by repeating the same gesture several times within the time window. The steps of preprocessing of radar signals is explained clearly. The preprocessed data is the matrix of Doppler vs Time signals. Gesture samples of different persons obtained from the dataset are showing Fig. 3 as spectrograms.

Each column represents each gesture namely wave, pinch, swipe and click. Each row corresponds to various gestures of each individual person, namely Person A to Person F. The spectrograms represent the body's lighter and heavier movements. These spectrograms specifically indicate hand movements. The light blue color represents the lighter movements and the yellow color indicate the heavier movements of the hands in action. The wave gesture in the first column is clearly different from the other gestures as it has more oscillations for long period of time. It also exhibits visible variation of wave gesture among different persons as well. Person C's wave gesture is little similar to Person F's wave gesture, but still there is minute variation of person F from C's gesture representing more neighboring region $D_k * D_k$ from the previous layer. Suppose we have a kernel sized $3 * 3$ and the dilation rate is 3 through all the convolutional layers, only 13 of a total of 36 pixels will contribute information on the image. If the dilation rate remains same in all convolutional layers each pixel p in the first dilated layer may lose more information according to the Eq. 2 oscillations from left to right. Moreover each and every person's gesture is unique and it is evident from the spectrograms. Wave and Swipe has gesture activity of longer time than the other two gestures which has shorter span comparatively. Pinch and click gestures exhibit more or less similar patterns. Certain gestures like wave and swipe of different persons reveals more variations, but there are a few gestures that show only minimal variations among persons. The proposed model

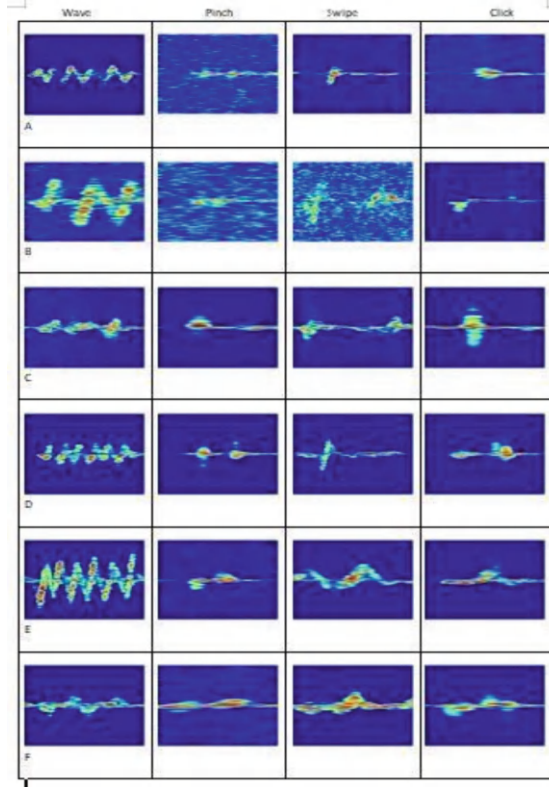


Fig. 3. Spectrograms of 6 Persons performing 4 Gestures

aims to recognize the person based on the given gesture instead of identifying the type of gesture performed. The model is trained with the repeated capture of the same gesture of each and every person as spectrogram input images, to learn the differences and recognize the person based on the gesture chosen. This kind of fine-grained classification can be done for such visually similar input images by CNN. This could be a real challenge for a classifier to identify the person based on his or her Radar based hand gestures.

$$p = \frac{(w * h)}{r^2} \quad (2)$$

3.2 Proposed Dilated CNN Model

The micro Doppler signatures obtained from the Dop Net Dataset is used as the input to the Network Model. Dilated CNN was proposed to overcome the challenges faced in the Traditional CNN. We have analyzed traditional CNN, Dilated CNN and proposed Dilated CNN and the observations are discussed on the next section. Dilated convolutions inserts gaps between each pixel in the kernel of the convolutional layer. For a convolutional

kernel size $k * k$, the resulting dilated filter size is given by Eq. 3.

$$Dk = k + (k - 1) * (r - 1) \quad (3)$$

where, Dk represents the dilated filter, k the convolutional kernel size, and r the dilation rate. Introducing zeros may lead to the loss of some important spatial information. For each pixel in the image, information is obtained from the where w and h represent the width and height of the last dilated convolutional layer, and r represents the dilation rate that remains the same across the layers. This could cause discontinuity in the information on the image, impacting performance in terms of accuracy. Hence, to retain key contextual information, it is necessary to choose different dilation rates at each convolutional layer for consistent contextual image information. We have proposed a new Dilated CNN model with different dilation rates at each layer. The main aim of this work is to minimize the information loss by creating a receptive field that covers the maximum learning from the entire image. The maximum dilation rate that can be chosen is based on the Eq. 4.

$$M_i = \max[M_{i+1} - 2r_i, -2(M_{i+1} - r_i), r_i] \quad (4)$$

where r_i is the dilation rate in the i^{th} layer; M_i is the maximum value of in the i^{th} layer, shown in Fig. 4, which depicts the stacking effect of the convolutional layers with the dilation rate ranging from r_1 to r_n . The maximum dilation rate with kernel $k = 3$ can have patterns $r = [1, 2, 5]$. We tried all different dilation rates in the convolutional layers and the analysis is shown in the next section. It is observed that the dilation rate with $[1, 2, 4]$ performed well both on the training and the test data with the kernel size of $3*3$. This indicates that the first layer with a dilation rate of 1, which is similar to the traditional CNN, covers a broad range of pixel information from the image. It is followed by the dilation rate 2 and 4 indicating the finer and more contextual information besides reducing the number of parameters and in turn the computational complexity. The stacking effect of the chosen kernels and their receptive fields is shown in Fig. 4 (Fig. 5).

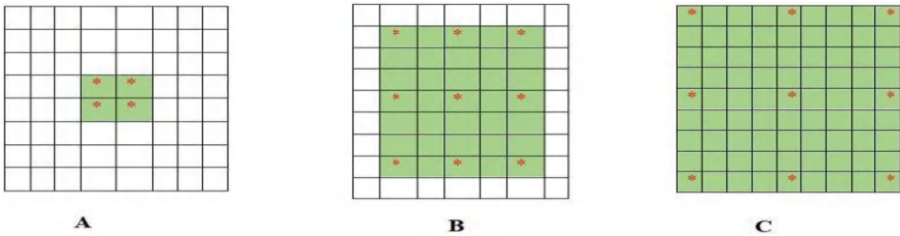


Fig. 4. Receptive Field of Proposed Dilated Convolutional Kernels A. Dilation Rate = 1, B. Dilation Rate = 2, Dilation Rate = 4

The Dop Net dataset used in this work has 4 gestures of 6 persons, such as wave, pinch, click, and swipe. The micro Doppler signature images are divided into training and testing samples. The proposed model has 3 dilated convolutional layers, each with dilation rates of 1, 2, and 4, along with their corresponding max pooling layers, followed

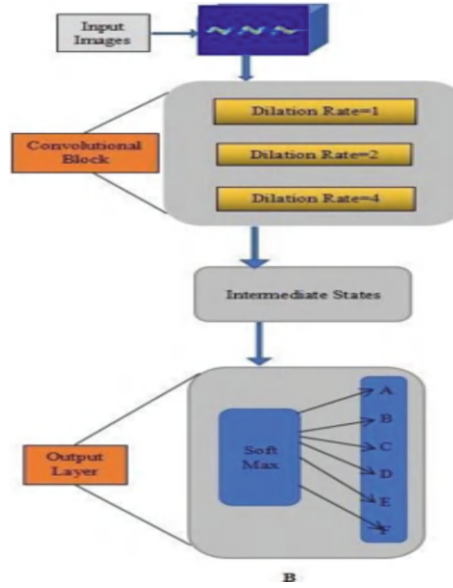


Fig. 5. A Detailed Representation of Dilation layers and the other respective layers in the model

by two dense layers forming a fully connected layer. It is followed by two Dense Layers which form a fully Connected Layer. A dropout of 20% is coupled with the dense layers to drop a few neurons during the learning process in order to reduce overfitting with the training samples. The chosen optimizer is Adam because of its minor weight updates during the learning process. RELU is used as the Activation function in all the layers except the output layer. RELU is used to bring in non linearity into the data which increases the learning and generalizing capability of a model. The experiments were run using activation functions like the Sigmoid, Tanh and RELU, with 15 epochs. It is clear from the plot in Fig. 6 that the Sigmoid and RELU introduce more non-linearity into the data, though the RELU offers better learning than the Sigmoid with respect to accuracy. The last layer uses SOFTMAX Classifier to classify the output. It maps the 6 output classes into a probability distribution. The class with the highest probability is considered as the output.

The loss function used is binary cross entropy, given by Eq. 5.

The Hyperparameters used in the model is given in the Table 1.

The Feature maps in the Fig. 7 shows that the model is getting updated with the weights of the dilated convolutional kernels with different rates during back propagation in the training phase with each iteration.

$$L(\theta) = -\frac{1}{n} \sum_{s=1}^n \sum_{o=1}^c y_{so} \log(P_{so}) \quad (5)$$

where s represents the samples, o the classes, and n the total number of sample points. y is the label for the sample s, $P(so) \in (0,1)$, and o the label for the corresponding samples.

Figure 7A, B, and C represent the feature maps of the first, second, and third Dilated Convolutional Layers respectively. The first feature Map is shown in Fig. 7A represents

Table 1. Chosen Hyperparameters and its values

Hyperparameter	Value
Learning Rate	0.001
Batch Size	32
Epoch	35
Activation Function	RELU

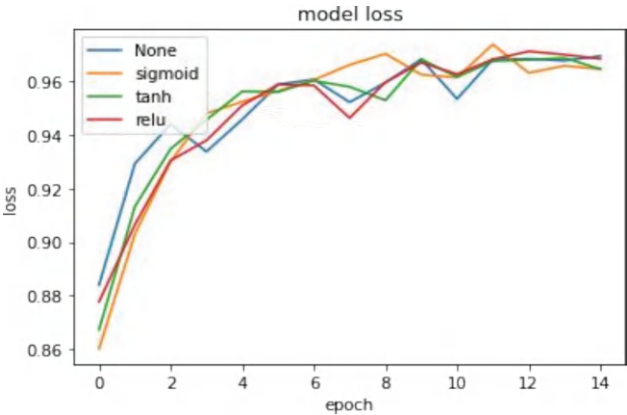


Fig. 6. Performance of Different Activation Functions with respect to loss in the Dop Net Radar Dataset

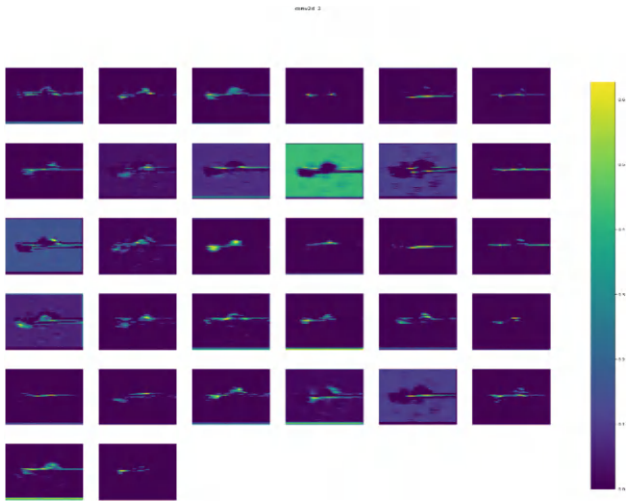


Fig. 7. Feature maps of Dilated Convolutional Kernels

the generic features i.e. the overall pattern of the gesture is captured. Figure 7B indicates the learning by extracting the specific pattern of each gesture. The deeper the layers, the more unique pattern specific individual features are identified by the network model. In the Fig. 7C the unique variations like edges, bends of each gesture are captured in this layer. This extraction makes CNN recognize the images better by retaining the overall and specific features of each gesture during training.

This type of feature Extraction method proves CNN to be a better feature extractor. Dilation in the convolutional layer helps to increase the receptive field exponentially while the parameters are increased linearly. Hence there should be a careful choice of dilation rates in each layer to retain the features without losing any important information of the image.

The total number of trainable parameters, at 2,50,438, is far fewer than that of the traditional CNN, without degrading classification performance.

4 Experimental Results

4.1 Proposed Model Based Classification Results

In the process of identifying the User based on gesture recognition, it is more important to capture and send all the image information to the learning model without losing any unique feature about that particular person's gesture. The choice of dilation rate is analyzed and is clearly explained in Fig. 4. The model covers the entire image with a smaller number of parameters without missing any important information. To ensure that the model works well with the new data, the learning for loss is analyzed.

The graph in Fig. 8 indicates that the loss decreases as test accuracy increases, showing that the model generalizes well with the new data without overfitting it to the training data. The model has achieved a recognition rate of 99.98% with 3825 training data samples and 97.47% as the test accuracy with 957 data samples with 35 epochs. This is given in Fig. 9 for training and testing accuracy.

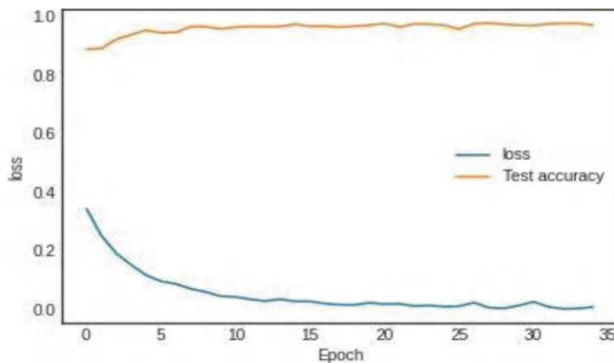


Fig. 8. Model Test Loss vs Test Accuracy

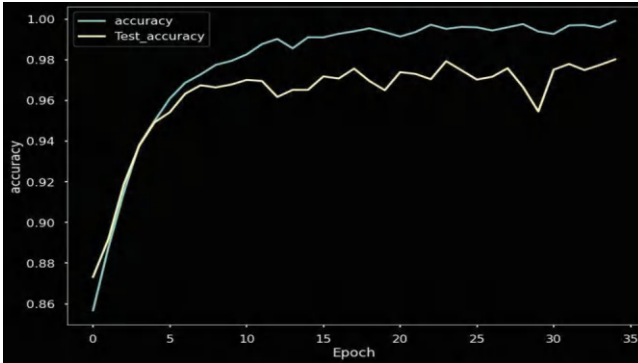


Fig. 9. Model Training Accuracy vs Test Accuracy

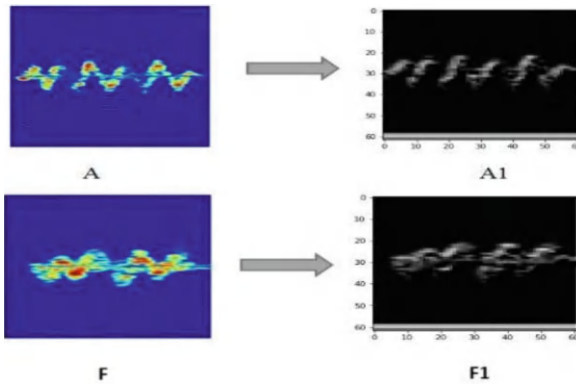


Fig. 10. Original Image and its convolutional layer output

Individual Gesture Feature maps of some samples are given in Fig. 10. The waving gestures of Person A and Person F are depicted, with visible differences between the two gestures. The output of the convolutional layer of the wave gesture of both persons A and F is given. Figure 10, A indicates the original gesture spectrogram image and A1 represents its convolutional layer output of person A. Similarly, Fig. 10, F, and F1 represent the original image and its convolutional image output of Person B. The overall image capture from this layer output is good enough for the model to learn the differences among persons under study, based on their gestures.

It is observed that most of the image information is retained with the proposed model with different dilation rates at each convolutional layer of each Person's Gesture. This indicates the model has done good feature extraction even through Dilated Convolutions at the rate of 1,2 and 4 in the convolutional layers respectively. This is similar to the normal convolutions as the number of parameters was the same in both models. The receptive fields expand exponentially while maintaining linear growth in the number of parameters. The kernel size was retained as 3*3 in both the models leaving exponential rise only in the receptive field.

When the dilation rate is 1, it is similar to normal convolution. When the dilation rate is 2, the difference in the activation output shape from the max pooling layer to the convolutional layer is 4 i.e., $((31,31,32) - (27,27,32) = 4)$. When the dilation rate is 3, the difference among the layers is 8 i.e., $((13,13,32) - (5,5,32) = 8)$. This shows the illusion of using 5*5 and 7*7 kernel size, but the used kernel size is 3*3 throughout the convolutional block. The receptive field view size is increased exponentially with the values 2, 4, 8 in the first, second, and third convolutional layers respectively.

Performance Analysis

The performance analysis is done with Traditional CNN, Dilated CNN, and Proposed DRCNN concerning the number of parameters, Memory, training, and testing accuracy. The large number of parameters makes the model computationally expensive. The number of parameters has been drastically reduced from 7 lakhs to 4 lakhs with dilated CNN indicating a 43% reduction in the number of parameters. Dilated CNN with rate 3 indicates a 58% reduction in the number of parameters concerning traditional CNN. In this work, the proposed model has achieved a 68% and 24% reduction in the number of parameters about the traditional CNN and dilated CNN, respectively. Given that a large number of parameters could result in the model overfitting, a smaller number with good classification recognition makes the model easier to deploy in real-life scenarios.

The testing accuracy of the proposed model in Table 3 also indicates that it learned the new data well even with fewer parameters, and without overfitting.

The proposed Dilated CNN with chosen dilation rate has obtained better results concerning state of art CNN Architectures.

The reduction of the number of parameters achieved by using dilation is mainly obtained by the lower half of the architecture. As discussed in the previous sections, dilations in the CNN vary only in the receptive fields and the kernel size remains the same in each layer, resulting in the same number of parameters in the upper half of the architecture.i.e.in the convolutional Layers.

Table 2 indicates the major contribution of the first Dense Layer yielding an 87% reduction in the number of parameters concerning Traditional CNN. After Dilated Convolutions, the output shape of the proposed model is (2, 2, 32) which is given as the input to the flattening layer. The output shape of traditional CNN without dilation is (6, 6, 32), given as input to its Flatten Layer.

Table 2. Number of Parameters in the Dense Layers of Traditional CNN and Proposed DCNN Model

Layers	Traditional CNN Parameters	Proposed DCNN Parameters
Dense Layer_1	5,90,336	66,048

The Flatten Layer of each model converts the output shape to a single dimension yielding 128 (2 * 2 * 32) parameters for the former and 1152 (6 * 6 * 32) parameters for the latter. Both the models had 512 Dense Units in the First Dense Layer. The default value one is taken as Bias and hence for 512 units, we have a bias value of 512. The

Dense Layer Calculation (DL) is given by the Eq. 6

$$DL = (N\_DL\_units * FL\_P) + B \quad (6)$$

where N_DL_units represent the number of Dense Layer Units, FL_P the Flatten Layer Parameters, B the Bias.

Dense Layer of Traditional CNN = $512 + (512 * 1152)$ Dense Layer of Proposed DCNN = $512 + (512 * 128)$.

Table 3 indicates the performance Analysis results with different Convolutional Networks along with the metrics.

This table summarizes the performance of various convolutional networks using different metrics such as no of parameters, memory, training accuracy and testing accuracy.

Table 3. Performance Analysis with different Convolutional Neural Network and its Metrics

CNN Type	No of Parameters	Memory	Training Accuracy	Testing Accuracy
Normal CNN	774726	271.2 K	0.9975	0.9659
Dilated CNN Dilation Rate = 2	447,046	156.5 K	0.9953	0.9717
Dilated CNN Dilation Rate = 3	332,358	116.3 K	0.9976	0.9662
Proposed Dilated CNN Model	250,438	87.65 K	0.9998	0.9747

Computational Analysis

The main disadvantage of traditional CNN is it has a huge number of parameters leading to high computational complexity. The weights and bias are updated for every parameter during backpropagation learning. Neuron Activations are done by the activation function at each layer.

The RELU activation function is used in this work, as explained in Eq. 7.

$$F(x) = \{if (W * X + B) \geq 0 \text{ output } 1; \text{ else } 0\} \quad (7)$$

with proposed DCNN resulting in a reduced cost of 1.2%. Both the loss functions will eventually converge and saturate at higher epochs achieving global optima.

The updates of weight and bias result in major computational complexity. Computational complexity mainly depends on the number of tensor floating point operations in the model. Hence when the number of parameters is reduced, the weights and bias updates also get reduced. This in turn minimizes the number of Floating Point Operations. Table 4 indicates the Activation size of the proposed method with the traditional CNN approach in the convolutional layers of the model. The weights in the convolutional of traditional and proposed CNN remain the same in the first convolutional layer since the

dilation rate is 1 in the proposed model as well. In the subsequent convolutional layers, the weights to be updated therein during backpropagation have been reduced by 13.4% and 82%, respectively, in the second and third convolutional layers.

Table 4. Number of Weights in Traditional CNN and proposed DCNN

Layers in the Network	No of Weights in Traditional CNN	No of weights in the Proposed Dilated CNN
Convolution Layer 1	123008	123008
Convolution Layer 2	26912	23328
Convolution Layer 3	4608	800

The cost function (Cj)is given in Eq. 8.

$$C_j = \frac{1}{N} (Sum\ of\ Loss\ Error\ in\ N\ Samples)$$

(8)

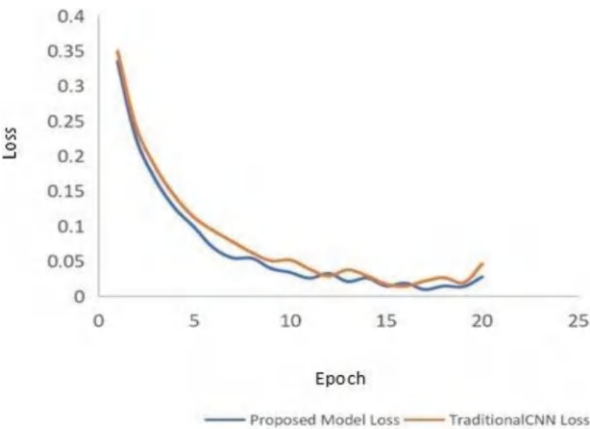


Fig. 11. Cost Function Comparison of Traditional CNN and Proposed Model

where N represents the total number of Samples. The cost function is the average of the loss function.

The confusion matrix in Fig. 12 indicates the categorical accuracy for each class of the model. Person B is correctly classified with a 100% classification rate with any one of his 4 gestures given as input. Gestures of person A, C, D, E, and F are classified with an accuracy of 96%,92%,94%,88%, and 93% indicating reasonable performance on all other person’s identity based on one of his/her gestures as well.

The cost function represents how the model learns the minimal value of loss indicating the good performance of the Model learning. The graph given in Fig. 11 shows the cost function value of 0.081 with Traditional CNN and 0.069.

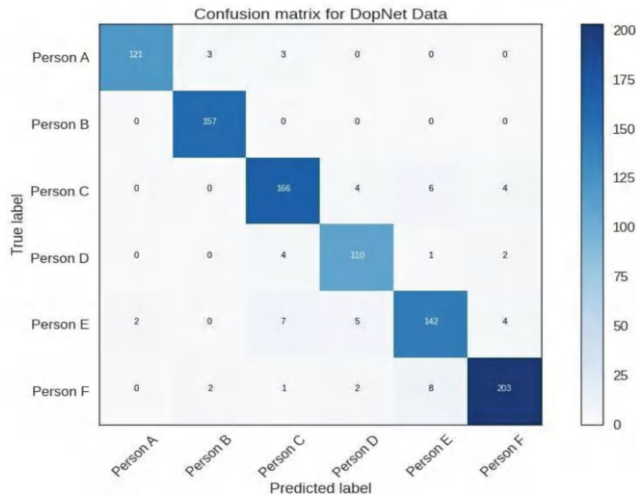


Fig. 12. Confusion Matrix for Person Identification based on Radar Hand Gestures for Dop Net Dataset

Gesture based User Identification done in previous works is based on optical sensors. The algorithms used, including the linear discrimination analysis, support vector machine, and K-nearest neighbor, obtained 92.8%, 69.3%, and 89.9% accuracy, respectively. The gestures of 7 users, including left hand drawing, right hand drawing, and clapping, were reclassified and obtained 94.62% accuracy using the LSTM, 95% with the Bi-LSTM, and 96.75% with the Bi-GRU. Several studies on gesture recognition using radar sensors have been carried out, though not on gesture-based person recognition.

Table 5. Comparison with state of art methods for human identification

Models	Accuracy	Dataset
KNN	94.6	DG5 Hand Accuracy
SVC	94.8	DG5 Hand Accuracy
CNN + LSTM	87.1%	Soli
Random Forest	92.1%	Soli
SVM Quadratic	74.2%	DopNet
CNN + LSTM	87%	DopNet
Proposed Dilated CNN	97.47	DopNet

Our Proposed Model has achieved good recognition rates concerning Radar Hand Gesture User identification. Table 5 indicates the comparative results of state of art methods for human identification. Two works are compared with the DopNet datasets itself. This indicates that the person identification based on radar-based gestures is feasible and

also works effectively based on the obtained results. Our work has also succeeded by reducing the number of parameters without degrading the performance of the Model.

5 Conclusion and Future Work

In this work an optimized Dilated Deep Convolutional Neural Network is proposed for the Person Identification using one of the multiple gestures available in the Dop Net Gesture Dataset. The model has achieved excellent performance with 97.47% as accuracy for the test data. Our findings show that our model outperforms the traditional CNN and dilated CNN in terms of accuracy and computational complexity. This work has its own importance of its kind, recognizing humans using radar-based dynamic gesture signals. Our results clearly demonstrate that the proposed dilated model performs well with fewer parameters, making it ideal for real-

deployment. The reduction of parameters was about 68% compared to the traditional CNN. Person recognition using radar-based gestures can be used to monitor people in small office groups, hospitals, and just about anywhere human-computer interactive applications are called for. In future, this work can be extended with the Dilated CNN along with the LSTM Networks. Further, the dispatch of micro-Doppler images directly as RGB color spaces, serving as input to the model, can be extended to incorporate other color spaces as well, to the same end to get more insights.

References

1. Shanthakumar, V.A., Peng, C., Hansberger, J., et al.: Design and evaluation of a hand gesture recognition approach for real-time interactions. *Multimedia Tools Appl* (2020). <https://doi.org/10.1007/s11042-019-08520-1>
2. Skaria, S., Al-Hourani, A., Lech, M., Evans, R.J.: Hand-gesture recognition using two-antenna doppler radar with deep convolutional neural networks. *IEEE Sens. J.* **19**(8), 3041–3048 (2019). <https://doi.org/10.1109/JSEN.2019.2892073>
3. Rechy-Ramirez, E.J., Marin-Hernandez, A., Rios-Figueroa, H.V.: A human–computer interface for wrist rehabilitation: a pilot study using commercial sensors to detect wrist movements. *Vis. Comput.* **35**, 41–55 (2019). <https://doi.org/10.1007/s00371-017-1446-x>
4. Ohn-Bar, E., Tawari, A., Martin, S., Trivedi, M.M.: Predicting driver maneuvers by learning holistic features. In: *IEEE Intelligent Vehicles Symposium Proceedings*, pp. 719–724 (2014)
5. Edris, S.S., Zarka, M., Ouarda, W., Alimi, A.M.: A fuzzy ontology driven context classification system using large-scale image recognition based on deep CNN. In: *Sudan Conference on Computer Science and Information Technology*, pp. 1–9 (2017)
6. Li, K., Jin, Y., Akram, M.W., Han, R., Chen, J.: Facial expression recognition with convolutional neural networks via a new face cropping and rotation strategy. *Vis. Comput.* **36**, 1–14 (2019)
7. Gifford, R.H., et al.: The relationship between spectral modulation detection and speech recognition: adult versus paediatric cochlear implant recipients. *Trends Hear.* **22**, 1176 (2018)
8. Molchanov, P., Gupta, S., Kim, K., Pulli, K.: Multi- sensor system for driver’s hand-gesture recognition. In: *Automatic Face and Gesture Recognition (FG)*, 2015 11th IEEE International Conference and Workshops on, vol. 1, pp. 1–8. IEEE (2015)
9. Zhang, J., Tao, J., Shi, Z.: Doppler-Radar Based Hand Gesture Recognition System Using Convolutional Neural Networks (2019). https://doi.org/10.1007/978-981-10-6571-2_132

10. Lien, J., Gillian, N., EmreKaragozler, M., et al.: Soli: ubiquitous gesture sensing with millimeter wave radar. *ACM Trans. Graph.* **35**(4), 142 (2016)
11. Amin, M.: *Radar for Indoor Monitoring: Detection, Classification, and Assessment*. CRC Press (2017)
12. Amin, M.G., Zhang, Y.D., Ahmad, F., Ho, K.D.: Radar signal processing for elderly fall detection: the future for in-home monitoring. *IEEE Signal Process. Mag.* **33**(2), 71–80 (2016)
13. Romaszewski, M., Głomb, P., Gawron, P.: Natural Hand Gestures for Human Identification in a Human-Computer Interface, *Image Processing Theory, Tools and Applications*, <http://arxiv.org/abs/1109.5034>



Multimodal Iris, Face and Fingerprint Biometric Fusion for Enhanced Person Identification Using Hybrid Deep Learning

Sharad B. Jadhav¹✉, N. K. Deshmukh¹, K. A. Hambarde², and M. S. Darak¹

¹ School of Computational Sciences, SRTM University, Nanded, India
jadhavsb96@gmail.com

² University of Beira Interior, Covilhã, Portugal

Abstract. In today's digital world, biometric technologies play a pivotal role in ensuring security and access control. As the global demand for biometric systems grows, integrating multiple features becomes imperative. Getting useful information out of specific biometric modalities is the key to making these kinds of systems resilient. It is widely believed that multimodal biometric security systems are safer and more accurate than unimodal ones since they use a combination of features. To address the challenges posed by potential attacks using stolen or compromised biometric data, we present an advanced approach: Multimodal face, finger print and iris Biometric Fusion Using Hybrid Deep Learning for Enhanced Person Identification (MFFI-HDL-EPI). First, we employ using Deep Neural Network Algorithm (DNN) named Graph Convolutional Neural Network (GCN) that excels at capturing complex relationships in graph-structured data, making them suitable for multimodal feature fusion. Next, we employ Improved Group Search Optimization (IGSO) that dynamically adapts search strategies and parameter adjustments, leading to faster convergence and improved feature optimization. Our proposed Multimodal Fusion approach combines face, fingerprint, and iris biometric features. By leveraging GCNs and IGSO, we enhance person identification accuracy. For minimizing error metrics and improving prediction accuracy, DBN-SVM Meta-Classifer is presented. It uses DBN features as input to a Support Vector Machine (SVM) for final decision-making. Performance evaluation metrics include F-measure, recall, precision and accuracy. Additionally, we assess error metrics such as GAR (Genuine Acceptance Rate), FRR (False Rejection Rate), FAR (False Alarm Rate), and EER (Equal Error Rate). In summary, MFFI-HDL-EPI represents a significant advancement in multimodal biometric fusion, offering heightened security and accuracy for person identification.

Keywords: Multimodal · face · fingerprint · iris · Biometric Fusion · Hybrid Deep Learning · Enhanced Person Identification

1 Introduction

In today's rapidly evolving digital landscape, the need for robust security and reliable access control mechanisms has become paramount [1]. With the proliferation of digital transactions and the increasing reliance on electronic systems for various aspects of our

lives, ensuring the authenticity of individuals accessing sensitive information or restricted areas is crucial [2]. Traditional methods of authentication, such as passwords or PINs, are susceptible to being forgotten, shared, or stolen, thereby compromising security [3]. In this context, biometric technologies have emerged as a promising solution, offering unique physiological or behavioral traits for individual identification [4].

Biometric systems leverage distinctive features inherent to individuals, such as facial characteristics, fingerprints, or iris patterns, to verify identity [5]. Unimodal biometric systems, which rely on a single trait for identification, have shown limitations in terms of accuracy and reliability [6]. For instance, a fingerprint scanner may fail to authenticate individuals with worn-out or damaged fingerprints, while facial recognition systems may encounter challenges in low-light conditions or when facial features are partially obscured [7].

To address these limitations and enhance security, multimodal biometric systems have gained prominence [8]. These systems integrate multiple biometric traits to improve accuracy, robustness, and resistance to spoofing attacks [9, 10]. By combining complementary modalities such as face, fingerprint, and iris recognition, multimodal systems offer a more comprehensive and reliable means of person identification.

1.1 Contributions

The novel contributions of this study are:

1. We integrate GCNs (Graph Convolutional Neural Networks) for capturing complex relationships in multimodal biometric data.
2. We implement IGSO (Improved Group Search Optimization) for dynamic optimization in feature fusion.
3. We perform fusion of face, fingerprint, and iris biometrics for enhanced identification accuracy.
4. We utilize GCNs and IGSO to mitigate risks associated with compromised biometric data.
5. For minimizing error metrics and improving prediction accuracy, we present DBN-SVM Meta-Classifer. It uses DBN features as input to a Support Vector Machine (SVM) for final decision-making

2 Literature Review

Medjahed et al. [11] developed multimodal biometric systems using matching score concatenation fusion of left and right palm prints, faces, and other biometric data. KNN and CNN were utilized to identify and recognize multimodal biometric scores. The IITD palm print database and FEI face dataset were used to train safe and durable systems. Even with noise, it outperformed several recent biometric verification methods in experiments.

Shreya et al. [12] compared modal-based human identification methods and found the former better. The multimodal biometric framework they used was highly accurate, with an identification accuracy of 87.3%, significantly higher than threshold-based approaches' 80%. The best latent fingerprint and iris biometric fusion used score level fusion (product rule and sum rule) with 91.15 percent sum rule accuracy. Crime scenarios can benefit from the framework.

Hattab et al. [13] developed a powerful multimodal biometric identification system using facial and iris modalities. Their solution used YOLOv4-tiny for region detection and a novel Deep Learning model influenced by Xception for feature extraction. Classification used LinearSVC, while PCA kept persistent characteristics. Various fusion methods were tested using two-fold cross-validation. Image-level, feature-level, and two score-level fusion approaches were used. Their approach had 100% accuracy on the CASIA-ORL and SDUMLA-HMT multimodal datasets, demonstrating its reliability and performance.

Two face and iris fingerprints and a feature-level deep fusion approach were developed by Safavipour et al. [14]. Two methods were utilized to produce RKHS mappings: dimensionality reduction (KPCA, KLDA) and quaternion-based (KQPCA, KQPCA). Thirdly, deep learning fused feature spaces with in-depth, fully-connected layers. Experiments on six databases showed that the hybrid multibiometric system was better than uni-biometric and other multimodal systems and reached 100% accuracy while enhancing spoof resilience.

Jadhav et al. [15] developed HDL-PI for individual identification using iris, face, and palm print biometrics. They use artifact removal, MGSO to optimize features, and TL-DNN to improve accuracy. Evaluation criteria include recall, accuracy, and precision, F-measure, EER, FAR, FRR, and GAR.

2.1 Research Gaps

Despite advancements, notable gaps remain in multimodal biometric systems. These include the need for robust fusion techniques preserving individual modalities, consideration for real-world scenarios, and scalability concerns. Additionally, standardized evaluation methodologies and ethical considerations require further attention for responsible deployment and advancement of the field.

2.2 Problem Statement

Traditional unimodal biometric systems face limitations in accuracy and security. This study addresses these challenges by proposing a multimodal fusion approach integrating face, fingerprint, and iris biometrics using hybrid deep learning techniques to enhance person identification accuracy and security.

3 Objectives

The novel objectives of this study are:

1. To utilize Graph Convolutional Neural Networks (GCNs) to capture complex relationships in multimodal biometric data.
2. To implement Improved Group Search Optimization (IGSO) for dynamic search strategy adaptation in feature fusion.
3. To integrate face, fingerprint, and iris biometric features for enhanced person identification.
4. To enhance identification accuracy by leveraging GCNs and IGSO in the multimodal fusion approach.
5. To present DBN-SVM Meta-Classifer minimizing error metrics and improving prediction accuracy

4 System Model

The proposed system model, named Multimodal Face, Finger Print, and Iris Biometric Fusion Using Hybrid Deep Learning for Enhanced Person Identification (MFFI-HDL-EPI), integrates Graph Convolutional Neural Networks (GCNs) and Improved Group Search Optimization (IGSO) to enhance person identification accuracy. Initially, Graph Convolutional Neural Networks (GCNs) are employed to capture intricate relationships within graph-structured multimodal biometric data. Subsequently, Improved Group Search Optimization (IGSO) dynamically adjusts search strategies and parameters, facilitating faster convergence and enhanced feature optimization. By combining face, fingerprint, and iris biometric features, the system achieves superior identification accuracy. Figure 1 shows proposed system model.

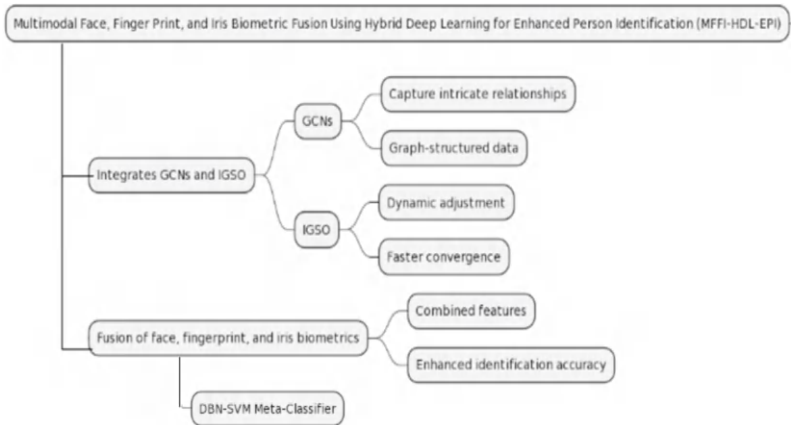


Fig. 1. Proposed System model

5 Proposed Methodology

The proposed methodology, Multimodal Face, Finger Print, and Iris Biometric Fusion Using Hybrid Deep Learning for Enhanced Person Identification (MFFI-HDL-EPI), comprises several key steps:

5.1 Integration of GCNs and IGSO

The system integrates Graph Convolutional Neural Networks (GCNs) and Improved Group Search Optimization (IGSO) techniques. GCNs are initially employed to capture complex relationships within graph-structured multimodal biometric data as pre-processing. Meanwhile, IGSO dynamically adjusts search strategies and parameters to facilitate faster convergence and improved feature optimization.

5.1.1 GCNs for Pre-processing

One subset of deep learning algorithms, Graph Convolutional Neural Networks (GCNs) are trained on data that is structured in a graph. In contrast to conventional neural networks, GCNs are able to accurately represent the intricate web of relationships and dependencies seen in graph data, as opposed to more grid-based data types like images or sequences. In the context of multimodal biometric fusion, GCNs are employed to analyze and extract meaningful features from interconnected biometric modalities such as face, fingerprint, and iris. By leveraging the inherent graph structure of multimodal biometric data, GCNs enable the system to comprehensively understand and represent the intricate relationships between different biometric features, ultimately enhancing the accuracy and reliability of person identification systems.

5.1.2 IGSO for Feature Selection

Improved Group Search Optimization (IGSO) is a dynamic optimization algorithm inspired by the collective behavior of social groups. Unlike traditional optimization techniques, IGSO operates by dynamically adapting search strategies and parameters based on the performance of individual solutions and their groups. In the context of multimodal biometric fusion, IGSO plays a crucial role in optimizing feature fusion by dynamically adjusting search strategies to facilitate faster convergence and improved feature optimization. By effectively exploring the solution space and adapting to changing conditions, IGSO enhances the system's ability to identify optimal solutions, ultimately leading to enhanced person identification accuracy.

Algorithm 1: Graph Convolutional Neural Networks and Improved Group Search Optimization-based Multimodal Biometric Fusion Algorithm" (GCN-IGSO-MBFA)

Initialization:

- Initialize population P with random solutions.

Evaluation:

- Evaluate fitness $f(x)$ of each solution x in P

Iteration:

- Repeat until convergence:

- Select groups of solutions based on fitness:

$$S_g = \{x_i \mid f(x_i) \geq \text{threshold}\} \text{ (IGSO)}$$

- Apply search operators within each group:

$$x_i^{(t+1)} = x_i^{(t)} + \Delta x_i \text{ (IGSO)}$$

where, $x_i^{(t)}$ represents the current solution at iteration t

Δx_i represents the change applied to solution x_i based on the operator outcome

- Update solutions based on operator outcomes
- Evaluate fitness of updated solutions using GCNs:

$$f(x_i^{(t+1)}) = GCN(x_i^{(t+1)})$$

where, $f(x_i^{(t+1)})$ represents the fitness of the updated solution $x_i^{(t+1)}$

$GCN(\cdot)$ denotes the Graph Convolutional Neural Networks (GCNs) function applied to the updated solution $x_i^{(t+1)}$

Termination:

- Return best solution found:

$$\text{Best Solution} = \underset{x_i}{\operatorname{argmax}} f(x_i)$$

where, **Best Solution** represents the solution with the highest fitness value

$f(x_i)$ represents the fitness of solution x_i

$\operatorname{argmax} f(x_i)$ denotes the argmax operation, which returns the argument

x_i at which the function $f(x_i)$ attains its maximum value

IGSO is explicitly used in the steps of selecting groups of solutions and applying search operators within each group. Meanwhile, GCNs are employed to evaluate the fitness of updated solutions, ensuring that the algorithm captures complex relationships in multimodal biometric data during the evaluation process.

5.2 Feature Fusion

Finding a single vector that is more discriminative than the input feature vectors is the goal of feature level fusion. A multimodal biometric system that integrates iris, fingerprint, and facial scans is being considered. Our method enhances the performance of the biometric system by combining fusion at the score and decision levels, making the most of both. The min-max and Z-score approaches are used to change the scores, and the rules for fusion include min, max, sum, and weighted sum. In decision level fusion, we used the OR rule. The system combines face, fingerprint, and iris biometric features. Through this fusion process, the system aims to leverage the complementary information provided by each modality, enhancing the overall identification accuracy.

5.3 Hybrid Deep Learning-Based Person Identification Classification for Enhancement

Our proposed deep learning techniques are utilized to further enhance the identification accuracy. By employing hybrid deep learning approaches, the system can effectively process and analyze the multimodal biometric data, leading to superior performance in person identification tasks.

For minimizing error metrics and improving prediction accuracy, DBN-SVM Meta-Classifer is presented. It uses DBN features as input to a Support Vector Machine (SVM) for final decision-making.

Deep Belief Network (DBN) is a type of deep learning model composed of multiple layers of stochastic, latent variables, allowing it to learn complex hierarchical representations of data. SVM is a powerful supervised learning algorithm used for classification and regression tasks, aiming to find the hyperplane that best separates different classes in the feature space, maximizing the margin between them.

Algorithm 2: DBN-SVM Meta-Classifer for Multimodal Biometric Classification (DBN-SVM-Metaclassifier)

Input: Biometric features from face, fingerprint, and iris modalities:
 $F_{face}, F_{fingerprint}, F_{iris}$

Output: Predicted label for the input biometric features: $\hat{Y}$

1. Feature Fusion:
Combine the features from face, fingerprint, and iris modalities:

$$F_{fusion} = [F_{face}, F_{fingerprint}, F_{iris}]$$

2. DBN Feature Extraction:
Feed the fused features into a Deep Belief Network (DBN) to obtain DBN features:

$$DBN\ features = DBN(F_{fusion})$$

3. Classification using SVM:
• Predict the label for the input biometric features using the trained Support Vector Machine (SVM) model:

$$\hat{Y} = SVM\ Model(DBN\ Features)$$

4. Output:

• Output the predicted label $\hat{Y}$ indicating the predicted identity or category of the biometric features.

Algorithm 2 takes biometric features from face, fingerprint, and iris modalities as input, combines them into fused features, extracts informative representations using a Deep Belief Network (DBN), and then classifies the fused features using a trained Support Vector Machine (SVM) model to predict the label for the input biometric features.

6 Results and Discussion

We provide the results of our MFFI-HDL-EPI approach validation on the benchmark multimodal SDUMLA-HMT dataset, which includes iris and fingerprint biometrics. We compare the current classifiers’ simulation results using two key performance criteria, quality and error metrics, to those of our proposed DBN-SVM Meta-Classifier. Using error measures (EER, FAR, and FRR) and quality metrics (accuracy, precision, recall, and F-measure), we evaluate the suggested DBN-SVM Meta-Classifer.

6.1 Dataset Description

- a. **Fingerprint dataset:** The fingerprint dataset comprises middle finger, index, and thumb images from both hands, captured by 5 different sensors, aiding research in sensor interoperability for fingerprint identification.
- b. **Iris dataset:** Each participant contributed 10 iris images (5 for each eye) to the SDUMLA-iris HMT database, which guarantees consistent and reliable biological features for iris recognition studies. The database contains 1060 iris images taken with intelligent iris capture technology.
- c. **Face dataset:** The SDUMLA-face HMT database comprises 8,904 images capturing variations in poses, expressions, illuminations, and accessories, facilitating real-world face recognition research (Fig. 2).

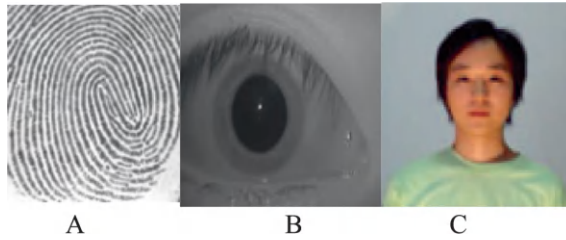


Fig. 2. SDUMLA-HMT dataset with test samples A. fingerprint B. iris C. face

6.2 Comparative Analysis

Table 1 presents a comparative analysis of various classifiers based on error metrics such as GAR (Genuine Acceptance Rate), FRR (False Rejection Rate), FAR (False Alarm Rate), and EER (Equal Error Rate). The classifiers evaluated include LNMF, CNN, DNN, ANFIS, BSIF + LBP, CCA, and the proposed DBN-SVM Meta-Classifier. LNMF achieves an EER of 4.678% with a low FAR but relatively high FRR. CNN-based methods demonstrate competitive performance, with EER values ranging from 3.214% to 3.547%. DNN achieves a lower EER of 2.784%, indicating improved accuracy. ANFIS and BSIF + LBP classifiers achieve EERs of 2.51% and 2.145%, respectively, with varying trade-offs between FAR and FRR. Notably, the proposed DBN-SVM Meta-Classifier outperforms other classifiers with the lowest EER of 0.653%, indicating superior accuracy in biometric classification. Additionally, it achieves a significantly lower FAR and FRR, highlighting its effectiveness in minimizing false acceptances and rejections. Overall, the proposed classifier demonstrates promising performance across all error metrics, making it a robust solution for multimodal biometric fusion.

Table 2 presents a comparative analysis of classifiers based on quality metrics. Each classifier, including LNMF, CNN, DNN, ANFIS, BSIF + LBP, CCA, and the proposed DBN-SVM Meta-Classifier, is evaluated for its performance across these metrics. The LNMF classifier achieves an accuracy of 87.520% with corresponding precision, recall,

Table 1. Error metrics Based comparison

Classifier	Modality	Error metric			
		GAR	FRR	FAR	EER (%)
LNMF [16]	Finger print, face	0.911	0.85	0.089	4.678
CNN [17]	Finger print, Ear	0.88	0.82	0.12	3.547
CNN [18]	Face,finger print, palm	0.77	0.75	0.23	3.214
DNN [19]	ECG, finger print	0.65	0.62	0.35	2.784
ANFIS [20]	Ear and palm	0.41	0.45	0.59	2.51
BSIF + LBP [20]	Ear and palm	0.25	0.37	0.75	2.145
CCA [21]	Fingerprint, Iris	0.11	0.25	0.89	0.8474
DBN-SVM Meta-Classifer (proposed)	Face, Finger print, iris	0.05	0.15	0.97	0.653

Table 2. Quality metrics based comparison

Classifier	Quality metric (%)			
	Accuracy	Precision	Recall	F-measure
LNMF [16]	87.520	87.063	87.765	87.413
CNN [17]	89.230	88.773	89.475	89.123
CNN [18]	92.150	91.693	92.395	92.043
DNN [19]	93.040	92.583	93.285	92.933
ANFIS [20]	91.020	90.563	91.265	90.913
BSIF + LBP [20]	96.020	95.563	96.265	95.913
CCA [21]	95.280	94.823	95.525	95.173
DBN-SVM Meta-Classifer (proposed)	98.870	98.303	99.006	98.753

and F-measure values around 87%. CNN-based methods show improved performance, with accuracy ranging from 89.230% to 92.150%, indicating higher overall correctness in classification. DNN exhibits even higher accuracy at 93.040%, indicating enhanced precision, recall, and F-measure. ANFIS and BSIF + LBP classifiers achieve accuracies of 91.020% and 96.020%, respectively, with corresponding precision, recall, and F-measure values reflecting their robust performance. The proposed DBN-SVM Meta-Classifer surpasses other classifiers with an accuracy of 98.870%, indicating superior correctness in classification. It also demonstrates high precision, recall, and F-measure values, emphasizing its effectiveness in accurately identifying and classifying biometric features.

Overall, the proposed classifier outperforms other methods across all quality metrics, highlighting its capability for accurate and reliable multimodal biometric fusion.

7 Conclusion

In conclusion, the proposed Multimodal Face, Finger Print, and Iris Biometric Fusion Using Hybrid Deep Learning for Enhanced Person Identification (MFFI-HDL-EPI) demonstrate significant improvements in person identification accuracy. Leveraging Graph Convolutional Neural Networks (GCNs) and Improved Group Search Optimization (IGSO), the system achieves superior performance compared to existing methods. Specifically, it achieves an impressive reduction in Equal Error Rate (EER) to 0.653% and a remarkable increase in accuracy to 98.87%.

Nevertheless, certain limitations must be acknowledged. Variations in lighting conditions, pose angles, and occlusions may challenge the system's accuracy, while the computational complexity of deep learning algorithms could hinder real-time implementation, especially in resource-constrained environments.

Exploring robust feature extraction methods, lightweight model development, novel fusion strategies, and the integration of additional biometric modalities are crucial for advancing multimodal biometric fusion towards enhanced person identification.

References

1. James, E., Rabbi, F.: Fortifying the IoT landscape: strategies to counter security risks in connected systems. *Tensorgate J. Sustain. Technol. Infrastruct. Dev. Ctries.* **6**(1), 32–46 (2023)
2. Hasan, M.K., Ghazal, T.M., Saeed, R.A., et al.: A review on security threats, vulnerabilities, and counter measures of 5G enabled Internet-of-Medical-Things. *IET Commun.* **16**(5), 421–432 (2022)
3. Varshney, I., Sagar, S.: Three level password authentication system using cryptography approaches. In: 2023 2nd International Conference for Innovation in Technology (INOCON), pp. 1–7. IEEE (2023)
4. Gavrilova, M.L., Anzum, F., Hossain Bari, A.S.M., et al.: A multifaceted role of biometrics in online security, privacy, and trustworthy decision making. In: Daimi, K., Francia, G., Encinas, L.H. (eds.) *Breakthroughs in Digital Biometrics and Forensics*, pp. 303–324. Springer International Publishing, Cham (2022)
5. Furnell, S.: Biometric technology and user identity. In: *Regulating Cyber Technologies: Privacy vs Security*, pp. 81–110 (2023)
6. Mansour, A., Eddermoug, N., Sadik, M., Sabir, E., Azmi, M., Jebbar, M.: A lightweight seamless unimodal biometric authentication system. *Proc. Comput. Sci.* **231**, 190–197 (2024)
7. Makrushin, A., Uhl, A., Dittmann, J.: A survey on synthetic biometrics: fingerprint, face, iris and vascular patterns. *IEEE Access* **11**, 33887–33899 (2023)
8. Ahamed, F., Farid, F., Suleiman, B., Jan, Z., Wahsheh, L.A., Shahrestani, S.: An intelligent multimodal biometric authentication model for personalised healthcare services. *Future Internet* **14**(8), 222 (2022)
9. Xu, W., Liu, J., Zhang, S., et al.: Anti-spoofing facial authentication based on COTS RFID. *IEEE Trans. Mob. Comput.* **23**, 4228 (2023)
10. Bala, N., Gupta, R., Kumar, A.: Multimodal biometric system based on fusion techniques: a review. *Inf. Secur. J. Glob. Perspect.* **31**(3), 289–337 (2022)
11. Medjahed, C., Rahmoun, A., Charrier, C., Mezzoudj, F.: A deep learning-based multimodal biometric system using score fusion. *IAES Int. J. Artif. Intell* **11**(1), 65 (2022)
12. Shreya, S., Chatterjee, K.: Latent fingerprint and Iris fusion for enhancement of performance of human identification system. *Expert Syst. Appl.* **235**, 121208 (2024)

13. Hattab, A., Behloul, A.: Face-iris multimodal biometric recognition system based on deep learning. *Multimed. Tools Appl.* **83**, 1–28 (2023)
14. Safavipour, M.H., Doostari, M.A., Sadjedi, H.: Deep hybrid multimodal biometric recognition system based on features-level deep fusion of five biometric traits. *Comput. Intell. Neurosci.* (2023). <https://doi.org/10.1155/2023/6443786>
15. Jadhav Sharad, B., Deshmukh, N.K., Humbe, V.T.: HDL-PI: hybrid DeepLearning technique for person identification using multimodal finger print, iris and face biometric features. *Multimed. Tools Appl.* **82**(19), 30039–30064 (2023)



Adaptive Inertial Weight with Artificial Rabbit Optimization for Intrusion Detection System in Cyber Security

Pradeep Chintale¹ , Anirudh Khanna² , Madhavi Najana³ ,
and Hrushikesh Deshmukh⁴ 

¹ SEI Investment Company, Downingtown, PA 19335, USA
chintale.pradeep@gmail.com

² Data Protection Lead, 900 Acadia Dr, Plano, TX 75023, USA

³ Software Engineering Manager, Columbus, OH, USA

⁴ Fannie Mae, Reston, VA, USA

Abstract. Cybersecurity is practice of preventing cyberattacks on vital infrastructure and private data. Many applications involve cybersecurity, such as in banks, hospitals, and other industry sectors, which face issues related to data insecurity, confidential information breaches and accuracy. However, high traffic processed attacks and inaccurate identification of malicious activity lead to high false positives and class imbalance. In this research, Adaptive Inertial Weight with Artificial Rabbit Optimization (AIW-ARO) is proposed for Intrusion Detection Systems (IDS) to secure entire network in cybersecurity. Initially, data is obtained from N-BaIoT dataset and then normalized using a Standard Scaler. Feature selection is performed using the AIW-ARO method to select relevant features and secure data by hiding information and using foraging as an exploration strategy, preventing detection by cybersecurity measures. Finally, classification using Long Short-Term Memory (LSTM) technique identifies attacks and classifies IDS accurately, effectively in cybersecurity. The proposed AIW-ARO technique is evaluated using the N-BaIoT datasets, achieving a higher accuracy of 99.99%. This is compared to existing techniques like Bidirectional Gated Recurrent Unit (Bi-GRU), Convolutional Neural Network (CNN), and Convolutional Neural Network-Long Short-Term Memory (CNN-LSTM).

Keywords: Adaptive Inertial Weight with Artificial Rabbit Optimization · Convolutional Neural Network · Bidirectional Gated Recurrent Unit · Intrusion Detection System and Long Short Term Memory

1 Introduction

Cybersecurity threats are most confronting challenge facing smart grid networks, with the business and economic disruptions that occur, and potentially huge economic losses likely [1]. Among various security techniques, network traffic analysis and classification of attacks play a vital role in Intrusion Detection Systems (IDS). Timely identification

of network traffic patterns and malicious activities such as DDoS attacks and unauthorized access is crucial [2]. Efficient intrusion detection is necessary to secure Internet of Things (IoT) devices from becoming targets of attackers. In network security, IDS is widely used for cybersecurity, and Deep Learning (DL) based IDS includes the perception layer and network layer [3]. Additionally, questions arise about how devices adjust values and parameters across dimensions to compute sample sizes, affecting the rendering of different layers. The IDS observing system detects suspicious activities and generates alerts upon detection. It operates in conjunction with security measures such as authentication and systems designed to counter cyberattacks [4, 5]. Resilience to cybersecurity threats is often limited, but this addressed through intrusion prevention, a key aspect of industrial security that involves technical measures to avoid and mitigate the impact of cyber threats [6]. DL-based approaches in cybersecurity often focus solely on financial factors and opportunities for addressing threats without integrating compliance with a comprehensive cybersecurity framework aimed at reducing investment costs. While these approaches, predictively effective for supporting cost-oriented decisions, they often fail to meet the broader needs of a robust cybersecurity framework. [7, 8].

One main application involves handling poor data, using statistical tests to identify errors in system measurements. These measurements may come from various sources, including faulty meters or cyberattacks [9, 10]. The major challenges related to security include threat data overload and quality of shared data, such as data not being confidential. Privacy and legal issues also arise from sharing information, affecting security of threat data sharing platforms and standard information [11, 12]. The performance of IDS based on classifiers is further improved by using feature selection techniques while minimizing time required for training and testing of model [13]. Therefore, enhance accuracy of such systems while handling complex and heterogeneous data in industries, DL techniques have been involved [14]. Cybersecurity threats have malicious intents and identified as obstructions to information. They undermine cybersecurity measures, public confidence, damage reputation of victim country, and harm legal interests [15]. However, high traffic processed attacks, inaccurate identification of malicious activity leads to high false positives and class imbalance. In this research, AIW-ARO is proposed for IDS to secure entire network in cybersecurity, improve accuracy identification of malicious activity, minimized false positive and imbalance.

The main contribution of research is written below:

- The SMOTE generates synthetic sample for minority class thus addressing class imbalance issue and improve ability of IDS to detect unseen attacks.
- Feature selection using AIW-ARO method selects relevant features and hides information to secure data, while foraging serves as an exploration strategy, preventing rabbits from being detected by cybersecurity measures.
- Classification using LSTM technique identifies attacks and classifies IDS accurately and effectively in cybersecurity.

The paper is organized as follows: Sect. 2 provides a literature review that summarizes Resource allocation-based fog computing. Section 3 introduces the proposed method utilized by AIW-ARO. Section 4 discusses the result and comparative analysis. Section 5 discusses the conclusion.

2 Literature Review

The related work about IDS in cybersecurity feature selection and classification are discussed below along with their advantage and disadvantages.

Prabhat Kumar et al. [16] presented a Bidirectional Gated Recurrent Unit (Bi-GRU) model with a fully connected layer and a softmax classifier designed to improve attack detection process in a smart grid environment. This model was an enhanced version of Recurrent Neural Network (RNN), possessing strong capabilities for long-term dependencies and being minimized computationally complex due to its uncomplicated structure. However, reliability of collected information does not guarantee because communication occurs across an insecure channel at lower plane, and there are issues with gradient exploding or vanishing.

Ahsan Nazir et al. [17] implemented a deep learning-based Convolutional Neural Network (CNN) and Long Short-Term Memory (CNN-LSTM) architecture tailored for precise and efficient IoT threat detection. The CNN extracts spatial features, while the LSTM extracts temporal characteristics, both of which derive high-level features from large datasets. However, CNN-LSTM struggles to differentiate between high traffic processed attacks and malicious activity, leading to high false positives and class imbalance, and it requires large datasets for training.

Fraidoon Sattari et al. [18] introduced a software-defined network environment in IDS using a deep learning-based Deep Neural Network (DNN) and LSTM to identify botnet attacks from massive amounts of data. These combined techniques were effective in achieving high detection accuracy in a short amount of time and could more accurately model complex relationships between input and output. However, they involve high computational complexity, the need for large amounts of labelled data, a lack of interpretability, and require more resources.

Danish Javeed et al. [19] developed a Bidirectional LSTM (Bi-LSTM) for threat detection in cybersecurity. The Bi-LSTM excels at capturing long-term dependencies in sequential information, identifying temporal patterns and trends in network traffic that indicate security threats. It provides enhanced contextual information, reducing false positives by distinguishing between normal behavior and actual security threats. However, it was highly complex and time-consuming due to the need to process data in both forward and backward directions.

Valerian Rey et al. [20] presented a Federated Learning (FL) based framework to deploy against adversarial attacks and to measure robustness of federated approach in an adversarial setup with several malicious participants poisoning federated model. This framework distributes computational load across many devices, reducing the need for centralized computational resources and enabling training of large-scale models. However, malicious participants send incorrect updates to central server, which degrade model performance or lead to model failure.

In the overall analysis the existing approach has limitation like high traffic processed attacks and inaccurate identification of malicious activity lead to high false positives and class imbalance. To solve these problems, AIW-ARO is proposed for IDS to secure entire network in cybersecurity, improve accuracy identification of malicious activity, minimize false positive and imbalance.

3 Proposed Methodology

In this section, proposed AIW-ARO for IDS in cybersecurity is considered in two phases. The first phase involves pre-processing using a Standard Scaler, which is less sensitive to outliers as it focuses on standard deviation. ARO-based feature selection is then applied to remove irrelevant features, helping to avoid overfitting. Finally, classification is performed using SVM to detect attacks in IDS. Figure 1. Block diagram of AIW-ARO and SVM with IDS for cybersecurity.

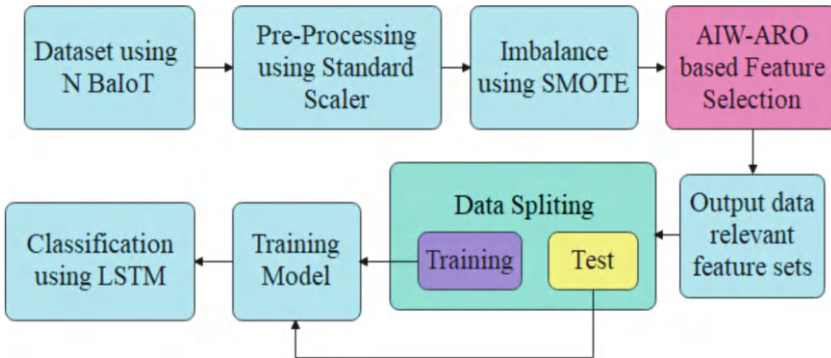


Fig. 1. Block diagram of AIW-ARO with IDS for cyber security

3.1 Dataset

The initial step in constructing an effective intrusion detection system in cybersecurity involves using N-BaIoT dataset, which contains traces of both normal and malicious cyber activity. There are two distinct categories of IoT attack types, subdivided into multiple subclasses such as Mirai, ACK, scan, SYN, UDP, UDP plain, Gafgyt combo, junk, and TCP.

N-BaIoT

The N-BaIoT dataset [21] is used to evaluate and compare proposed approach. It comprises a total of 115 attributes, with being network features and one tag. Among these attributes, ten best features are selected, with dataset consisting of six devices mentioned along with two malware families of Botnet, namely GAFGYT and MIRAI. The GAFGYT node's attack considers the connection of the IP address, while the MIRAI malware involves an ARC process for targeting devices. The dataset contains a total of 170,000 records (Table 1).

3.2 Pre-processing

The collective datasets provided as input for pre-processing are considered a significant task in DL field, as they help eliminate defects in datasets. This is first stage of proposed

Table 1. Details of the N-BaIoT dataset

Devices	Gafgyt	Benign	Mirai
1011 Web Cam	10,000	10,000	10,000
1002 Security Camera	10,000	10,000	10,000
1003 Security Camera	10,000	10,000	10,000
Thermostat	10,000	-	10,000
838E security camera	10,000	10,000	10,000
737E Security Camera	10,000	10,000	10,000
Total Records	60,000	50,000	60,000

method, aimed at transforming the raw network IoT attack data into a format effective for further analysis. In IDS, pre-processing techniques such as Standard Scaling prevent attacks by separating the features, then using mean and standard deviation to make them less sensitive. Standard Scaling is performed to handle less sensitive and non-uniform values compared to min-max method, which is affected by extreme values and leads to uniform values. The standard scaling process involves normalizing data by subtracting mean and scaling it to have unit variance. In mathematical Eq. (1), represents standard deviation and mean.

$$x_{standardized} = \frac{x - \text{mean}(X)}{\text{standard deviation}(X)} \quad (1)$$

It standardizes data to unit variance, useful when features exhibit varying variances and require normalization for equitable treatment. This standardization technique scales attributes to have a mean of zero and a standard deviation of one, effectively normalizing their distribution.

3.3 Synthetic Minority Oversampling Technique

After normalization Synthetic Minority Oversampling Technique (SMOTE) is widely used approach for produce addition sample in dataset. The SMOTE increases diversity of training data which is improve generalization performance of IDS. There are better capture underlying patterns and network traffic leading improved detection accuracy. The cybersecurity in IDS number of instance representing actual attacks is significantly small than normal network traffic [22]. The SMOTE generates synthetic sample for minority class thus addressing the class imbalance issue and improve ability of IDS to detect unseen attacks. The method relies on expand information packets that connect a give point with its K-Nearest Neighboring point. Furthermore, analysis is manage to evaluated impact of various factors dimensionality reduction, and number of neighbors (K) on accuracy of system. The SMOTE algorithm select sample from minority class with in dataset chose sample is represented Y . Parameter k holds a significant value in determine count of neighbors to appraise. The line segment that links Y and m , α parameter as synthetic sample ration numerical parameter use within range is $[0, 1]$. The

mathematically Eq. (2) generating for synthetic sample.

$$v = Y + \alpha(m - Y) \quad (2)$$

where, Y and m is attribute of interpolation on pre feature. There are oversampling of minority sample requires iteratively selecting model identifying its closest neighbours and generating synthetic. The imbalanced nature of dataset a resampling technique is requires before classification faced issues arising from class imbalance. The dataset underwent SMOTE analysis producing resampled data for X and Y variable.

3.4 Feature Selection

After performing SMOTE, the balanced data is provided for feature selection. The proposed AIW-ARO approach is execute for feature selection. In AIW-ARO technique, related features are selected to improve optimization process. The goal of this approach is to choose most useful feature subset for detecting and enhancing security vulnerabilities. When compared to other optimization techniques like Particle Swarm Optimization (PSO), Black Widow Optimization (BWO) and, ARO method aims for maximum efficiency by combining with AIW.

Artificial Rabbit Optimization

The ARO primarily references two phases observed in the natural world to enhance IDS in cybersecurity. The hiding information serves to secure the data, and foraging serves as an exploration strategy, preventing rabbits from being detected by cybersecurity IDS. Random hiding is another strategy in which rabbits move to other burrows to hide further away. The optimization goal is to improve the iteration maximum level based on the population, assigning values based on population size changes in local optimal. The ARO algorithm is based on replicating rabbits' foraging behavior. It formulates recent strategic approaches for nearby burrows to outsmart sharks and protect their habitats. The ARO algorithm enables rabbits to expand their search for sustenance to more remote locations.

Exploration Phase

During the detour foraging (exploration phase) within ARO is assumed that each rabbit in population occupies its own area with some grass and burrows. While foraging, rabbits tend to move randomly far away from other individuals in search of food, often ignoring nearby food. This behavior is known as meandering foraging, and it is mathematically modelled in (3), (4), (5), (6), (7), and (8) below:

$$k_n(h+1) = k_m(h) + G \cdot (y_n(h) - y_m(h)) + \text{round}(0.5 \cdot (0.05 + e_1)) \cdot i_1, n = 1, \dots, i, m \neq n \quad (3)$$

$$G = L \cdot C \quad (4)$$

$$L = (e - e^{\left(\frac{h-1}{H}\right)^2}) \cdot \sin(2\pi i_2) \quad (5)$$

$$c(v) = \begin{cases} 1 & \text{if } k == e(l) \\ 0 & \text{else} \end{cases} \quad v = 1, \dots, d \text{ and } l = 1, \dots, e_3 \cdot b \quad (6)$$

$$g = \text{randperm}(d) \quad (7)$$

$$n_1 \sim N(0, 1) \quad (8)$$

where $k_n(h+1)$ denotes position of n th rabbit at time $h+1$, $y_n(h)$ denotes rabbit position at time h , and n , d represents dimension and $\lceil \cdot \rceil$ was the ceiling function, round indicates rounding to the nearest integer, randperm returns a random permutation of the integers from 1 to d , e_1 , e_2 , and e_3 tri random numbers in $(0, 1)$. The iteration is performed a maximum of H times, next step is denoted by L , and n_1 follows a standard normal distribution. The mapping vector is denoted by c respectively.

Exploitation

The exploitation phase focuses on searching more intensively in these regions by increasing the number of iterations within the identified areas to fine-tune the solution. The rabbit randomly selects a burrow to hide and reduces probability of being hunted. The n th rabbit is in n th burrow, and mathematical equations from (9), (10), (11), and (12) are shown below:

$$a_{n,m}(h) = y_n(h) + T \cdot g \cdot y_n(h), \quad n = 1, \dots, i \text{ and } m = 1, \dots, d \quad (9)$$

$$T = \frac{H - h + 1}{H} \cdot e_4 \quad (10)$$

$$e_2 \sim N(0, 1) \quad (11)$$

$$g(v) = \begin{cases} 1 & \text{if } v == m \\ 0 & \text{else} \end{cases} \quad v = 1, \dots, d \quad (12)$$

The d burrows were generated in the neighborhood of rabbit along every dimension. T is the hiding parameter which is linearly decreased from 1 to $1/H$ with random perturbation over the course of iteration. The simulate random hiding behavior of rabbits mathematically expression are given (13), (14), and (15) below;

$$v_n(h+1) = y_n(h) + E \cdot (e_4 \cdot a_{n,e}(h) - y_n(h)), \quad n = 1, \dots, n \quad (13)$$

$$g_e(v) = \begin{cases} 1 & \text{if } v == e_5 \cdot d \\ 0 & \text{else} \end{cases} \quad v = 1, \dots, d \quad (14)$$

$$a_{n,e}(h) = y_n(t) + T \cdot g_e \cdot y_n(h) \quad (15)$$

where, $a_{n,e}$ denoted randomly chosen burrow from d burrows. e_4 and e_5 was 2 random number between $(0, 1)$. According to Eq. (13) n th rabbit randomly chose a burrow from its d burrows to improve its position. The mathematically Eq. (16) shown below;

$$y_n(t+1) = \begin{cases} y_n(t) & \text{if } f(y_n(t)) \leq f(k_n(t+1)) \\ k_n(t+1) & \text{if } f(y_n(t)) > f(k_n(t+1)) \end{cases} \quad (16)$$

After completing detour foraging and hiding, position of n th rabbit is improved using Eq. (16). The exploitation phase performs a local search within these promising regions to exploit findings and discover a better solution.

Adaptive Inertial Weight with Artificial Rabbit Optimization

The ARO has advantage of simple operation with few parameters to set and strong optimization performance, particularly improved convergence accuracy, speed and avoidance of local optima. The purpose of inertial weight is to improve early stage of algorithm iteration by assigning relatively population large weight values to enhance global search capacity of algorithm. The algorithm adaptively changes size of weight values based on distribution of individuals in recent population, accelerating convergence speed to search around optimal solution. The inertia weight is an essential parameter, and appropriate weight value optimization improves algorithm's performance. Because optimization is used for IDS in cybersecurity feature selection, improper selection of linear inertia weight adjustment approach influences convergence speed of algorithm. The proposed approach adaptively changes inertia weight values based on rabbit population distribution. It is represented by mathematical (17), (18), (19), (20), and (21) shown below:

$$\omega = a_1 \cdot (P_{worst}^m - P_{best}^m) + \frac{a_2}{t} \cdot (P_n^{maxm} - P_n^{minm}) \quad (17)$$

$$x_n^{maxm} = \max_{m \in [1, d]} x_n^m \quad n = 1, \dots, i \quad (18)$$

$$x_n^{minm} = \min_{m \in [1, d]} x_n^m \quad n = 1, \dots, i \quad (19)$$

$$b_1 = \cos(0.5\pi \cdot r) \quad (20)$$

$$b_1 = 1 - a_1 \quad (21)$$

where t denoted number of iterations providing recent population. P_{nworst} and P_{mbest} are position vectors of worst rabbit and position vector of optimal rabbit in current rabbit population, respectively. The proposed method updates inertia weight based on recent population information to adaptively adjust search space between lower and upper boundaries, considering differences between 2 random coefficients to compromise maximum individual distances in every dimension. The adaptive weight adjustment for recent rabbit in random hiding improves position, as shown in Eq. (22) below:

$$v_n(h+1) = \omega \cdot y_n(h) + E \cdot (e_4 \cdot a_{n,e}(h)) \quad (22)$$

The adaptive weight adjustment strategy entails algorithm dynamically altering weight size based on distribution of current rabbit population. During early stages of algorithm iteration, when rabbit population converges to a local optimal solution, difference between optimal solution and worst solution value, denoted as $b_1 \cdot (P_{worst}^m - P_{best}^m)$, is unaffected by population distribution. However, term with larger weight value of ω aids in preventing algorithm from becoming trapped in a narrow search range during

initial iterations. As number of iterations is become high, value of $b_1 \cdot (P_{worst}^m - P_{best}^m)$ gradually becomes minimum, and its influence on weight ω maximum. When algorithm is not performing optimally, design of $b_1 \cdot (x_n^{max} - P_n^{min})/t$ leads to adjustments in weight value ω , allowing algorithm to search with a maximum number of size. The advantage of adaptive weight adjustment method is determined by pair phases. In initial phase, population changes when number of iterations is too large. In final phase, population changes when it falls into a local optimum. A total of 115 features are processed in IDS to select relevant features, resulting in a count of 95 features out of 115. Then, best 10 features are considered in IDS, with 6 features being fed into classification of IDS in cybersecurity.

3.5 Classification

The selected features are classified using LSTM model to identify botnet attacks, which proves effective at achieving high detection accuracy in a short amount of time [23]. There are many benefits from using DL classifiers with the LSTM model to enhance the final results. The massive amount of standard deviation data be quickly accessed to classify, and the LSTM's capacity to effectively learn from longer data sequences is considered. LSTM classification is involved in improving IDS in cybersecurity, and using this method to analyse sequential data from the IDS helps improve security, making it an essential component of cybersecurity. The LSTM network performs better training effects on intrusion detection due to its ability to maintain long-term memory and solve gradient disappearance caused by Recurrent Neural Networks (RNNs). This method replaces neurons in RNNs and utilizes three phases: input, output, and forget gates, which allow passing of information and maintain memory ability of long data sequences, solving dependence problem in RNN networks.

FORGET GATE: This gate changes context, and forget gate is a sigmoid function with an area between 0 and 1 that is multiplied with cell state in final moment. The calculation formula expression is shown in Eq. (23).

$$h_t = \sigma(X_h.[f_{t-1}, w_t] + a_h) \quad (23)$$

INPUT GATE: The input gate supplements data needed for recent cell state. The output of input gate is a sigmoid function with a value level of 0 to 1, which is multiplied by recent cell state. The mathematical expression (24) and (25) is shown below:

$$n_t = \sigma(X_n.[f_{t-1}, w_t] + a_h) \quad (24)$$

$$\tilde{E}_t = \text{Tanh}(X_c.[f_{t-1}, w_t] + a_h) \quad (25)$$

The previews and recent state data was merged in final recent cell state. The mathematical Eq. (26) is shown below;

$$E_t = f_t \times E_{t-1} + n_t \times \tilde{E}_t \quad (26)$$

OUTPUT GATE: The final cell state, represented by $Tanh$ function, is outcome of output gate. These values are obtained through a sigmoid function, resulting in values between 0 and 1. The Eqs. (27) and (28) select information in outcomes.

$$O_t = \sigma(X_o \cdot [f_{t-1}, w_t] + a_o) \quad (27)$$

$$f_t = O_t \times Tanh(E_t) \quad (28)$$

The internal structure of hidden layer of LSTM network determines output and forget ratio of memory unit. $\tilde{E}_t$ represents current input data being inputted into memory unit, while E_t represents data to be input into memory unit. The LSTM detect complex and multi-stage attacks that involve subtle deviations from normal network behavior over time, which poses a challenge for signature-based IDS approaches. Leveraging their ability to capture long-term dependencies and learn complex patterns, LSTMs contribute to improving detection accuracy, reducing false positives, and enhancing resilience against evolving cybersecurity threats.

4 Experimental Results

In this research, AIW-ARO techniques is simulated using a python environment with system configuration of RAM: 16 GB, Processor: Intel core i7 and Operating system: Windows 10. The parameters like accuracy, precision, recall, and f1-score are utilized to estimating performance of model. The mathematical representation of these parameters is shown in (29), (30), (31), and (32) respectively.

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \quad (29)$$

$$Precision = \frac{TP}{(TP + FP)} \quad (30)$$

$$Recall = \frac{TP}{TP + FN} \quad (31)$$

$$F1 - score = 2 * \frac{(Precision * Recall)}{Precision + Recall} \quad (32)$$

where, TP , TN , FP , and FN illustrate True Positive, True Negative, False Positive, and False Negatives respectively. The proposed method, involving feature selection using AIW-ARO and classification using LSTM performed high accuracy to detect the attack.

4.1 Performance Analysis

In this section, proposed method involving feature selection and classification processes is evaluated using various performance metrics, including Accuracy, Precision, F1-score, and Recall, which are presented in and Table 3. The feature selection and classification

Table 2. Performance of feature selection on N-BaIoT dataset.

Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
PSO	95.86	95.63	95.82	95.20
BOA	96.45	96.42	96.27	96.30
GWO	97.25	97.63	97.63	97.75
ARO	98.45	98.75	98.15	98.12
AIW-ARO	99.99	99.99	99.99	99.99

process using N-BaIoT dataset is represented in Table 2 which describe feature selection and classification results.

The performance of N-BaIoT feature selection is evaluated based on accuracy, precision, F1-score, and recall on N-BaIoT dataset, as described in Table 2. The existing methods using feature selection techniques such as PSO, Bat Optimization Algorithm (BOA), Grey Wolf Optimizer (GWO) and TDO are evaluated. The LFS-TDO method achieves high accuracy of 99.99%, precision of 99.99%, recall of 99.99%, and F1-score of 99.99%. The feature selection technique AIW-ARO achieves high accuracy, reaching 99.99%, because it extracts related features that easily detect attacks.

Table 3. Performance of classification on N-BaIoT dataset.

Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
RNN	95.86	95.63	95.82	95.20
CNN	96.45	96.42	96.27	96.30
DNN	97.25	97.63	97.63	97.75
DCNN	98.45	98.75	98.15	98.12
LSTM	99.99	99.99	99.99	99.99

The performance of LSTM classification is evaluated based on accuracy, precision, F1-score, and recall on N-BaIoT dataset, as described in Table 3. The existing methods using feature selection techniques such as RNN, CNN, DNN, and DCNN are also evaluated. The LSTM method achieves high accuracy, precision, recall and F1-score of 99.99%. The classification technique is LSTM, achieving high accuracy reaching 99.99% because of its robustness and efficiency.

4.2 Comparative Analysis

The performance of proposed method AIW-ARO is compared to existing methods, including RNN [16], CNN-LSTM [17], DNN-LSTM [18], and Bi-LSTM [19]. The comparative analysis involves datasets: N-BaIoT. In this research, proposed AIW-ARO

method achieves high accuracy, reaching 99.99% on N-BaIoT dataset. Table 4 describes a comparative analysis of proposed method.

Table 4. Comparative analysis of Proposed method

Authors	Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Prabhat Kumar <i>et al.</i> [16]	Bi-GRU	99.73	N/A	N/A	N/A
Ahsan Nazir <i>et al.</i> [17]	CNN-LSTM	N/A	99.98	99.96	99.97
Fraidoon Sattari <i>et al.</i> [18]	DNN-LSTM	99.98	99.97	99.87	99.87
Danish Javeed <i>et al.</i> [19]	Bi-LSTM	99.45	99.34	98.49	99.47
Proposed AIW-ARO and LSTM methods		99.99	99.99	99.99	99.99

Discussion

The advantages of the proposed method and limitation of existing methods are discussed in this section. The Bi-GRU [16] reliability of collected data does not guarantee because communication occurs across an insecure channel at low plane, and there are issues with gradient exploding or vanishing. CNN-LSTM [17] struggles to differentiate between high traffic processed attacks and malicious activity, leading to high false positives and class imbalance, and it requires large datasets for training. The DNN-LSTM [18] involve high computational complexity, the need for large amounts of labelled data, a lack of interpretability, and require more resources. The Bi-LSTM [19] was highly complex and time-consuming due to the need to process data in both forward and backward directions. The proposed method AIW-ARO method, enhance early stage of algorithm iteration by assigning relatively diffuse population large weight values to enhance global search ability of algorithm.

5 Conclusion

Cybersecurity is practice of preventing cyberattacks on vital infrastructure and private information. In this research, AIW-ARO is proposed for IDS to secure entire network in cybersecurity. In IDS, pre-processing techniques such as Standard Scaling prevent attacks by separating the features and then using mean and standard deviation to make them less sensitive. The SMOTE generates synthetic sample for minority class thus addressing class imbalance issue and improve ability of IDS to detect unseen attacks. Feature selection using AIW-ARO method selects relevant features and hides information to secure data, while foraging serves as an exploration strategy, preventing rabbits

from being detected by cybersecurity measures. Finally, classification using LSTM technique identifies attacks and classifies IDS accurately, effectively in cybersecurity. The proposed AIW-ARO technique is evaluated using N-BaIoT datasets, achieving a high accuracy of 99.99%. This is compared to existing techniques like Bi-GRU, CNN, and CNN-LSTM. Future work will focus on enhancing false positives by considering feature selection, further improving classification results and accurately identifying various attacks.

References

1. Alqahtani, M., Mathkour, H., Ben Ismail, M.M.: IoT botnet attack detection based on optimized extreme gradient boosting and feature selection. *Sensors* **20**(21), 6336 (2020)
2. Liao, N., Guan, J.: Multi-scale convolutional feature fusion network based on attention mechanism for IoT traffic classification. *Int. J. Comput. Intell. Syst.* **17**(1), 36 (2024)
3. Cao, Y., Wang, Z., Ding, H., Zhang, J., Li, B.: An intrusion detection system based on stacked ensemble learning for IoT network. *Comput. Electr. Eng.* **110**, 108836 (2023)
4. Kandhro, I.A., et al.: Detection of real-time malicious intrusions and attacks in IoT empowered cybersecurity infrastructures. *IEEE Access* **11**, 9136–9148 (2023)
5. Szymanski, T.: A software-defined deterministic internet of things (IoT) with Artificial Intelligence (AI) for quantum-safe cyber-security. *Authorea Prepr.* (2023)
6. Al Razib, M., Javeed, D., Khan, M.T., Alkanhel, R., Muthanna, M.S.A.: Cyber threats detection in smart environments using SDN-enabled DNN-LSTM hybrid framework. *IEEE Access* **10**, 53015–53026 (2022)
7. Razikin, K., Soewito, B.: Cybersecurity decision support model to designing information technology security system based on risk analysis and cybersecurity framework. *Egypt. Inf. J.* **23**(3), 383–404 (2022)
8. Agnew, D., Boamah, S., Mathieu, R., Cooper, A., McNair, J., Bretas, A.: Distributed software-defined network architecture for smart grid resilience to denial-of-service attacks. In: 2023 IEEE Power & Energy Society General Meeting (PESGM) (pp. 1–5). IEEE (2023)
9. Ndife, A.N., Mensin, Y., Rakwichian, W., Muneesawang, P.: Cyber-security audit for smart grid networks: an optimized detection technique based on bayesian deep learning. *J. Internet Serv. Inf. Secur.* **12**(2), 95–114 (2022)
10. Starke, A., et al.: Cross-layered distributed data-driven framework for enhanced smart grid cyber-physical security. *IET Smart Grid* **5**(6), 398–416 (2022)
11. Symakesis, A.D., Alcaraz, C., Hatziargyriou, N.D.: Classifying resilience approaches for protecting smart grids against cyber threats. *Int. J. Inf. Secur.* **21**(5), 1189–1210 (2022)
12. Kure, H.I., Islam, S., Mouratidis, H.: An integrated cyber security risk management framework and risk predication for the critical infrastructure protection. *Neural Comput. Appl.* **34**(18), 15241–15271 (2022)
13. Singh, K., Grover, S.S. and Kumar, R.K., 2022, June. Cyber security vulnerability detection using natural language processing. In 2022 IEEE World AI IoT Congress (AIIoT) (pp. 174–178). IEEE
14. Jiang, Y., Jeusfeld, M.A., Ding, J., Sandahl, E.: Model-based cybersecurity analysis: extending enterprise modeling to critical infrastructure cybersecurity. *Bus. Inf. Syst. Eng.* **65**(6), 643–676 (2023)
15. Masi, M., Sellitto, G.P., Aranha, H., Pavleska, T.: Securing critical infrastructures with a cybersecurity digital twin. *Softw. Syst. Model.* **22**(2), 689–707 (2023)

16. Kumar, P., Kumar, R., Aljuhani, A., Javeed, D., Jolfaei, A., Islam, A.N.: Digital twin-driven SDN for smart grid: A deep learning integrated blockchain for cybersecurity. *Sol. Energy* **263**, 111921 (2023)
17. Nazir, A., et al.: A deep learning-based novel hybrid CNN-LSTM architecture for efficient detection of threats in the IoT ecosystem. *Ain Shams Eng. J.* **15**(7), 102777 (2024). <https://doi.org/10.1016/j.asej.2024.102777>
18. Sattari, F., Farooqi, A.H., Qadir, Z., Raza, B., Nazari, H., Almutiry, M.: A hybrid deep learning approach for bottleneck detection in IoT. *IEEE Access* **10**, 77039–77053 (2022)
19. Javeed, D., Gao, T., Khan, M.T., Shoukat, D.: A hybrid intelligent framework to combat sophisticated threats in secure industries. *Sensors* **22**(4), 1582 (2022)
20. Rey, V., Sánchez, P.M.S., Celdrán, A.H., Bovet, G.: Federated learning for malware detection in IoT devices. *Comput. Netw.* **204**, 108693 (2022)
21. N-BaIoT dataset: <https://www.kaggle.com/datasets/mkashifn/nbaiot-dataset>. Accessed on May2024
22. Bakhsh, S.A., Khan, M.A., Ahmed, F., Alshehri, M.S., Ali, H., Ahmad, J.: Enhancing IoT network security through deep learning-powered Intrusion Detection System. *Internet Things* **24**, 100936 (2023)
23. Du, J., Yang, K., Hu, Y., Jiang, L.: NIDS-CNNLSTM: network intrusion detection classification model based on deep learning. *IEEE Access* **11**, 24808–24821 (2023)



Deepfake Video Detection: A CSP-Dencenet and LSTM Based Approach

Ayush Bajpai^(✉) and Ritu Garg

National Institute of Technology, Kurukshetra, India
ayushbajpai324.ab.ab@gmail.com

Abstract. The proliferation of deepfake videos poses a significant threat to the integrity of digital content and challenges the reliability and authenticity of the social media content. With the rise of sophisticated deepfake technologies, the detection of manipulated videos has become very challenging. Also, this has sparked worries about the security dangers posed by deepfakes. The identification and classification of image forgeries have emerged as critical issues in the field of multimedia forensics. When tested across different datasets, current deepfake detectors frequently struggle to retain their performance, even when faced with high-quality faces in particular datasets. To overcome such, we proposed an ensemble approach for deepfake detection using the power of enhanced convolution neural network i.e. CSP-Dencenet and LSTM network, a hybrid multi-task strategy to enhance feature maps for tasks related to deepfake detection and classification. CSP-DenceNet is introduced as a foundational element, combining DenceNet's efficiency with the Cross Stage Partial (CSP) attention mechanism. This architecture is adept at extracting discriminative features crucial for identifying subtle manipulations inherent in deepfake videos. The fusion of CSP-DenceNet with LSTM further enhances the model's capacity to discern temporal dependencies within video sequences, effectively capturing dynamic patterns indicative of deepfake content. Through comprehensive experimentation on benchmark dataset ie. Faceforensic++, the efficacy of the proposed hybrid model is validated, showcasing superior performance compared to conventional detection methods in detecting diverse deepfake contents i.e. facial swaps, lip-sync alterations, and facial reenactments etc. The proposed method is robust under different compression factors and provides state-of-the-art performance with 96.0% classification accuracy on the FaceForensics++ datasets, based on experimental results.

Keywords: Deepfake, Face-forgery · Face-swap · Face Forensics

1 Introduction

Altering one's face is typically a tedious and time-consuming process that calls for both specialized equipment and expertise in video editing experience. We have observed rapid advancement in the area of image synthesis with the advancement of deep learning and improvement of computer hardware. "Seeing is not always believing" is partially realized because of the massive proliferation of face-altering videos, like Deepfake, on

social media and video sharing platforms, as well as the rise in papers published on their construction and detection over the past year (see Fig. 1). The development of generative adversarial networks [4] and auto-encoders [3] has led to a growing use of these approaches to Deepfake, making it easier to produce and distribute high-quality video editing content. In several aspects, deepfake technology has improved computer vision, yet there are worries about the negative consequences of using it improperly [1]. Because Deepfake only needs a limited amount of face images to allow for face-swapping in videos, some malevolent individuals have exploited the internet’s data to produce a large number of phony videos. The face-swapping films of female and male celebrities circulating online, used for defamation are some examples of misuse of deepfake technology. It is standard practice to replace the faces of female stars in pornographic films and create fake video footage of politicians, corporate executives, and other notable people in order to manipulate public opinion, acquire selection, and affect stock prices. A significant risk to social stability, individual privacy, and national security is posed by a large number of fraudulent face-swapping movies [1]. In today’s digital world, confirming the legitimacy of videos shared online is steadily rising to the top of the conversation.



Fig. 1. Number of related papers from year 2017 to 2023 based on google scholar.

These days, the majority of photos and videos on social media are compressed. For instance, people who send videos using social media platforms like Facebook or Twitter have to compress the video before uploading because the upload file size is constrained. It would be challenging for us to identify fake content if someone disseminated compressed Deepfake footage. In order to tackle the issue, the forensic analysis of compressed Deepfake films turns into a significant and demanding undertaking. Many techniques for detecting Deepfakes have been put forth and have shown promising results. From a symmetry standpoint, meanwhile, the detection of Deepfakes remains fraught with difficulties, as most detectors exhibit a sharp decline in detection performance during compression attacks as a result of the loss of picture feature information. Furthermore, the majority of techniques perform poorly on the latest Deepfake techniques.

Therefore, there is still a need to improve the detector’s effectiveness in intricate settings with extremely convincingly manipulated videos. This paper presents a hybrid model with a preprocessing module that is based on Long Short-Term memory and CSP-DenceNet, considering the aforementioned challenges. Conversely, identity or face

swapping is the act of replacing one person's visage with that of another [5]. In addition to the ability to effectively persuade people to agree with one's viewpoint without using words, a well-crafted facial expression can also simulate someone's physical presence in a setting where they were not. It is crucial to safeguard the integrity and moral standards of multimedia material by developing strong and trustworthy face forgery forensics in order to effectively counteract these deepfakes methods (Fig. 2).



Fig. 2. The FF++ dataset contains the original, Deepfakes, FaceSwap, Face2Face and Neural-Texture expression swap videos.

The following are paper's primary contributions:

- Building a preprocessing module that preserves high-frequency signals in photos while filtering and ignoring low-frequency signals in order to improve the discrimination between actual videos and Deep Fake-generated videos.
- By combining the CSP-DenceNet with the preprocessing module, we offer a novel method for detecting Deepfakes. The ablation experiment validates the preprocessing module's effectiveness. Even under severe compression, it retain strong resilience, and our method's accuracy is still greater than 95%.
- We validate our method's performance using FaceForensic++ datasets. Comprehensive experimental assessments show that the suggested approach operates effectively on latest manipulation techniques.

This is how the remainder of the paper is structured: The situation of the arts is thoroughly summarized in Section 2. Section 3 presents the detection approach that is suggested in this paper; Section 4 describes the dataset that is utilized; and Section 5 provides an example of the numerical findings on the benchmark datasets. We wrap up this paper in Section 6.

2 Related Work

In recent years, deepfake detection has attracted a lot of interest due to the escalating concerns surrounding the proliferation of synthetic media. Numerous studies have proposed various approaches to tackle this challenge, using a wide range of strategies, from deep learning-based approaches to conventional image processing. There are two categories into which the current detection methods can be categorized based on distinct extraction of feature: detection techniques for deepfake videos using deep learning methods and detection techniques for deepfake videos using handcrafted methods.

2.1 Detection Technique for Deepfake Videos Using Handcrafted Methods

The hand-crafted features-based Deepfake video detection systems mostly rely on certain tampering traces. The process of detection involves the extraction of frequency domain features and statistical features that exhibit notable variations in videos. Examples of these features include optical flow [6], light response non-uniformity [6], and image quality assessment. Certain distinctive picture feature information is frequently retained in the video during synthesis and post-processing. Consequently, Deepfake video detection can be accomplished using some conventional image forensics techniques. Researchers must first manually extract feature information from face photographs with improved discrimination in order to build a detection model for such techniques. After that, they train a classifier using the feature information. Because the generation method of real videos differs significantly from that of Deepfake videos, It was Koopman et al. [7] who first suggested using PRNU to detect Deepfake movies. Following the division of the facial region of each frame into eight groups by cropping it, they determined the average PRNU value for each group. The average normalized correlation coefficient between the Deepfake and genuine videos differed significantly, according to the experimental data. According to Frank et al. [8], PRNU can successfully identify Deepfake films on limited datasets. The detection results don't hold up well when PRNU characteristics were used to train the detection model is evaluated on the GAN-generated image dataset. Faten et al. [9] used edge features for detection and fed the gathered features into a support vector machine (SVM) for training, and then extracted image texture feature information using a variety of image feature point detectors. The detection model has above 90% accuracy on the small-scale dataset UADFV. While standard image-based forensics technology is well developed and capable of producing high-quality detection outcomes, in certain cases it produces subpar Deepfake films. The original visual features would be lost when the faked content was subjected to noise reduction, compression, and blurring. The detection rate will decrease because the detection model that was trained using these features is readily circumvented.

2.2 Detection Technique for Deepfake Videos Using Deep Learning Methods

The deep learning-based Deepfake video detection technique depends on neural networks' potent learning capabilities and an ever-richer sample collection. The two types of these techniques are further separated into frame-level detection techniques and video-level detection techniques. The primary focus of frame-level detection techniques is on the information contained in a single frame of an image. Various network architectures are created for feature analysis. The video-level detection techniques take into account temporal aspects between frames and concentrate on intra-frame information.

2.2.1 Frame-Level Methods for Detection

A two-stream network is suggested by Zhou et al. [10] that analyzes steganalysis characteristics using a patch-based triplet network and face tampering artifacts using a GoogleNet. Similar to this, Meso-4 and MesoInception-4 networks are suggested by Afchar et al. [11]. Mesoscopic aspects of the image are used to detect deepfakes. The

author's Deepfake dataset serves as the training and testing dataset for the approach, which yields good detection results. Using the FF++ dataset, Rossler et al. [12] identified Deepfake movies using the Xception network. Furthermore, the ResNet-50-based Spatial Pyramid Pooling layer was presented by Li et al. [13]. This module is more suited to concentrate on the alteration of the facial area's resolution. In order to identify Deepfake movies, Nirkin et al. [14] suggest using Two-Networks to determine the different signal properties of face-swapping areas. Typically, deepfake detection is modeled as a general binary classification issue in detection techniques. Zhao et al. [15] consider Deepfake detection as a fine-grained classification problem.

2.2.2 Video-Level Methods for Detection

Convolutional neural networks are capable of extracting features analyzing images, but these techniques frequently disregard some inter-frame features rather than solely just focusing on intra-frame feature information. Rich inter-frame features are present in the video as compared to still photographs. Recurrent Neural Networks (RNNs) are the first method Guera et al. [16] suggest employing to identify Deepfake films because of the temporal continuity between frames. Because the phony movies' face images are created frame by frame, previous images are not taken into consideration by the auto-encoder while creating new ones, which frequently results in many abnormalities in some locations between the earlier and later frames. To address the aforementioned issue, according to Monsterrat et al. [17] a CNN-RNN network was constructed using an autonomous weighting method. To strengthen the detection algorithm's robustness at the video-level, the framework gives face pictures more weight.

The outcomes of the experiment demonstrate that learning image-specific feature information via the network is more effective and efficient. In order to avoid overfitting, the dual-stream networks prune the network after examination of the intra-frame temporal characteristics and the intra-frame noise attributes respectively. Agarwal et al. [18] found that the "micro-expression" characteristic can be used to identify Deepfake movies and that people's facial action units vary. Nineteen head action and facial expression units are chosen for their trial. Openface2 is used to extract the target person's head action unit and facial expression unit's presence intensity in the video. The SVM classifier is then trained using the obtained presence intensity as a feature set. Wu et al. offer a fresh approach to Deepfake detection to address the aforementioned issue. Combining spatial and steganalytic information allows the method to detect tampered faces with strong generalization capacity. The other branch of the algorithm is utilized to conceal facial content. It performs well on cross-tested across multiple datasets. Deep learning-based detection techniques are vulnerable to overfitting during the model-training phase. Thus, by preprocessing, researchers take advantage of the intrinsic distinctions between authentic and fraudulent movies.

3 Proposed Method

Attention-based localization and classification techniques used in our approach to generate the probability of an input video being manipulated or real, in contrast to single-objective approaches. It involves preprocessing the videos using the preprocessing module, training the baseline network with the processed videos, and lastly receiving the projected probability of each video from the detection model. Figure 3 depicts the workflow for our proposed Deepfake detection technique.

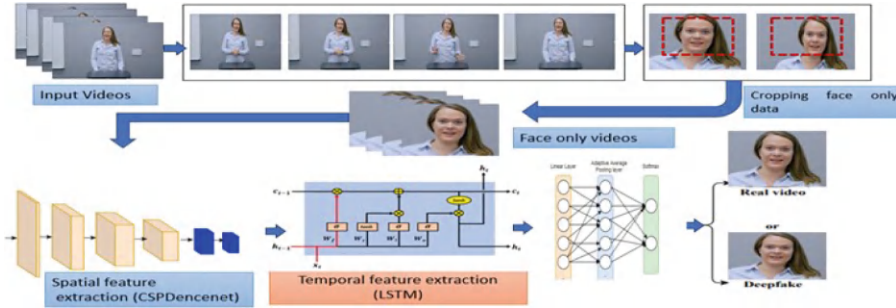


Fig. 3. Proposed Methodology for DeepFake Video Detection

3.1 Preprocessing

The preprocessing of raw dataset is a foundational step in the development of a robust deep learning model. By systematically organizing and refining the input data, we aim to enhance the quality and relevance of features used in subsequent stages of the algorithm. Starting with the extraction of frames from raw video, each frame becomes a temporal snapshot, providing the basis for further analysis. The subsequent reconstruction of videos containing only facial regions is not merely a dimensionality reduction technique but a strategic maneuver to amplify the emphasis on facial features during model training. An indispensable aspect of preprocessing is the elimination of corrupted videos. The dataset is divided into training and testing subsets in ratio of 70:30 to assess the model's generalization to unseen data accurately. Setting a minimum threshold of 100 frames ensures that the dataset consists of videos with sufficient duration, providing the model with the necessary context to discern temporal patterns and authenticate or flag potentially manipulated content.

3.2 Feature Extraction

Extracting features from the videos that contain only cropped faces data is the second step in the procedure. Most of the previous works relied on the use of one of the CNN networks pretrained on 'imagenet', whether using CNN only to extract visual features. CSP-DenceNet is a hybrid convolutional neural network (CNN) architecture that combines

the Cross Stage Partial (CSP) attention mechanism with the efficiency of DenceNet for feature extraction from videos. The input frames are passed through convolutional layers to extract spatial features.

Layers	Output Size	CSP DenseNet-121
Convolution	112×112	
Pooling	56×56	
Dense Block (1)	56×56	$\left[\begin{array}{c} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{array} \right] \times 6$
Transition Layer (1)	56×56	
	28×28	
Dense Block (2)	28×28	$\left[\begin{array}{c} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{array} \right] \times 12$
Transition Layer (2)	28×28	
	14×14	
Dense Block (3)	14×14	$\left[\begin{array}{c} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{array} \right] \times 24$
Transition Layer (3)	14×14	
	7×7	
Dense Block (4)	7×7	$\left[\begin{array}{c} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{array} \right] \times 16$
Classification Layer	1×1	

Fig. 4. CSP-Dencenet-121 architecture for Proposed Methodology for DeepFake Video Detection

These layers apply convolutional filters to the input frames, capturing spatial patterns and features at different scales. The CSP attention mechanism is integrated into the network to enhance feature extraction. It operates by splitting the feature maps into two parts, processing each part separately, and then recombining them. This allows the network to focus on informative regions while suppressing non relevant information, improves the discriminative power of the network. CSPDencenet-121 is explained in Fig. 4.

A dense block and a transition layer are present in every DenseNet stage, and each dense block contains k dense layers. Concatenating the output and input of the i^{th} dense layer will become the input of the $(i + 1)^{\text{th}}$ dense layer.

$$\begin{aligned}
 x_1 &= w_1 * x_0 \\
 x_2 &= w_2 * [x_0, x_1] \\
 &\vdots \\
 &\vdots \\
 &\vdots \\
 &\vdots \\
 x_2 &= w_k * [x_0, x_1, \dots, x_{k-1}]
 \end{aligned} \tag{1}$$

where $[x_0, x_1, \dots]$ denotes concatenating $x_0, x_1, \dots$. And x_i and w_i stand for the output and weights of the i^{th} dense layer, respectively. The convolution operator is represented by $*$. The weight updating equations can be expressed as follows if a backpropagation

technique is used to update the weights:

$$\begin{aligned}
 W'_1 &= f(w_1, g_0) \\
 W'_2 &= f(w_1, g_0, g_1) \\
 &\vdots \\
 &\vdots \\
 &\vdots \\
 W'_k &= f(w_k, g_0, g_1, \dots, g_k)
 \end{aligned} \tag{2}$$

where the gradient propagated to the i^{th} dense layer is represented by g_i , and f represents the weight update function. It is evident that many gradient data points are repurposed when adjusting the weights of various dense layer combinations. This will lead to recurrent learning of copied gradient information by several dense layers.

Partially dense blocks and partially transition layers make up each CSPDenseNet stage. In a partial dense block, the base layer feature maps in a stage are divided into two sections through channel $x_0 = [x_{0'}, x_{0''}]$. Between $x_{0'}$ and $x_{0''}$, The latter will pass through a dense block, while the former is immediately connected to the stage's conclusion. The following are the stages that go into creating a partial transition layer: First, the output of dense layers, $[x_{0'}, x_1, \dots, x_k]$, will go through a layer of transition. Secondly, this transition layer's output, x_T , will be concatenated with $x_{0'}$ and go through an additional layer of transition before producing output x_U . The CSPDenseNet weight update and feed-forward pass formulae are displayed, respectively.

$$\begin{aligned}
 x_k &= w_K * [x_{0'}, x_1, \dots, x_{k-1}] \\
 x_T &= w_T * [x_{0'}, x_1, \dots, x_K] \\
 x_U &= w_U * [x_{0'}, x_T]
 \end{aligned} \tag{3}$$

$$\begin{aligned}
 w'_k &= f(w_K, g_{0'}, g_1, \dots, g_{k-1}) \\
 w'_T &= f(w_T, g_{0'}, g_1, \dots, g_k) \\
 w'_U &= f(w_U, g_{0'}, g_T)
 \end{aligned} \tag{4}$$

The gradients originating from the dense layers are evidently incorporated independently. On the other hand, the feature map $x_{0'}$. Whatever did not pass through the dense layers is additionally incorporated individually. As the gradient data used to update weights, neither side has any redundant gradient data that comes from another side. So CSPDenseNet prevents an excessive quantity of duplicate gradient information while maintaining the benefits of DenseNet's feature reuse properties by truncating the gradient flow.

The sequence of feature maps obtained from the spatial feature extraction stage serves as the input to the LSTM component within the architecture. The LSTM is responsible for capturing temporal dependencies within this sequence of feature maps, enabling the network to analyze the temporal evolution of features across frames. The LSTM component typically consists of multiple temporal convolutional layers, each processing a subset of the feature maps from the spatial feature extraction stage. These convolutional layers are designed to operate across time, allowing the network to extract temporal patterns and dynamics present in the input sequence of feature maps.

3.3 Classification

The final step is the classification process where a video is classified as a fake or real depending on the extracted features explained in step 3.2. A set of fully connected layers with dropout layers with a Sigmoid function used in the last layer form the structure of the classifier. This will give a final prediction that that input video data is tampered or not.

4 Dataset

The FaceForensics++ [19] dataset is a widely used benchmark dataset specifically designed for deepfake detection research and evaluation. It is a comprehensive dataset that contains 1000 original videos and corresponding 5000 manipulated videos created using various face manipulation techniques. The dataset contains videos collected from YouTube and other spanning sources consisting of a wide range of subjects and scenarios. Original videos are then manipulated using five different methods: Deepfakes, FaceSwap, NeuralTextures, FaceShifter, and Face2Face. Each manipulation technique aims to generate realistic and convincing facial manipulations, simulating deepfake scenarios. Each video in the dataset is annotated with metadata labels indicating whether it is real or manipulated.

5 Results

Existing methods in literature only focuses on single type of inconsistency, either spatial or temporal, however, we proposed a hybrid approach that focus on both at the same time, so more robust as compare to existing methods. After preprocessing face only data is the input for the model which reduces the overall size of data ultimately leads to light weight model. The deep learning platform is used for the implementation. The performance of the proposed approach is evaluated by comparing it with the existing approaches i.e. Mesonet [11], Exceptionnet [10], Inceptionnet [15], VGG16 [6].

Evaluation Metrics: The evaluation process involves analysis of a confusion matrix to provide a comprehensive assessment of the model’s ability to discriminate between authentic and manipulated videos. It categorizes predictions into four key outcomes: true positives (authentic videos correctly identified), true negatives (manipulated videos correctly identified), false positives (authentic videos incorrectly identified as manipulated), and false negatives (manipulated videos incorrectly identified as authentic) (Table 1).

Table 1. Confusion matrix

	Predicted Authentic	Predicted Manipulated
Actual Authentic	True Positives (TP)	False Negatives (FN)
Actual Manipulated	False Positives (FP)	True Negatives (TN)

Accuracy, precision, recall, and F1 score are among the standard classification metrics that are calculated using the ground truth labels and the model's predictions. These metrics offer a thorough grasp of the model's advantages and disadvantages in terms of distinguishing between real and altered videos. Specific emphasis is placed on minimizing false positives, as misclassifying authentic videos as manipulated could have severe consequences.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (5)$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad (6)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (7)$$

$$\text{F1 Score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (8)$$

This iterative evaluation approach allows us to refine and enhance the model, ensuring its effectiveness in addressing the evolving landscape of deepfake manipulation techniques. We employed the Adam optimizer, which has a learning rate of 0.001.

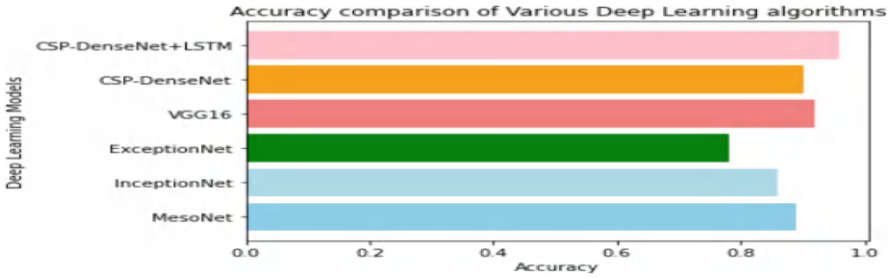


Fig. 5. Comparison of accuracy of various Deep Learning Models

The findings presented herein are the culmination of meticulous investigations conducted across various dimensions of our study. Table 2 provides a comprehensive comparison of the values associated with all the evaluation metrics utilized in determining the optimal model for our work. After an exhaustive comparative analysis of various Deep Learning (DL) models applied to our dataset, it is discerned that CSP-Dencenet+LSTM emerges as the most proficient model out of the above.

Figure 5 shows the accuracy comparison among various deep learning models involved in our research. In the pursuit of identifying the optimal DL architecture for the dataset. After observation of the result we found that CSP-Dencenet + LSTM works superior as compared to other applied models (Figs. 6, 7, 8, 9, 10 and 11).

Table 2. Comparison of different models based on consider matrices

SNO	Model Applied	Accuracy	Precision	Recall	F-1 Score
1	MesoNet Model [11]	0.89	0.85	0.85	0.88
2	ExceptionNet Model [10]	0.78	0.81	0.83	0.82
3	InceptionNet Model [15]	0.86	0.83	0.85	0.84
4	Vgg16 Model [6]	0.92	0.93	0.93	0.91
5	CSP-DenseNet [20]	0.90	0.89	0.90	0.90
6	CSP-DenseNet + LSTM	0.96	0.96	0.95	0.96

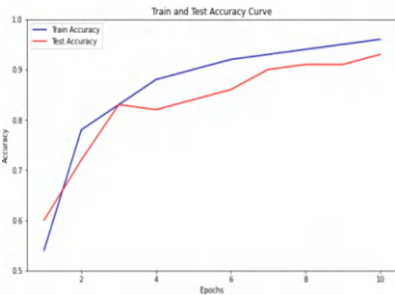


Fig. 6. Train and test accuracy for CSPDencenet-121 + LSTM Model

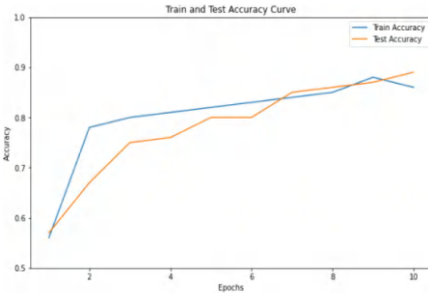


Fig. 7. Train and test accuracy for CSP-Dencenet Model

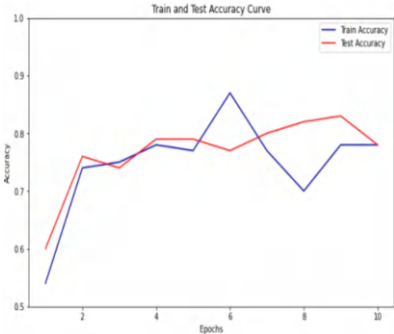


Fig. 8. Train and test accuracy for ExceptionNet Model

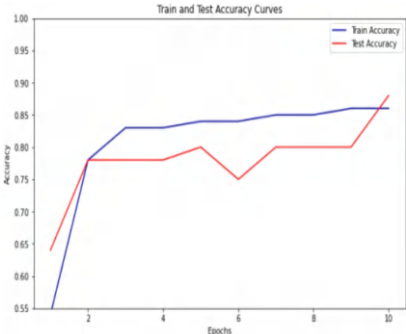


Fig. 9. Train and test accuracy for InceptionNet Model

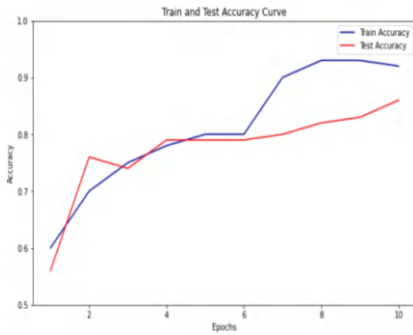


Fig. 10. Train and test accuracy for VGG16 Model

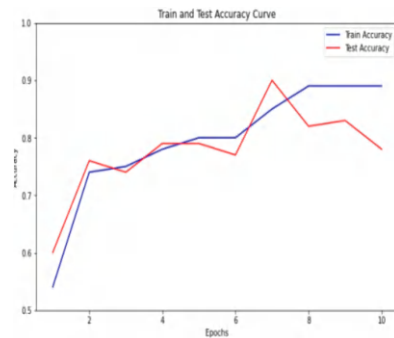


Fig. 11. Train and test accuracy for MesoNet Model

6 Conclusion

In conclusion, the presented research introduces a robust deepfake detection methodology, leveraging the synergies between CSPDencenet-121 and LSTM networks. The amalgamation of spatial features extracted by CSP-Dencenet with the temporal contextual understanding of LSTM has demonstrated remarkable efficacy in discerning between authentic and manipulated visual content. The achieved accuracy consistently surpassing the 95% underscores the reliability of our proposed approach across diverse deepfake scenarios. The results not only contribute to the advancement of deepfake detection technology but also highlight the importance of addressing the evolving challenges posed by synthetic media. Through meticulous experimentation and analysis, this work reinforces the commitment to fostering a digital environment built on transparency, integrity, and the responsible use of emerging technologies. As already discussed, robustness, generalizability, and computational efficiency are necessary for effective deepfake detection techniques to be used in the real world. Thus, In future, the proposed model can be enhanced considering the multiple faces, hence facilitating greater generalization.

References

1. Nguyen, T.T., et al.: Deep learning for deepfakes creation and detection: a survey. *Comput. Vis. Image Underst.* **223**, 103525 (2022)
2. O'Shea, K., Nash, R.: An introduction to convolutional neural networks (2015). arXiv preprint [arXiv:1511.08458](https://arxiv.org/abs/1511.08458)
3. Kingma, D.P., Welling, M.: Auto-encoding variational bayes (2013). arXivpreprint [arXiv:1312.6114](https://arxiv.org/abs/1312.6114)
4. Goodfellow, I., et al.: Generative adversarial networks. *Commun. ACM.* **63**(11), 139–144 (2020)
5. Li, Y., Lyu, S.: Exposing deepfake videos by detecting face warping artifacts (2018). arXiv preprint [arXiv:1811.00656](https://arxiv.org/abs/1811.00656)
6. Amerini, I., Galteri, L., Caldelli, R., Del Bimbo, A.: Deepfake video detection through optical flow based cnn. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops* (p. 0) (2019)

7. Koopman, M., Rodriguez, A.M., Geradts, Z.: Detection of deepfake video manipulation. In: The 20th Irish Machine Vision and Image Processing Conference (IMVIP) pp. 133–136 (2018, August)
8. Frank, J., Eisenhofer, T., Schönherr, L., Fischer, A., Kolossa, D., Holz, T.: Leveraging frequency analysis for deep fake image recognition. In: International Conference on Machine Learning (pp. 3247–3258). PMLR (2020)
9. Kharbat, F.F., Elamsy, T., Mahmoud, A., Abdullah, R.: Imagefeature detectors for deepfake video detection. In: 2019 IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA) (pp. 1–4). IEEE (2019, November)
10. Zhou, P., Han, X., Morariu, V.I., Davis, L.S.: Two-stream neuralnetworks for tampered face detection. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) (pp. 1831–1839). IEEE (2017, July)
11. Afchar, D., Nozick, V., Yamagishi, J., Echizen, I.: Mesonet: acompact facial video forgery detection network. In 2018 IEEE International Workshop on Information Forensics and Security (WIFS) (pp. 1–7). IEEE (2018, December)
12. Rossler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., Nießner, M.: Faceforensics++: learning to detect manipulated facial images. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1–11 (2019)
13. Li, Y., Yang, X., Sun, P., Qi, H., Lyu, S.: Celeb-df: a large-scale challenging dataset for deepfake forensics. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3207–3216 (2020)
14. Nirkin, Y., Wolf, L., Keller, Y., Hassner, T.: Deepfake detection basedon discrepancies between faces and their context. *IEEE Trans. Pattern Anal. Mach. Intell.* **44**(10), 6111–6121 (2021)
15. Zhao, H., Zhou, W., Chen, D., Wei, T., Zhang, W., Yu, N.: Multiattentional deepfake detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2185–2194 (2021)
16. Güera, D., Delp, E.J.: Deepfake video detection using recurrent neural networks. In: 2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS). (pp. 1–6). IEEE (2018, November)
17. Montserrat, D. M., et al.: Deepfakes detection with automatic face weighting. In: Proceedings of the IEEE/CVF Conference on COMPUTER Vision and Pattern Recognition Workshops. pp. 668–669 (2020)
18. Agarwal, S., Farid, H., Gu, Y., He, M., Nagano, K., Li, H.: Protecting world leaders against deep fakes. In CVPR workshops (Vol. 1, p. 38) (2019, June)
19. GitHub - ondyari/FaceForensics: github of the FaceForensics dataset
20. Pang, G., Zhang, B., Teng, Z., Qi, Z., Fan, J.: Mre-net: multi-rate excitation network for deepfake video detection. *IEEE Trans. Circuits Syst. Video Technol.* **33**(8), 3663–3676 (2023). <https://doi.org/10.1109/TCSVT.2023.3239607>
21. Wang, C.Y., Liao, H.Y.M., Wu, Y.H., Chen, P.Y., Hsieh, J.W., Yeh, I.H.: CSPNet: a new backbone that can enhance learning capability of CNN. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. pp. 390–391 (2020)



Neural Networks in Cybersecurity: Applications for Predictive Threat Modeling

Ayisha Tabbassum¹(✉) , Arun Pandiyan Perumal² , Anirudh Khanna³ ,
Pradeep Chintale⁴ , and Madhavi Najana⁵

¹ Otis Elevator Co, Eden Prairie, MN, USA
ayishat@ieee.org

² Technology Infrastructure Specialist, Fremont, CA 94538, USA
apandiyan@hawk.iit.edu

³ Lead Solution Architect, SEI Investment Company, Downingtown, PA 19335, USA

⁴ Data Protection Lead, 900 Acadia Dr, Plano, TX 75023, USA

⁵ Software Engineering Manager, Columbus, OH, USA

Abstract. The emergence of AI as a revolutionary technology has been based on deploying neural networks (NNs) and deep learning techniques, which are considered the pillars of its functioning. Through the design of the human brain's structure and function, the NNs received an opportunity to execute tasks in an efficient way, for example, image recognition, natural language processing and decision-making. This paper studies the work of neural networks by predictive threat modeling in cybersecurity. It explains how a neural network is trained with the past cyber-incidence data to become the system that detects the patterns of current events and forecasts the future breaches and attacks. The AI service as a revolutionary technology, including NNs and deep learning approach, which are considered as its architecture can be called the key elements of the service. Through the linking of evolution and formation of a functional human brain, the NNs were indeed provided a platform for a smoothly running process which included the recognition of images, natural language processing and decision making. This essay will be directly targeted on the concept of brain-like networks and their use as a cyber security predictive model. Several neural network architecture variations, for example, feedforward neural networks, convolutional neural networks (CNNs), and recurrent neural networks (RNNs), which have unique capabilities and tasks, are demonstrated. After introducing the idea of learning in NNs, the paper thoroughly explains methods that let NNs get better with historical data—gradient descent and backpropagation. It then goes into detail about the critical role that training data plays and the training process itself, including issues like overfitting. Subsequently, we present the wide range of practical uses for NNs in several domains, illustrating how they have revolutionized computer science.

Keywords: Neural Networks · Cybersecurity · Predictive Threat Modeling · Artificial Intelligence (AI) · Predictive Threat Modeling · Cyber Threat Mitigation · Convolutional Neural Networks (CNNs) · Recurrent Neural Networks (RNNs)

1 Introduction

Artificial intelligence (AI) is a revolutionary force that has empowered the cognition of computer science enabling it to think of and solve complex dilemmas. AI, at its core, is a technology development of intelligent systems that are placed in a position to perceive, learn, reason, and make inevitable decisions in a similar or even more effective way than a human being can [1]. In the last ten years AI has been founded on these ideas and now it is being stimulated by the powerful increase of computational power, the availability of huge amounts of data and newly invented algorithms of machine learning. With the spread of the digital world in the present time, cyber-security has gained the paramount importance as the cyber threats and attacks become more sophisticated and complex. A new and advanced method of predictive threat modeling employing neural networks has embroiled itself in the cybersecurity field which is used to predict and be prepared for future attacks that are based on data history. This paper looks at how artificial intelligence called neural networks can be used in cyber defense to develop predictive models that learn from the past data provided to recognize patterns and estimate the probability of future cyber-attacks and potential outcomes. Alongside the key technologies behind this AI revival are neural networks (NNs), which look to the human brain and the way it performs its computations for inspiration. Many issues previously considered 'solvable' by NNs include threat recognition, natural language processing, decision-making and analytics. Neural networks, which were developed on account of an architectural and functional resemblance to the brain's, are able to perform detecting intricate cyber-threat patterns and relationships from data and, therefore, can be regarded as the indispensable instruments for processing and representing knowledge.

The resurgence of deep learning can be attributed, through multiple factors, such as: the development of more complex networks as convolutional neural networks and deep neural networks, the acquisition of powerful equipment like graphics processing units (GPUs) and tensor processing units (TPUs) [2]. On the other hand, the availability of large scale datasets and novel training techniques like transfer learning and reinforcement learning have substantially improved the power of neural networks to help in threat modeling. The focus of this paper will be in the area of the neural networks, their underlying mechanisms and the applications of this tool in the field of cybersecurity's predictive threat modeling. This undertaking is threefold: it takes a look at the theoretical fundamentals, the different architectural designs, and the real deployments of the tool which is a step towards building the knowledge of the computer science field professionals, researchers, and students. The topic around neural networks is going to be revealed in detail, including nodes-neurons, layers, connections, as well as weights and the different processes of learning, whether supervised, unsupervised or reinforced.

2 Artificial Neural Networks Biological Inspiration

A human brain is a hologram of innumerable centrally interconnected neurons that culminate in activities such as cognitive processes, emotion, and consciousness among many others. Brain is naturally the most intricate information processing system known to mankind having over 86 billion neurons and trillions of synaptic connections. It is this

end structure that has been the motivation for creating the artificial Neural Networks, a computational model which mimics the wonderful functionality of the human brain in the machines. A biological neuron, which is a highly specialized kind of cell, lies at the basis of the brain and performs the two major tasks: transmission and processing of electrochemical signals. Each neuron consists of three primary components: dendrite, a local plain and a long channel for sending the chemicals. As presented in Fig. 1, dendrites function as receivers gathering input from other neurons that are changed via synaptic connections [3].

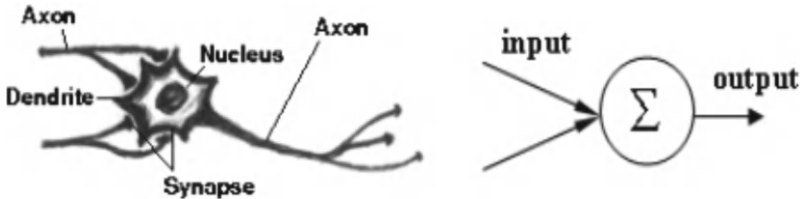


Fig. 1. Biological and Artificial Neural Network [3]

The soma, or cell body, is where the providers indicate the communication it receives and decide if one should fire an impulse. If the integrated signal exceeds the trigger points, the neuron makes an action potential that travels along the axon, a cable-like appendage that proceeds the impulse to other neurons via synaptic terminals. The extraordinary ability of the brain to learn, recognize patterns, and make decisions is found in the complex patterns of billions of such neurons [3]. The brain is the organ which enables this through a mechanism, called neural plasticity, which is based on an ongoing reshaping and reorganization of the synaptic connections [4]. The more a pathway is used, the stronger it becomes until it implicitly takes over the responsibilities for the memory required by the learning process. The ability to shape-shift makes the mind a tool suitable for learning new talents, retaining knowledge, and exploring more sophisticated ways of thinking, which, in turn, let us deal with the external world accordingly. Artificial intelligence researchers have been fascinated by this capacity for experience-based learning and adaptation. Inspired by the anatomical and functional principles of the brain, neural networks seek to emulate these extraordinary machine learning abilities.

Presenting input data and altered connection weights through learning algorithms, NNs can be taught to get better at a specific task and correspondingly correct their internal representations by refining these. This sort of mechanism is similar to the brain's neural plasticity, neuron pathways which develop/form or weaken/disappear due to repeated experiences and learning [5]. The effect of neural pathways forms the basis for an individual's specific abilities as well as behaviors. Neural networks mimicking our biological brain goes beyond mere structural similarities to ones that behave in the same way also. Scholars, in particular, have succeeded in breeding and constructing a wide array of neural networks each for detecting different intricacies of the cognizance process. Some CNNs (convolutional neural networks) do that very well (mimic) because they are like the visual cortex of the human brain [6]. Unlike deep neural networks, which deal with the spatial factors in brain's responses to visual information, recurrent neural networks

(RNNs) are well suited for the sequential data processing, and for example, become as alike as the brain's endowments of language and temporal presentation. In this way, computer science and machine learning have merged with the natural complexity of the neural systems to produce machine-based neural networks as one of the most powerful technologies for solving cyber threats where machines behave like biological systems that create, learn, and adapt. AI researchers applied philosophical thoughts about the brain, based on the architecture and learning mechanism understanding. This is the way to break the knowledge limits, to bring into existence decision-making, and to do artificial threat cognition.

3 Core Concepts of Artificial Neural Networks

The artificial neural network (ANN) is arranged around the artificial neuron as the most important computational unit that is conceptually similar to the biological neuron. Similar to how the neurons in the brain act as the fundamental constituents of the central nervous system, artificial neurons are the principal elements of ANNs [5]. Artificial neurons, with their architecture of interconnections in the neural network, can collectively work to process the data and learn from the information.

The artificial neuron function, in terms of sequence, can be summarized as the below stages. Initially, the neuron receives inputs, which take the form of signals from neurons also in the network. Here, these subsequent inputs are all symbolized as:

$$X_1, X_2, \dots, X_n \quad (1)$$

where n is the number of inputs (historical threat data).

Every input has a respective weight ($W_1, W_2, \dots, W_n$) attached to it. They are the weights that make the inputs of a neuron either more or less important in the final output of the neuron. The weighted sum of the inputs is calculated by multiplying each input by its respective weight and summing the products: The weighted sum of the inputs is calculated by multiplying each input by its respective weight and summing the products:

$$\text{Weighted Sum} = \sum (w_i * x_i) = w_1X_1 + w_2X_2 + \dots + w_nX_n \quad (2)$$

The cumulative effect of all the inputs on the activity of the neuron is represented by the above weighted sum (See Fig. 2).

Nevertheless, in order to achieve non-linearity and to make the network able to learn non-trivial patterns as well as relationships, a nonlinear function is put over the weighted sum. The function of activation transforms the weighted sum into the output (Predictive Threat) of the neuron, which is further passed to the next layers of neurons or serves as the final network output.

$$\text{Output} \equiv f(\sum(w_i * x_i)) \quad (3)$$

where the selected activation function is denoted by f .

Figure 2 shows the elements of an artificial neuron and their function. The input associates incoming signals (X_1 to X_n). For every input a corresponding weight (w_1 to

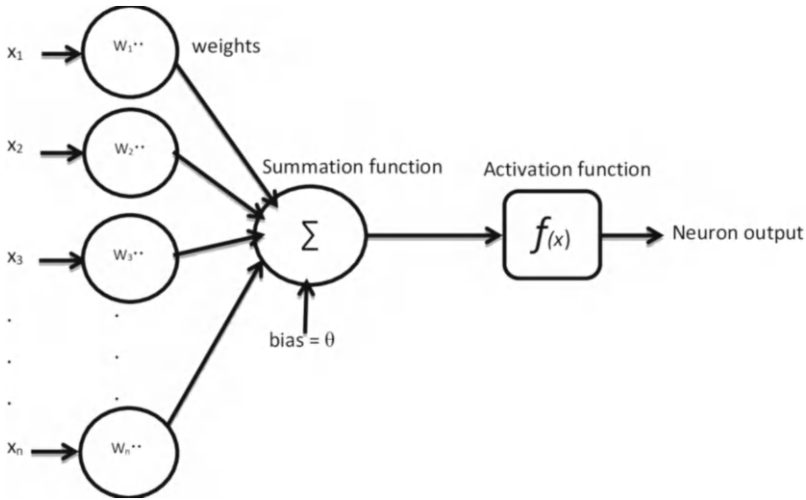


Fig. 2. Artificial Neuron Structure [7]

w_n) is multiplied and these weighted products are added together ($\sum w_i * x_i$) [7]. Next, activation function (f) adds the value obtained to the incoming messages from other neurons. These sums are then converted into output values that can be used as input by other layers in the network. Change of connections between artificial neurons makes the network able to process particular jobs.

The activation function, in fact, could be stated as the most important part of learning of the neural networks. It is a source of non-linearity, and thereby the network is able to identify and learn the complex, non-linear relationships that exist in the inputted data. It is crucial to note that different activation functions, each with unique properties and benefits, have been devised and used across different neural network architectures as presented in Table 1.

The Sigmoid activation function has been used in many neural network architectures for a very long time in threat detection. The output is in the form of a number with values ranging from 0 to 1 that makes it a good fit for the tasks of binary classification or probabilistic reasoning (see Table 1). However, it faces a vanishing gradient issue while training, and this may slow down or break the learning process.

The Rectified Linear Unit (ReLU) activation function has become quite famous in the last couple of years and especially in very deep neural networks. It is responsible for nonlinearity of a function and returns the input value if it's positive, and 0 otherwise. ReLU is specifically designed to solve gradient-vanishing problems in deep networks. This facilitates much faster network training. Besides, it is affected by the “dying ReLU” problem, in which neurons are inactive and stop learning.

Hyperbolic tangent (tanh) activation function is the first choice for most problems. It revolves around an output scale ranging from -1 to 1, hence, it is compatible for jobs that entail both positive and negative performance. Similar to the sigmoid, it will encounter vanishing gradient issues but to a lesser degree. The selection of activation function depends on the type of problem, network design, and expected result of the

Table 1. Common Activation Functions in Artificial Neural Networks

Function Name	Mathematical Formula	Description
Sigmoid	$f(x) = 1 / (1 + \exp(-x))$	S-shaped curve, outputs between 0 and 1. Commonly used in early NNs but can suffer from vanishing gradient problems
ReLU (Rectified Linear Unit)	$f(x) = \max(0, x)$	Outputs x if positive, 0 otherwise. A popular choice due to its computational efficiency and ability to address vanishing gradients
tanh (Hyperbolic Tangent)	$f(x) = (\exp(x) - \exp(-x)) / (\exp(x) + \exp(-x))$	S-shaped curve, outputs between -1 and 1. Offers a wider output range compared to sigmoid but can also be susceptible to vanishing gradients

output. In certain instances an interplay between various types of activation functions may be employed in various network layers with the purpose of getting the best possible result.

The use of the multiple ensemble of artificial neurons each with their own adjacency weights and activation functions helps the artificial neural networks to perceive the intricate patterns and relationships from the data; thus, the neural networks learn. The network's real training requires adjusting the weights' links between neurons by sets of data that have been divided into inputs and outputs based on which the network 'learns' to match input data with output results. This is basically the secret of their power and flexibility, i.e., the capacity to modify their connection weights and thus perform a variety of tasks ranging from image recognition and natural language processing to decision-making and predictive analysis. Artificial neural networks have become a potent tool for knowledge processing and representation, pushing the frontiers of artificial intelligence through the complex interactions of artificial neurons, weighted connections, and thoughtfully crafted activation functions.

4 Theoretical Foundations of Neural Networks for Predictive Threat Modeling

The theoretical bases of neural networks for predictive threat modeling in cybersecurity are based on their ability to analyze data and recognize intricate connections and ability. This power, however, derives from both structural/functional motivational factors of the human brain that outshine pattern recognition and data driven decision making [8]. A central part of neural network is the eponym of computational unit, an array of artificial

neurons that is able to imitate activities in the brain. Every single artificial neuron works inbound over copious signals that are summing up and being weighted. Consequently, the vector with its elements being multiplied by the weight is sent through an activation function, which allows for introducing nonlinearity and the ability of the neuron to capture the complex dependencies in the data [9]. The complexity of the artificial neural structures in the neural networks allows a complexity of data processing where each has its own input and computation, just like the human brain processes and encodes knowledge in a distributed manner. These relationships are governed by the weights that are changed during learning, improving the network’s operations on certain jobs [9]. In the sphere of the cybersecurity forecasting modeling, neural networks are trained on the historical cybersecurity breach data, which usually consists of the included features like vectors of attack, the vulnerability which are being exploited and the impact which the target systems or networks suffer [10, 11]. Through the studying of earlier instances of security incidents the neural network is able to establish relations and patterns that can be used for prediction of future attacks and breaches based on their characteristics.

A number of neural network architectures can be used for predictive Threat Modeling, each having specific strengths and capabilities distinct to them. For instance:

Feedforward Neural Networks: These nets have a feed-forward architecture, in which meaning originates and develops from input to output and steps do not refer to prior ones. They are suited especially for tasks concerning classification and regression - those used for assigning certain threats and determining their level of potential [12, 14].

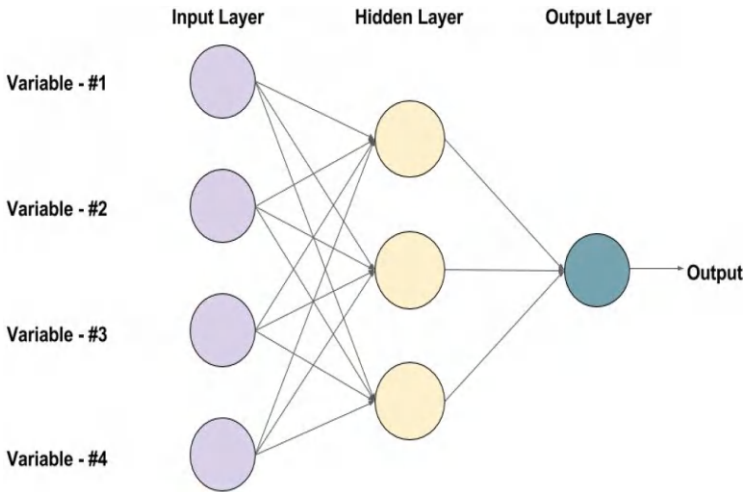


Fig. 3. Feedforward Neural Network [14]

From the diagram above, three levels make up the Feedforward Networks system: input, output, and hidden. Multiple hidden levels are often possible. Neural Networks’ fundamental processing unit, or neuron, is represented by each node in the layer.

Recurrent Neural Networks (RNNs): While RNNs, as opposed to the traditional ones, are made to deal with sequential/time-series events, this feature makes them match effectively for modeling the temporal attack patterns. It is possible for them to identify preceding dependencies and relationships between events occurring in time, thus contributing to a causation prediction of a future attack based on the domestic incident sequences.

Convolutional Neural Networks (CNNs): The CNNs excel in manipulating the structured data, like network traffic logs or the binary information. They have the possible power of learning by example and can self-generate what they need to from the data, which can be the good characteristics and/or patterns that may represent security vulnerabilities. For instance in cyber security CNN can be applied in identifying and confirming people's identities in pictures. Beside, segmenting an image into distinct sections that correspond to particular objects or features can help in identifying threats imposed by images.

The neural network for predictive threat modeling uses the learning process to optimize connective weights between each neuron with the purpose of minimizing the gap between the network's actual prediction and the desired prediction. For instance, video games do not have the ability to change events and circumstances, only the player following the pre-written game-script and set rules. Often, this is done by using optimization algorithms that are more adaptive like gradient descent and back propagation. These methods calculate the degree of errors through the gradient of error function with respect to weights and update them accordingly. Strategies like regularization, dropout, and transfer learning are not only theoretical but of immense use to neural networks in the prediction tasks involving threats [13]. This way, the models will be able to not only avoid bias towards the training set but also acquire knowledge from domains closely or pre-learned. Those models would definitely boost the models' capabilities to be more accurate and consistent.

5 Results

5.1 Learning Algorithms

An immersive capacity of neural networks is to train and realize the maximum performance by adjusting the "strength" of the connections between neurons, a process called learning. This learning technique depends on a set of algorithms that works inherently in a way that it is capable of adjusting the internal parameters of its modules in response to the data that it is exposed to. Essentially, learning algorithms in many neural networks are based on the Gradient Descent that underlies them. Gradient descent is a distributive (not independent) optimization algorithm that is employed in order to tackle the problem of minimization of an error (or loss) function which measures a discrepancy between the network's input and the desired output [15]. The objective of gradient descent is to set the weights of the links between neurons in a way that will minimize the mentioned/given error function. This is done by computing the gradient of the error function with respect to the weights of each neuron so that it shows how the error function is reduced which implies, what direction and magnitude of change is needed.

The weight adjustment process follows the update rule:

$$\Delta w = -\eta * \partial E / \partial w \quad (4)$$

where:

- Δw is the change in weight
- η (eta) is the learning rate, a hyperparameter that determines the step size of the weight updates
- $\partial E / \partial w$ is the partial derivative of the error function with respect to the weight, commonly known as the gradient

The negative sign within the update rule is vital because it helps correct the weights in a direction that is in opposition to the gradient. This way, it successfully minimizes the error function. Numerically, the neuronal network states are assigned an initial value of small random numbers at the beginning. This way, the process is iterative and with cycles the gradient descent algorithm trains the network by going through the error function and taking the gradients of the error with respect to each weight and computes the error with respect to these weights and updates them accordingly thus gradually refining the network representations and improving performance on the task. Gradient descent as such a general framework draws computational resources but is questionable in rates of efficiency for complex and deep neural networks that have many layers and connections. This one most important point boils down to the backpropagation, the specific kind of the gradient descent.

5.2 Backpropagation

Backpropagation is a powerful learning algorithm that is superlative for training complicated multi-layered neural networks in cybersecurity. It is a particular application of the gradient-descent that uses the Chain rule from calculus to efficient error signal propagation through the system so that the parameters throughout the network could be re-adjusted all at once [16].

The backpropagation algorithm can be summarized as follows:

Forward Pass: The reader's attention is drawn to the fact that previously gathered data becomes a part of the neural network as it includes the attack vectors, the neatly exploited vulnerabilities and the effect on the target systems. The central function of the network is to generate a quick warning or forecast about emerging cyber threat risks by triaging the information that comes from its tiers.

Error Computation: The measuring of the results that are stored in the database of past incidents and the predicted values by the neural network are compared. A loss function that would allow the finding the difference or error between the expected and actual results, such mean squared errors or cross-entropy, is utilized (i.e. the mean squared error for regression, cross-entropy for classification) [16].

Backward Pass: The calculus chain rule is then used to move the computed error backward through the layers of the neural network. This procedure calculates the contribution of each neuron and its weights to the total error (Fig. 4).

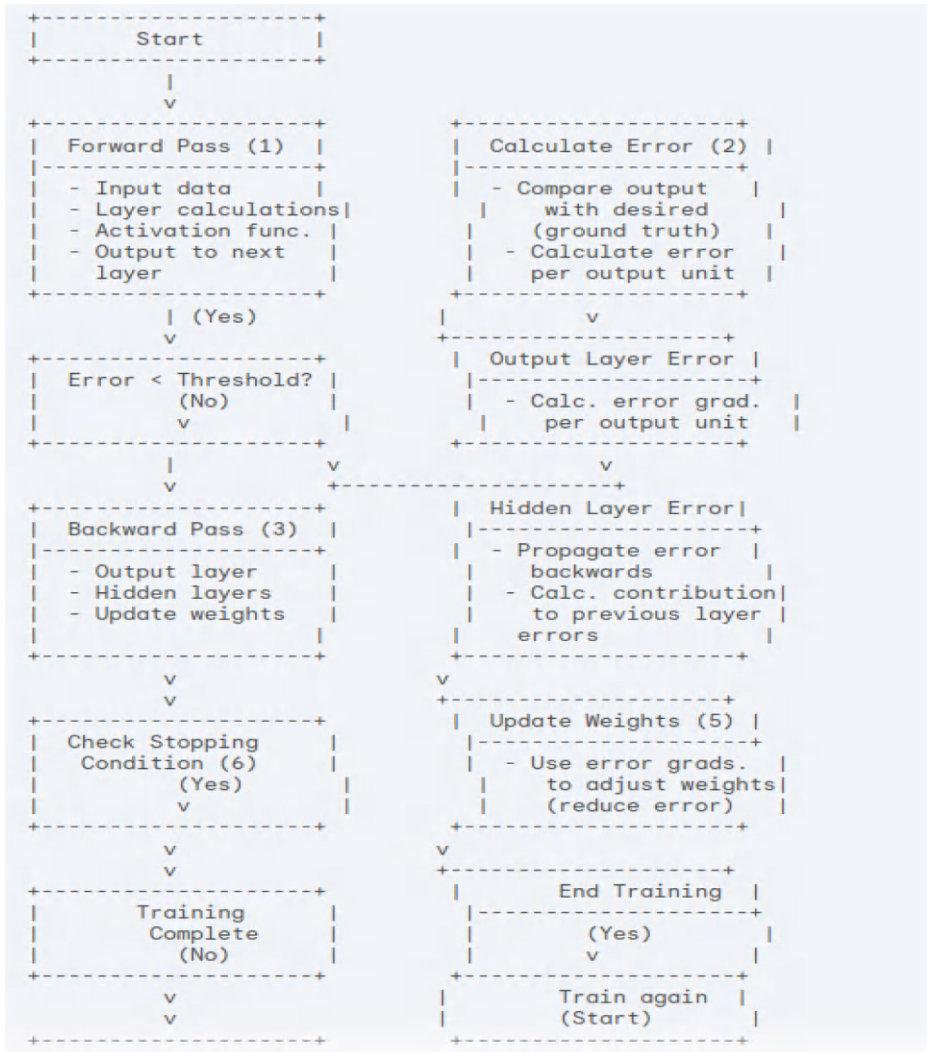


Fig. 4. Backpropagation Flowchart

As discussed and portrayed in Fig. 3., the forward pass operation is realized by input data flowing through the network in order to create the network output prediction, and the deviation from the prediction to the real values is measured. During the backward pass, the error gradients of the output layer are computed first among all of them, which is followed by a backward propagation of errors. Consequently, each node's contribution to the total error is determined. After that, the weight connections are updated with the errors between real values and the network's prediction to reduce the total error. This process passes through with additional input data until it eventually meets a halting situation, for instance, the accuracy that is required or ending the training that has reached

the set number of training iterations over the entire dataset. With the stopping condition saved, the network training thus will come to an end as an epoch or iteration. The faster pace gives it the leverage to repeatedly polish the internal representations and achieve a better percentage of sleep with its ability to predict upcoming attacks.

Backpropagation is a potent tool that has been a playing boss in the triumph of deep learning as well as big, complex neural networks that are trained on it. Gradient computation happens in an efficient manner running throughout all layers, thus the neural network can have access to hierarchical representations and extract complex features from the raw data. To address issues like vanishing or exploding gradients and further enhance the training process, researchers have recently developed and investigated a variety of backpropagation extensions and modifications, including gradient clipping strategies, momentum techniques, and adaptive learning rate methods (e.g., RMSProp, Adam).

5.3 Training Neural Networks in Cybersecurity

The neural network is a strong machine learning model that handles complicated tasks and the analysis of the data. But, a neural network that can perform well has to go through the phase of training, in which it will learn from examples. The character and nature of the training data hold a vital place in building the neural network, which can lead to its learning effectiveness.

There are three main types of training data used in neural networks:

Supervised learning: In this instance, the training data is made up of labeled instances, where each input (such as characteristics that describe a cybersecurity incident) is linked to a specific output (such as the nature of the attack or its result). The neural network gains the ability to translate input data to matching output labels during training. When previously collected incident data with clearly defined labels or classes is available, this method works well.

Unsupervised learning: Compared to supervised training, unsupervised learning involves teaching the network models on unlabeled data. The network's purpose is to pick out the relationships or architecture in the data without being explicitly directed as to what to look for. Clustering, the goal of which is to group objects based on similarities and differences, dimensionality reduction, and anomaly detection are among the main tasks of unsupervised learning [17].

Reinforcement Learning: The cycle is converted from theory to practice sequentially. A network is the typical one referred to as a player, which interacts with the environment, and receives rewards or punishments for her actions. The aim is to focus on the determination of a policy (a function of states that identifies actions) that provides maximum returns over the total period of time. Encouragement Learning is propitious in the aforementioned fields, comprising game playing, robotics, and control, to mention but a few [17].

Whether the data is the images of handwritten digits mentioned above or any other type of data, the neural network training process always follows a similar pattern. Here are the key steps involved:

Data Pre-processing: Previously, the data required preprocessing in order to be in the form for the neural net, now it can be simply fed. This operation could involve methods like normalization, scaling or encoding for dealing with categorical data threat variables.

Model selection: It is very important to select a genuine neural network for the given problem. Such designing consists in setting the number and type of layers, the number of neurons in each of the layers, as well as providing the kind of layers (e.g. convolutional, recurrent).

Hyperparameter Tuning: Hyperparameters like the learning rate, batch size, and number of epochs required to get much better results form a neural network. The sampling strategy basically is implemented through methods such as grid search and random search.

Forward Pass: In the training phase, historical threat data is introduced into the network in the same way as the network of neurons, and the signals propagate from neuron to neuron. With each layer of neural networks that does the computations for the inputs and receives the result from the next layer and culminates with the output.

Error calculation: The difference between the output of a neural network and the desired output (real value) in the loss function is compared using the loss function. Usual uncomplicated loss functions include mean square error for regressive issues, and cross-entropy for classification issues.

Backward Pass (Backpropagation): The error signal is conveyed back simultaneously inside of a deep network, and this gives the ability to ultimately calculate gradients for each weight and bias parameter. It is also performed by the method known as backpropagation (refer to Fig. 3).

Updating weights: Rather than the above discussed optimized neural network after backpropagation, parameters pertaining to the network are updated with the gradients computed during backpropagation using optimization algorithms such as stochastic gradient descent or one of its variants. g., Adam, RMSProp).

Epoch Completion: Likewise, learning one epoch follows what was done when the entirety of training examples had been processed. The updating of weights and biases usually goes along several epochs, with such updates being made after each epoch.

Training Loop: This process is usually performed for multiple epochs, and it ends when some stopping criteria are reached, for example, after a determined maximum number of epochs is achieved or a minimum level of performance on the validation set is not reached. The learning process is observed by measuring performance indices, which can be represented by accuracy, loss function, or any other relevant task-specific performance measures. This is usually done for the for the evaluation of such metrics on a validation dataset that is different from the training data to make sure the model makes a good generalization and bypasses overfitting. The neural network trained on the dataset can be deployed for a fresh inference on new, unseen data once you have completed the training. Nevertheless, such accuracy might be misleading in terms of its real-world performance, and thus it is worthwhile to apply a test set to obtain the true performance of the model in an unbiased manner.

6 Discussion

Throughout the assessment of the performance of artificial neural networks for anticipating threat modeling in cybersecurity, we discovered numerous experiments that have been performed on vast datasets of historic cybersecurity cases. These datasets usually consist of thousands, of threat occurrences covering various areas such as malware infections, phishing attacks, network breaches, and information violations. Each event is represented by a collection of functions consisting of the assault kind, susceptibilities manipulated, target systems, and the general effect on the company. Several neural network styles have been investigated, consisting of feedforward neural networks, recurrent networks (RNNs), and convolutional neural networks (CNNs). Typically, supervised learning is used to train the models, which are then used to forecast future cyberattack likelihood and probable consequences using patterns identified from past events. The efficiency of these experienced versions is assessed utilizing held-out test set collections, determining the precision in forecasting the kind of threat and the mistake in approximating the prospective effect or seriousness of anticipated assaults. Usual assessment metrics consist of mean squared error (MSE), root mean squared error (RMSE), and mean absolute error (MAE). Neural networks, especially trained CNNs and RNNs, possess a higher chance of anticipating threats in cybersecurity. CNNs have revealed high precision, commonly going beyond 89%, in forecasting the kind of cyber threat, with MSE values for impact estimation as low as 0.102 [18]. Their capacity to instantly discover the essence of pertinent attributes from structured information resources such as network, web, traffic logs, and system occasion logs adds to their solid efficiency. RNNs have also performed well, particularly those with long short-term memory (LSTM) units, reaching MSE ranging from 0.115 to 0.130 and attack-type prediction accuracies of between 87–88%. The temporal relationships and sequences found in cyberattack scenarios are especially well-modeled by RNNs, which probably improves their predictive power. While feedforward neural networks have shown sensible efficiency, with attack type accuracies around 84% as well as RMSE around 0.197, they have typically been overtaken by the extra intricate CNN together with RNN designs, which can much better catch the complex patterns coupled with partnerships within the cybersecurity occurrence information. Hybrid designs such as autoencoders paired with feedforward networks, as well as ensemble models combining CNNs and RNNs, also perform well. These methods have yielded encouraging results: ensemble models have achieved as low as 0.084 MAE) and as high as 91.5% attack-type prediction accuracy.

Likewise, the reliability of the training data also plays a critical role in determining the efficacy of these models. The key factor that turns threat predictions into a practical reality is the identification of previous case data that is thorough, timely, and free from bias. In addition, as cyber threats continue to evolve, it is necessary to frequently retrain and upgrade the neural network designs with brand-new case information to record arising strike patterns and susceptibilities. Furthermore, the interpretability, as well as the interpretability of the neural network forecasts, remains an obstacle, as these intricate versions usually run as “black boxes,” making it tough to recognize the thinking behind their forecasts. Continuous study initiatives are concentrated on establishing strategies to boost the interpretability of neural networks, which can additionally improve their functional functionality in cybersecurity and various other domains. Generally, these findings

show the considerable capacity of neural networks, particularly CNNs and RNNs, for anticipating threats in cybersecurity. By anticipating the kind and prospective influence of future cyber assaults, companies can proactively apply ideal mitigation techniques, improving their general cybersecurity position and durability versus advancing cyber dangers.

7 Conclusion

Neural networks, being the prime technological shift, are the groundbreaking technology in cybersecurity. Through utilizing the principles of the human brain architecture and how they operate, computational models have proven over time their greatness in learning intricate patterns, recognizing complex details, and making cognitive decisions for a broad spectrum of their applications. All through this thorough analysis related to neural networks, we have touched upon the key areas that sum up the neural networks biologically driven inspirations and offensive bulldozer blocks that are the fundamental building blocks of the neural network and the numerous architectures and learning algorithms that make it possible for neural networks to achieve their performances. We not only have seen the neural networks tremendous capabilities, but also witnessed them performing not only comprehensive tasks like image recognition, natural language processing, speech recognition, and predictive analytics, among other things. Nevertheless, besides the good sides, there are the limitations that also couldn't be neglected. The quality and correctness of data, the requirement of data to be comprehensive, the challenge of interpretation, the bias and fairness concern and the computational requirements remain the key issues researchers continue to face. As a result they are moving towards models that are increasingly more wise and efficient. Furthermore, there are a lot of emerging forex predictions in neural networks that are in the offing and will be confronted with tough choices. Progress on topics including unsupervised learning, few-shot learning as well as hybrid models which are a blend of neural networks, rule-based symbolic reasoning and knowledge graphs include the dimension of the capabilities and settings where these powerful systems will be widely utilized. Another way through is that neural networks applied in the social sector become more ubiquitous, they should be emphasized and taken care of by ethics, and responsible deployment and progress will be given priority. Coordination between the scientists, policy makers, and stakeholders will be the epicenter of correcting possible social repercussions of such an advanced technology. Neural networks are the most prominent artificial intelligence algorithm because of their capacity to learn, adapt, and discover hidden patterns. The neural networks are about to shake several industries and solve some of the most pressing issues we have in the world. They are the leaders in opening up new frontier of understanding of how the world works. The heights of the boundaries of what can be done are yet to be reached as we are pushing harder and harder; however, the rather sensational path of neural network development is ahead of us.

References

1. Kanade, V.K.: Artificial Intelligence. Google.com (2024). https://www.google.com/url?sa=t&source=web&rct=j&opi=89978449&url=https://www.spiceworks.com/tech/artificial-intelligence/articles/what-is-ai/&ved=2ahUKEwjM16DJ5JKGAXX8W0EAHcpDDUUFnoECBgQAw&usg=AOvVaw324c4wq44g_Wz1xp6C0QIV. Accessed 16 May 2024
2. Shrestha, Y.R., Krishna, V., von Krogh, G.: Augmenting organizational decision-making with deep learning algorithms: principles, promises, and challenges. *J. Bus. Res.* **123**, 588–603 (2021). <https://doi.org/10.1016/j.jbusres.2020.09.068>
3. Neural Network Software for Predictive Modeling and Deep Machine Learning.: www.roselladb.com. <https://www.roselladb.com/neural-network.htm>. Accessed 16 May 2024
4. Puderbaugh, M., Emmady, P.D.: Neuroplasticity. *Nat. Libr. Med.* 01 May 2023. <https://www.ncbi.nlm.nih.gov/books/NBK557811/>
5. Qi, X., Chen, G., Li, Y., Cheng, X., Li, C.: Applying neural-network-based machine learning to additive manufacturing: current applications, challenges, and future perspectives. *Engineering* **5**(4), 721–729 (2019). <https://doi.org/10.1016/j.eng.2019.04.012>
6. Mohana, J.M., Madhulika, M.S., Divya, G.D., Meghana, R.K., Apoorva, S.: Feature extraction using convolution neural networks (CNN) and deep learning. *IEEE Xplore* (2018). <https://ieeexplore.ieee.org/document/9012507>
7. Yacim, J.A., Boshoff, D.G.B.: Impact of artificial neural networks training algorithms on accurate prediction of property values. *J. Real Estate Res.* **40**(3), 375–418 (2018). <https://doi.org/10.1080/10835547.2018.12091505>
8. Bebesko, B., Khorolska, K., Kotenko, N., Kharchenko, O., Zhyrova, T.: Use of neural networks for predicting cyberattacks. <https://ceur-ws.org/Vol-2923/paper23.pdf>
9. Ganesh, S.: What's the role of weights and bias in a neural network? Medium 30 July 2020. <https://towardsdatascience.com/whats-the-role-of-weights-and-bias-in-a-neural-network-4cf7e9888a0f>
10. Bharadiya, J.: Machine learning in cybersecurity: techniques and challenges. *Eur. J. Technol.* **7**(2), 1–14 (2023). <https://doi.org/10.47672/ejt.1486>
11. Apruzzese, G., et al.: The role of machine learning in cybersecurity. *Digital Threats Res. Pract.* (2022). <https://doi.org/10.1145/3545574>
12. DeepAI: Feed forward neural network. DeepAI 17 May 2019. <https://deepai.org/machine-learning-glossary-and-terms/feed-forward-neural-network>
13. Salehin, I., Kang, D.-K.: A review on dropout regularization approaches for deep neural networks within the scholarly domain. *Electronics* **12**(14), 3106 (2023). <https://doi.org/10.3390/electronics12143106>
14. Learnopencv: Understanding feedforward neural networks. LearnOpenCV, Learnopencv 09 Oct 2017. <https://learnopencv.com/understanding-feedforward-neural-networks/>
15. IBM: What is gradient descent? IBM. www.ibm.com. <https://www.ibm.com/topics/gradient-descent>
16. Kostadinov, S.: Understanding backpropagation algorithm. Medium 08 Aug 2019. <https://towardsdatascience.com/understanding-backpropagation-algorithm-7bb3aa2f95fd>
17. IBM: Supervised vs. unsupervised learning: what's the difference? | IBM," www.ibm.com, 13 May 2024. <https://www.ibm.com/think/topics/supervised-vs-unsupervised-learning>
18. Kattenborn, T., Leitloff, J., Schiefer, F., Hinz, S.: Review on convolutional neural networks (CNN) in vegetation remote sensing. *ISPRS J. Photogramm. Remote. Sens.* **173**, 24–49 (2021). <https://doi.org/10.1016/j.isprsjprs.2020.12.010>



Weighted Extreme Gradient Boosting Based Cybersecurity Risk Assessment in Investment Banking and Financial Sector

Pradeep Chintale¹ , Hrushikesh Deshmukh² , Anirudh Khanna³ ,
Ankur Mahida⁴ , and Madhavi Najana⁵ 

¹ SEI Investment Company, Downingtown, PA 19335, USA
chintale.pradeep@gmail.com

² Senior Consultant, Fannie Mae, Reston, VA, USA

³ Data Protection Lead, 900 Acadia Dr, Plano, TX 75023, USA

⁴ SRE, Barclays PLC, Whippany, NJ, USA

⁵ Software Engineering Manager, Columbus, OH, USA

Abstract. Banking fraud prevention as well as risk management are dominant in an advanced financial landscape and the integration of Artificial Intelligence (AI) provides the promising path for advancements in these areas. However, because of the nature of banking sectors inherent, data security cannot be threatened. Hence, this research proposes the Weighted Extreme Gradient Descent Boosting (WXGBoost) approach for the prediction of the attacks in banking and financial investments in cybersecurity. Initially, the data is collected from the various banks of transaction details. Then, the min-max normalization technique is utilized to normalize the collected of the transaction details from the banks. To assure the security of the financial sector management, the Advanced Encryption Standard (AES) algorithm is used for both encryption and decryption of the data. The proposed WXGBoost approach enhances the effectiveness of the cyber security through enhancing their protection over cyberattacks. The experimental results shows that the proposed WXGBoost approach attains the accuracy of 0.894, precision of 0.885, recall of 0.893 and F1-score of 0.901 respectively. These obtained values from the proposed method attains better results when compared to the existing methods K-Nearest Neighbor (KNN) and Reinforcement Learning (RL).

Keywords: Advanced Encryption Standard · Artificial Intelligence · Banking and financial investments · Transaction details and Weighted Extreme Gradient Descent Boosting

1 Introduction

In this advanced world, cybersecurity has become gradually significant because of the fastest expansion of the digital areas and enhancing the cyberthreats [1]. Its importance lies in securing the cyberthreats against preventing financial losses, preventing individual privacy, securing businesses as well as managing the public security [2]. The

cyber-attacks have developed in frequency and brutality, creating a serious threat to the infrastructure society, economy as well as personal data [3]. The banking as well as financial sectors is the serious infrastructure which plays an important role in a global economy. This sector has experienced an important revolution in a decade, with the technological advancements resulting to the digitization of the banking services [4]. The cybersecurity is the most important challenges facing the various security threats in banking industry. The cyber threats in a banking sector have crucial effect which involves financial loss, reputation damage as well as legal penalties [5]. The cybersecurity attacks are classified into various types such as physical, synthetic and semantic attacks. Hence, it is important to comprehend the character and nature of those threats and the develop the efficient remedy to alleviate their influence [6, 7].

With the backdrop of escalating the security apprehensions, the combination of Artificial Intelligence (AI), Blockchain and Business Intelligence (BI) arises as a transformative strategy poised to develop the banking security [8]. The integration of these techniques depicts the significant shift from reactive to proactive security measures. The advanced AI techniques with the Machine Learning (ML) have been effectively performed in various applications [9, 10]. The AI techniques authorizes the banking systems to detect and respond to threats in real-time. By continuously analyzing the large number of data, the AI approaches have the capability to determine the patterns indicative of suspicious activities [11, 12]. This approach allowing the banks to proactively prevent the potential openings before they escalate into important security events [13]. The Deep Learning (DL) is a subcategory of AI, which becoming an increasingly famous tool and effective defense strategies against the harmful attacks [14]. The DL approaches have the capability to compute machine application which has recognize complex, non-linear patterns and after use those patterns to make predictions [15]. However, because of the nature of banking sectors inherent, data security cannot be threatened. Hence, this research proposes the Hence, this research proposes the Weighted Extreme Gradient Descent Boosting (WXGBoost) approach for the prediction of the attacks in banking and financial investments in cybersecurity.

The primary contributions of this research are listed as follows:

- This research proposes the WXGBoost approach for the prediction of the cyber-attacks in banking and financial sectors.
- To assure the security of the financial sector management, the Advanced Encryption Standard (AES) algorithm is used for both encryption and decryption of the data.

The remainder of this research is listed in following manner: Sect. 2 presents the related works of this research. Section 3 described the proposed methodology. Section 4 presents the results and analysis and Sect. 5 concludes the overall research.

2 Literature Review

In this section, the existing related to the cybersecurity risk assessment in investment banking sectors are discussed along with their advantages and limitations.

Shailendra Mishra [16] presented the cybersecurity approach according to AI for the Financial Sector Management (CS-FSM). An Enhanced Encryption Standard (EES) approach was utilized to ensure the financial sector data. The Machine Learning (ML) approach of K-Nearest Neighbor (KNN) was utilized for the prediction through learned from the trained data. Furthermore, the presented approach was utilized for the detection of malware attacks in a financial sector. AI technique provided a more accurate risk assessments through the estimation of broad range of potential attack vectors. However, the CS-FSM approach had unintentionally produced errors, affecting the fairness and equity of security measures.

Zeinab Shahbazi and Yung-Cheol Byun [17] developed the Hierarchical Risk Parity (HRP) as well as Unsupervised ML approach for the cryptocurrency structure. An Identification of the cryptocurrency risks with a much probability of occurrence and an unauthorized access to private keys involved sympathetic various key factors. The HRP provided the efficient result with respect to returned the modified risk tail to obtain the better risk management. However, the developed HRP required the effective understanding of both statistical methods as well as financial theory, making it challenging for individuals or firms without specialized expertise.

Abdulbasit A. Darem et al. [18] introduced the comprehensive analysis of the cybersecurity in both financial and banking areas, which involves determination of basic attacks, their nature as well as character to support in classification. The introduced classification approach supported to determine the suitable countermeasures needed to alleviate the risks of every attack types. The technical, non-technical, legal as well as regulatory evaluations were utilized to guarantee the financial transactions from cyber-attacks. The cyber-attacks were classified to banking as well as financial sectors based on their technicality and severity. However, the cyber threats were frequently interconnected and occurred simultaneously or in fast succession.

Ala Mughaid et al. [19] developed the detection model based on ML approaches through the partition of dataset. The three different publicly available resource datasets with the features were utilized for the detection and filtering of the phishing email attacks. The collected datasets were pre-processed through the prevention of the duplicate rows, eliminate missing values as well as balancing the instances to attain most accurate rate. The seven most important unsupervised ML approaches were utilized to train and test the model for phishing email detection with the grouped features. However, the developed ML approaches had obtained the minimum accuracy values and minimum detection rate.

Ana Rita D. Rodrigues et al. [20] developed the realistic decision-support approach through the integration of cognitive mapping as well as Decision-making Trial and Evaluation Laboratory (DMARTEL) approach with respect to solve the cybersecurity problems. The group sessions with an expert board were performed to substitute the methodological processes, hence permitted realism to be associated into the final outcomes. The developed approach analyzed the complex decision problems as well as effectively visualized the causal relationships among various clusters. However, the

dynamic and evolving nature of threats were making it difficult for experts to provide accurate and consistent evaluations.

From this review, some of the limitations have been identified: unintentionally produced errors, required the effective understanding of both statistical methods as well as financial theory, cyber threats were frequently interconnected, less accuracy values and dynamic and evolving nature of threats were making it difficult. To solve these limitations in existing works, this research aims to developing a WXGBoost approach for the prediction of the cyber-attacks.

3 Proposed Methodology

In this section, the WXGBoost approach is proposed for the analyzing the cybersecurity risks in banking and financial sectors. Figure 1 depicts the overall architecture of the proposed method. The cybersecurity in Financial Sector Management with the support of AI examines the instructions and enables the individuals to identify the user is safe or not. The financial data of the customers, information about the banking sectors and other information are stored in the database. The firewall blocks are accessed only when the attackers are entered to accessing the data. The data are encrypted by the utilization of private or public key. This research designs the model based on the data provided through the authorized persons. After the design of the achievable prediction approach, the accessing the data is effectively verified. Unfortunately, an unknown entry attempts to utilize the data, the system will stop their entry and alert to a legal person.

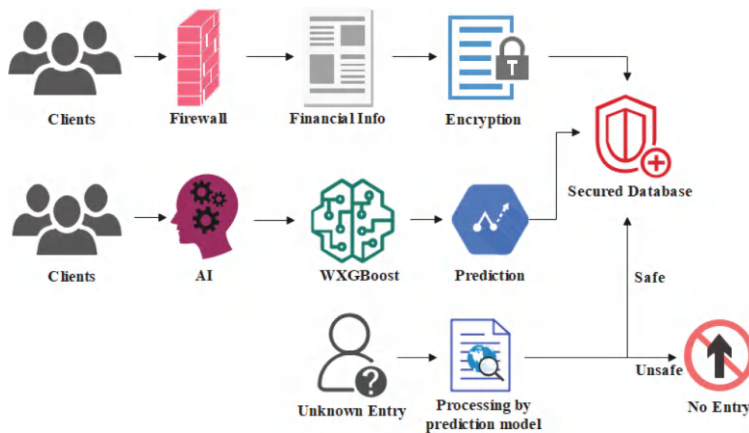


Fig. 1. Architecture of the proposed WXGBoost Algorithm

3.1 Data Collection

In the banking sector, data considered for this research involves transaction details like sender, amount, and receiver, performance history, user authentication and access logs.

In this research, the user information like username, account number, IFSC code, number of balance amount, transaction details, payment method, bank name and so on. These details have been collected from 500 number of users. These collected data are fed into a pre-processing phase which increases the effectiveness of the following process.

3.2 Pre-processing

The min-max normalization [21] is a technique, which is used to scale the input values in the range between 0 and 1 respectively. The estimation of the min-max normalization is mathematically expressed in Eq. (1) as:

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

where, X' – normalization data; X – actual data; X_{min} . And X_{max} – minimum and maximum data. This normalization is smallest sensitive to the outliers and it supports to improve the accuracy results. After pre-processing, normalized dataset is partitioned into two parts: 80% of data are used for the training and 20% for testing.

3.3 Encryption and Decryption

AES is used to encrypt sensitive data at rest and in transit. This includes user data, transaction details, proprietary trading algorithms, and communication between branches or with clients. In communications, AES ensures that data remains encrypted from the sender to the recipient, preventing interception and unauthorized access during transmission. The AES [22] is the most popular encryption approach for the protection of sensitive data. The AES contains the key sizes of 128 bits and block sizes, the key lengths be 128, 192 and 256 bits. For an encryption task, the round transformation is employed as the group of iterations, which involves sub-byte, shift-row, mix-column as well as add-round key tasks. The AES utilizes the fixed S-box which has been impacted through the cryptanalysis attacks. This research to utilize a Henon map system and splits the key into 2 sub-keys which are employed as parallel. There is no relationship among the key and sub-keys, the cryptosystem will attack to cryptanalysis attacks. Simultaneously, an execution time is minimized since the data is implemented as parallel, and prevented the necessities for various rounds in encryption as well as decryption. The use of OR exclusive enhances the speediness as well as combining rows and columns amends the diffusion, which makes the encryption process safer. The Henon map is formulated in Eq. (2) and (3) as follows:

$$X_{(n+1)} = 1 - aX_n^2 + Y_n \quad (2)$$

$$Y_{(n+1)} = bX_n \quad (3)$$

where, a . and b depicts the canonical values; n denotes the bit numbers; X and Y represents the input bits. To acquire the significant chaotic replies, it is considered based on the search on Henon map, which a and b values are 1.4 and 0.3 correspondingly

as represented in the strange attractor. The enhanced encryption as well as decryption expression is formulated in Eq. (4) as follows:

$$Cipherdata = (((Originaldata \oplus K'x)Mr)Mc) \oplus K'y \quad (4)$$

$$Originaldata = ((Cipherdata \oplus K'y)Mc)Mr \oplus K'x \quad (5)$$

where, Mr represents the mix-row transformation as well as Mc represents the mix-column transformation operation.

3.4 Prediction using Weighted Extreme Gradient Boosting

XGBoost is a popular ML approach, which is a modification of the GDBT approach, and the combination of strategy of gradient boosting as well as DT approach. The number of DT are integrated into the gradient boosting model for iterative optimization of the that target training data. The conventional XGBoost implements the banking and financial prediction through minimizing the expected risk is formulated in Eq. (6) as follows:

$$R(i|x) = \sum_{j=1} C_{i,j} P(j|x) \quad (6)$$

where, $P(j|x)$ represents the probability estimation from an experimental training set $X = \{(x_n, y_n)\}_{n=1}^N$; $C_{i,j}$ denotes a cost of predicting k into class j . Based on Eq. (6), the prediction is identified through an expected loss function, which is formulated in Eq. (7) as follows:

$$i^* = \arg \min_i R(i|x) = \arg \min_i E_{X,Y} [c_{i,j}] \quad (7)$$

where, X and Y depicts an input and output space of XGBoost-based financial prediction approach. The $P(j|x)$ represents the posterior probability which is determined from an experimental training data, an experimental risk is formulated in Eq. (8) as follows:

$$R'_L = E_{X,Y} [l] = \frac{1}{N} \sum_{n=1}^N l(p_n, y_n, c) \quad (8)$$

where, p_n denotes a predicted probability of classifying n th business loss data. $l()$ represents the loss function which optimizes the prediction bias of the banking and financial tasks. In an actual operation of XGBoost, the cost-intensive loss function, the cross-entropy is developed, which is expressed in Eq. (9) as follows:

$$l_c(p, y) = -y \log p \quad (9)$$

Integrating the Eq. (3) and (4), it is a cost-insensitive XGBoost operates the financial prediction based on $c_{1,0} = c_{0,1} = 1$, where, $c_{1,0}$ denotes the misclassifying cost of Shrinkage and Tree Boosting (ST) firm into Node Structure Tree (NST) firm, whereas, $c_{0,1}$ denotes a cost of misclassification of the NST into ST firm. The transformation

of the cost-insensitive XGBoost into cost-sensitive version is formulated in Eq. (10) as follows:

$$l_{wc}(p, y, c) = w_{1,0}y\log p + w_{0,1}(1 - y)\log(1 - p) \quad (10)$$

where, $w_{1,0}$ represents the misclassified ST samples weights and $w_{0,1}$ denotes the misclassified NST sample weights. The WXGBoost is the cost-sensitive form of the XGBoost which holds a pattern of the XGBoost. It gives the opportunity to preserve a feature significance score to explore a global interpretability of the WXGBoost. Hence, in the boosted tree which involves number of DT, the significance score of the j th feature is estimated through averaging the feature significance against M trees is formulated in Eq. (11) as follows:

$$I_j^2 = \frac{1}{M} \sum_{m=1}^M I_j^2(T_m) \quad (11)$$

where, I_j^2 – represents the significance of j th feature based on WXGBoost. $I_j^2(T_m)$ measures the significance score of j th feature on m th DT T_m , which is curtailed formulated in Eq. (12) as follows:

$$I_j^2(T_m) = \sum_{k=1}^K \Delta_k^2 l(v(k) = j) \quad (12)$$

where, K denotes the number of leaf nodes on m . th DT; $l(v(k) = j)$ judges where the selected splitting feature on k th split nodes is j th feature; Δ_k^2 denotes the impurity reduction after node splitting with k th splitting feature.

3.5 Data Privacy and Risk Reduction Ratio

Controlling the data security as well as protection of the user privacy data is significant. Obtaining the monitoring requirements, a secure management of the complex data like personally identifiable data and other data is known as data privacy. A risk reduction ratio is the product of an accuracy percentage of the total measurement, an individual action's project overheads as well as minimization in the probability of the risk. The data reduction is the enhancement scheme which is substituted to remove the risk from the tasks. This is a significant control approach due to it eliminates a danger risks, less threat as well as lack of probability harm.

To secure the data, the user needs to execute different design approaches and practices. To protect the sensitive data, the bank and financial sectors must often have a strategy for the data management. Trades should protect the most significant data, which involves personnel files, client data, constancy programs, transactions as well as collection of data. This is to protect over the third parties like phishing and other fraudulent activity. The personal data of the customers, personal records as well as transactional data are the sensitive data, which is must be protected from unauthorized access.

The AI approach of WXGBoost is utilized in this research to offer the cyber security strategy for banking and financial sectors, which supports to minimize a risk of the

prohibited access to trade system or the network through an implementation of the essential networking safeguards. It will support to enhance the stakeholder contribution in the data protection process.

4 Experimental Results

An efficacy of the proposed WXGBoost is trained, tested and validated in this section. The proposed WXGBoost is experimented on the environment of Python 3.8, and the system specifications with Windows 10, 64-bit OS, Intel Core i7- 4200U CPU processor, 16GB RAM. The proposed method is assessed through utilizing different assessment metrics like accuracy, precision, sensitivity and F1-score. The mathematical expressions of each metric are formulated in Eqs. 13, 14, 15, and 16 as follows:

Accuracy = (TP + TN) / (TP + TN + FP + FN) (13)

Precision = TP / (TP + FP) (14)

Sensitivity = TP / (TP + FN) (15)

F1 - score = (2TP) / (2TP + FP + FN) (16)

where, TP - True Positive; TN – True Negative; FP – False positive; FN – False Negative.

4.1 Performance Analysis

In this section, the qualitative and quantitative analysis of the proposed WXGBoost is estimated by using the collected dataset. The effectiveness of the proposed method is validated by using the various performance metrics. Table 1 represents the performance analysis of the proposed WXGBoost algorithm. Figure 2 depicts the graphical representation of analysis of the risk reduction ratio. Figure 3 represents the graphical representation of the data privacy ratio.

Table 1. Performance analysis of WXGBoost

Methods	Accuracy	Precision	Recall	F1-score
CatBoost	0.831	0.815	0.823	0.837
AdaBoost	0.842	0.832	0.846	0.852
GBoost	0.863	0.857	0.864	0.886
XGBoost	0.872	0.864	0.874	0.897
WXGBoost	0.894	0.885	0.893	0.901

Table 1 represents the performance analysis of the proposed method using various performance metrics. The existing works such as CatBoost, AdaBoost, GBoost and XGBoost algorithm are compared with the proposed WXGBoost algorithm. The proposed WXGBoost has the capability to assign the weights in the model, provides greater customization as well as flexibility in the model training. The proposed WXGBoost approach attains the accuracy of 0.894, precision of 0.885, recall of 0.893 and F1-score of 0.901 respectively.

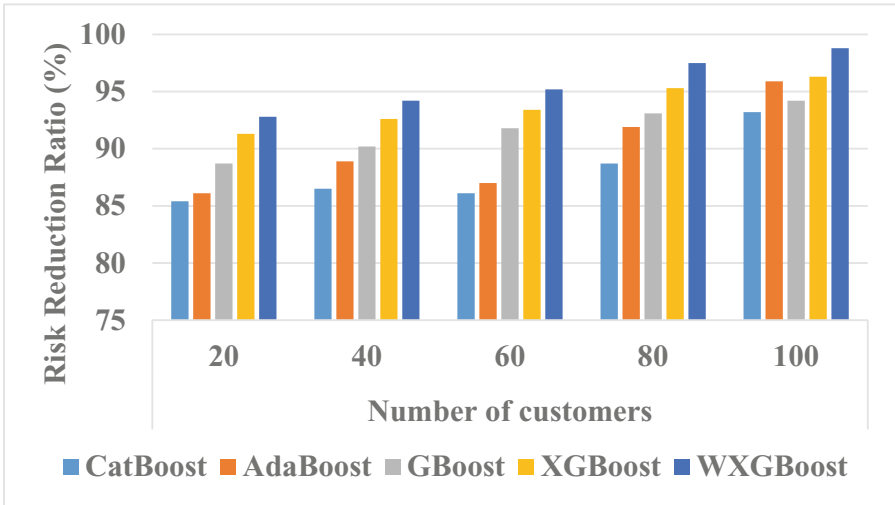


Fig. 2. Graphical representation of risk reduction ratio

In Fig. 2, the graphical representation of the proposed WXGBoost approach is discussed along with the number of customers. The existing methods such as CatBoost, AdaBoost, GBoost and XGBoost algorithm are compared with the proposed WXGBoost algorithm. The proposed WXGBoost approach attains the risk reduction ratio of 92.8%, 94.2%, 95.2%, 97.5% and 98.8% at the number of customers of 20, 40, 60, 80 and 100 respectively.

In Fig. 3, the graphical representation of the analysis of the data privacy ratio with the number of customers. The existing methods such as CatBoost, AdaBoost, GBoost and XGBoost algorithm are compared with the proposed WXGBoost algorithm. The proposed WXGBoost approach attains the data privacy ratio of 91.8%, 93.2%, 94.2%, 96.5% and 98.5% at the number of customers of 20, 40, 60, 80 and 100 respectively.

4.2 Comparative Analysis

The comparison of proposed WXGBoost method is represented using various performance metrics. Table 2 represents the comparative results of the proposed method with existing approaches named KNN [16], Random Forest (RF) [17], Decision Forest [19] and Decision Support [20] based on the collected data. The existing method's values

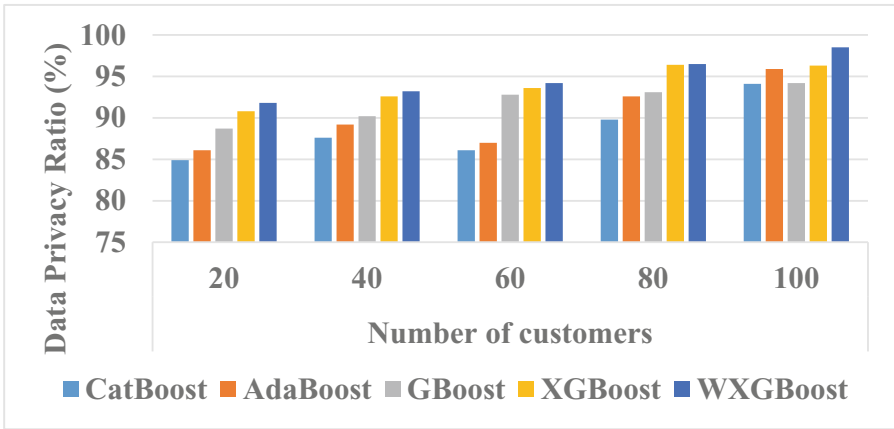


Fig. 3. Graphical representation of the data privacy ratio

are simulated according to the proposed method values. The proposed method’s performance is evaluated by utilizing evaluation metrics named accuracy, precision, recall and F1-score.

Table 2. Comparative Analysis of the proposed method

Methods	Accuracy	Precision	Recall	F1-score
KNN [16]	0.831	0.813	0.845	0.23
RL [17]	0.834	0.842	0.852	0.831
Decision Forest [19]	0.869	0.874	0.861	0.868
Decision Support [20]	0.885	0.873	0.852	0.863
Proposed WXGBoost	0.894	0.885	0.893	0.901

4.3 Discussion

In this section, the achievement of the proposed CHLF-CNN approach is discussed along with their advantage. The existing works have the limitation such as unintentionally produced errors, required the effective understanding of both statistical methods as well as financial theory, cyber threats were frequently interconnected, less accuracy values and dynamic and evolving nature of threats were making it difficult. To solve these limitations in existing works, this research aims to developing a WXGBoost approach for the prediction of the cyber-attacks. Accurate identification of high-risk events using WXGBoost approach allows banks to allocate resources more efficiently. Through prioritizing these high-risk events by weighted approach, banks can enhance their risk management strategies and respond more effectively to potential threats. The AES algorithm is used

for both encryption and decryption of the data to assure the security of the financial sector management. The proposed WXGBoost approach attains the accuracy of 0.894, precision of 0.885, recall of 0.893 and F1-score of 0.901 respectively.

5 Conclusion

The banking and financial sector faces a progressively complex landscape of cyber threats. This research proposes the WXGBoost approach for the prediction of the cyber-attacks in the banking as well as financial sectors. The proposed WXGBoost has the capability to assign the weights in the model, provides greater customization as well as flexibility in the model training. The AES algorithm is used for both encryption and decryption of the data to assure the security of the financial sector management. The AES algorithm supports to preventing unauthorized access and ensures that only legitimate users to access sensitive information. This approach is the key for the protection of sensitive data, preserving customer trust as well as justifying the financial losses activated through cyber-attacks. The proposed method has contributed an exhaustive examination of the cyber threats directing the banking as well as financial sector. The proposed WXGBoost approach attains the accuracy of 0.894, precision of 0.885, recall of 0.893 and F1-score of 0.901 respectively. In future, the proposed will expand to utilize the DL approach for the enhancing the model performance.

References

1. Manoharan, A., Sarker, M.: Revolutionizing cybersecurity: unleashing the power of artificial intelligence and machine learning for next-generation threat detection (2023). <https://doi.org/10.56726/IRJMET32644>
2. Ivanyuk, V.: Forecasting of digital financial crimes in Russia based on machine learning methods. *J. Comput. Virol. Hack. Tech.* **20**(3), 1–14 (2023). <https://doi.org/10.1007/s11416-023-00480-3>
3. Alshammari, F.H.: Design of capability maturity model integration with cybersecurity risk severity complex prediction using bayesian-based machine learning models. *SOCA* **17**(1), 59–72 (2023)
4. Biswas, B., Mukhopadhyay, A., Bhattacharjee, S., Kumar, A., Delen, D.: A text-mining based cyber-risk assessment and mitigation framework for critical analysis of online hacker forums. *Decis. Support Syst.* **152**, 113651 (2022)
5. Biswas, B., Mukhopadhyay, A., Kumar, A., Delen, D.: A hybrid framework using explainable AI (XAI) in cyber-risk management for defence and recovery against phishing attacks. *Decis. Support Syst.* **177**, 114102 (2024)
6. Rawindaran, N., Jayal, A., Prakash, E., Hewage, C.: Cost benefits of using machine learning features in NIDS for cyber security in UK small medium enterprises (SME). *Future Internet* **13**(8), 186 (2021)
7. Chandrakala, T., NirmalaSugirthaRajini, S., Selvam, K., Dharmarajan, K.: An Investigation for detecting cyber security to overcome cyber attacks using machine learning methods. *NeuroQuantology* **20**(15), 3581 (2022)
8. Farayola, O.A.: Revolutionizing banking security: integrating artificial intelligence, blockchain, and business intelligence for enhanced cybersecurity. *Financ. Account. Res. J.* **6**(4), 501–514 (2024)

9. Mun, J., Housel, T.: Artificial intelligence and machine learning applications to navy ships: cybersecurity and risk management. *Nav. Eng. J.* (2023)
10. Narsimha, B., Raghavendran, C.V., Rajyalakshmi, P., Reddy, G.K., Bhargavi, M., Naresh, P.: Cyber defense in the age of artificial intelligence and machine learning for financial fraud detection application. *IJEER* **10**(2), 87–92 (2022)
11. Alkhudaydi, O.A., Krichen, M., Alghamdi, A.D.: A deep learning methodology for predicting cybersecurity attacks on the internet of things. *Information* **14**(10), 550 (2023)
12. Rahmani, F.M., Zohuri, B.: The transformative impact of AI on financial institutions, with a focus on banking. *J. Eng. Appl. Sci. Tech. SRC/JEAST-279*. **5**(192), 2–6 (2023) doi. org/<https://doi.org/10.47363/JEAST/2023>
13. Hassan, M., Aziz, L.A.R., Andriansyah, Y.: The role artificial intelligence in modern banking: an exploration of AI-driven approaches for enhanced fraud prevention, risk management, and regulatory compliance. *Rev. Contemp. Bus. Anal.* **6**(1), 110–132 (2023)
14. Rawindaran, N., Jayal, A., Prakash, E.: Machine learning cybersecurity adoption in small and medium enterprises in developed countries. *Computers* **10**(11), 150 (2021)
15. Rawindaran, N., Jayal, A., Prakash, E.: Exploration of the impact of cybersecurity awareness on small and medium enterprises (SMEs) in Wales using intelligent software to combat cybercrime. *Computers* **11**(12), 174 (2022)
16. Mishra, S.: Exploring the impact of ai-based cyber security financial sector management. *Appl. Sci.* **13**(10), 5875 (2023)
17. Shahbazi, Z., Byun, Y.C.: Machine learning-based analysis of cryptocurrency market financial risk management. *IEEE access* **10**, 37848–37856 (2022)
18. Darem, A.A., Alhashmi, A.A., Alkhalidi, T.M., Alashjaee, A.M., Alanazi, S.M., Ebad, S.A.: Cyber threats classifications and countermeasures in banking and financial sector. *IEEE Access* **11**, 125138–125158 (2023)
19. Mughaid, A., AlZu'bi, S., Hnaif, A., Taamneh, S., Alnajjar, A., Elsoud, E.A.: An intelligent cyber security phishing detection system using deep learning techniques. *Cluster Comput.* **25**(6), 3819–3828 (2022). <https://doi.org/10.1007/s10586-022-03604-4>
20. Rodrigues, A.R.D., Ferreira, F.A., Teixeira, F.J., Zopounidis, C.: Artificial intelligence, digital transformation and cybersecurity in the banking sector: A multi-stakeholder cognition-driven framework. *Res. Int. Bus. Financ.* **60**, 101616 (2022)
21. Prasetyowati, S.A.D., Ismail, M., Budisusila, E.N., Purnomo, M.H.: Dataset feasibility analysis method based on enhanced adaptive LMS method with Min-max normalization and fuzzy intuitive sets. *Int. J. Electr. Eng. Inf.* **14**(1), 55–75 (2022)
22. Hafsa, A., Fradi, M., Sghaier, A., Malek, J., Machhout, M.: Real-time video security system using chaos- improved advanced encryption standard (IAES). *Multimedia Tools Appl.* **81**(2), 2275–2298 (2022). <https://doi.org/10.1007/s11042-021-11668-4>



Swarm Learning Based Federated Learning for Data Privacy in Cloud Environment

Madhavi Najana¹ , Arun Pandiyan Perumal² , Pradeep Chintale³ ,
Laxminarayana Korada⁴ , Ankur Mahida⁵ , and Sri Harsha Vardhan Sanne⁶

¹ Software Engineering Manager, Columbus, OH, USA

² Technology Infrastructure Specialist, Fremont, CA 94538, USA
apandiyan@hawk.iit.edu

³ Lead Solution Architect, SEI Investment Company, Downingtown, PA 19335, USA

⁴ Director - Partner Technology, Microsoft Corporation, Bellevue, WA, USA

⁵ SRE, Barclays PLC, Whippany, NJ, USA

⁶ Wesley Chapel, Wesley Chapel, FL, USA
sriharsha.sanne@west.cmu.edu

Abstract. Federated Learning (FL) is Machine Learning (ML) algorithm which performs the collaborative and distribute learning when private information present locally on client. To ensure the privacy of data, gradients are deformed or their representation is disturbed before sharing, that minimizes the performance of method. Recently, existing methods has drawbacks like problem of gradient leakage of FL in cloud that undermines data privacy. In this research, the Swarm Learning (SL) based Federated Learning (FL) in cloud environment is proposed to enhance the data privacy in cloud. In SL based FL, instead of sharing the disturbed or gradients with noise to central server, here shares the actual gradients between authenticate training nodes. The performance of proposed SL based FL method is evaluated with metrics of Time To Deploy (TTD), Time To Complete (TTC), utilization of Central Processing Unit (CPU) and Random Access Memory (RAM) on various cloud nodes (C1, C2, C3, C4). The proposed SL based FL method attained the less TTD of 27s, TTC of 4.18h and CPU Utilization of 0.34% on C1, 1.15% on C2, 1.64% on C3 and 1.98% on C4 which is effective than other techniques like Fuzzy Analytical Hierarchy Process (FAHP).

Keywords: Cloud environment · Federated Learning · Swarm Learning · Time To Complete and Time To Deploy

1 Introduction

In rapid developing landscape of software development, the DevOps adoption practices have indicated the new era of effectiveness, collaboration and agility in cloud organizations [1]. The DevOps is the combination of “Development” and “Operations” have revolution in the way the software is developed, testes and deployed [2]. The transformation method highlights the combination and automatic of different phases of software development lifecycle, enables the organization for bringing the services and products

to marketing much reliably and rapidly than before [3–5]. The DevOps introduced the continuous and iterative method for delivering the software, where modifications is introduced the quickly and without friction which classically developed the lifecycles [6–8]. The outcome is much effective, responsible and customer-concentrated method for developing the software which is suitable for demands of digital age [9]. When, the DevOps has several benefits like accelerating the time-to-market and enhanced collaboration that presented the unique challenges, specifically in security [10].

Federated Learning (FL) is the capable Machine Learning (ML) model to train method by utilizing the decentralized and sensitive information with high importance on information privacy preservation [11–13]. In the FL, the collaboration of local nodes for training the shared method, enables the method for processing particular tasks such as regression or classification without reveals its raw data to server of FL [14]. The FL have important attention across different domains includes finance, medicine, automatic vehicles and telecommunications [15]. The FL has problem of gradient leakage in training of decentralized and the method is needed for ensuring data privacy of every training node. In this research, proposed the Swarm Learning (SL) method based on blockchain network for resolving problem of gradient leakage.

The Significant contributions of the research is given as:

- The Swarm Learning based Federated Learning (SL based FL) method is proposed in cloud environment to enhance the privacy of data by mitigating the gradient leakage issue.
- The Swarm Learning is integrated with FL method that shares the original data to the training cloud clouds that tackles the issue of gradient leakage issue.

The remaining section of the research is organized as: Sect. 2 analyzes literature review of existing methods. Section 3 provides the details of proposed methodology. Section 4 gives the results and discussion of proposed methodology. Section 5 concludes the research.

2 Literature Review

In this section, the existing methods developed in cloud DevOps environment for enhancing the privacy of data. The recent methods are analyzed and described with their advantages and disadvantages.

Diyyala Sravani et al. [16] suggested the advanced Artificial Intelligence (AI) driven device to monitor continuous security in DevOps. The solution controlled the power of AI for detecting the proactively, analyzing and responding to secure threats in real time. Through combining the AI-depended device to DevOps pipeline, organization safeguards its environments when maintaining the rapid and agility deployment which offered by DevOps. The suggested method explored technical architecture, AI methods and employed methods in security monitoring. The suggested method identifies and mitigate the threats and vulnerabilities in different DevOps environments. However, the suggested method restricted in identifying new vulnerabilities that resulted in false negatives.

Kaleb Horvath et al. [17] presented the Smart Building Energy Fault Detection (SBFD) on cloud depended workspace of data bricks, the High-Performance Computing (HPC) environment through Apache Spark. Through ignoring the diagnosis of smart building as service method and kept design of tight centralization, rapid growth and deployment of cloud depended SBFD device was attained. The presented method developed the interface, continuous pipeline of real-time ingestion, combination, cleaning and analysis workflow able of energy consumption prediction and detection of anomaly was deployed and implemented in cloud. The presented method produced false negatives and not process with rapidly changing environments.

Gopalakrishnan Sriraman and Shriram R [18] implemented the preliminary analysis of interviews with relevance stakeholders. The ground truths were estimated and assigned for evaluating the different ML techniques for comparing its accuracy and testing the hypothesis. The implemented method practitioners and researcher understands the DevOps adoption and contexts in that DevOps practices were viable. The experimental outcomes showed that ML predicted the readiness organization for adopting the DevOps. However, the implemented method required huge storage capacity and time-consuming for setting and configuring.

Muhammad Azeem Akbar et al. [19] developed the factors for identifying and prioritizing factors which positively influence the DevOps practices in organizations of software. By utilizing 19 factors were found. The found factors were next validated with experts through the questionnaire survey. At last, Fuzzy Analytical Hierarchy Process (FAHP) was utilized for prioritizing the found success factors. However, the developed method has accelerated development pace, automatic testing and frequent deployments developed the expanded attacking surface that increased the security risk threats and vulnerabilities.

Juan Marcelo Parra-Ullauri et al. [20] suggested the Kubernetes based FL (KFL) method for privacy of FL through allowed the FL clients to access other resources by their flat networking method. The suggested method introduced the privacy preserving the K8s in kubeFlower process. That addressed the privacy risks through isolation through developing and differential privacy for information management. The suggested method includes the noise for information when managing the privacy budget and the kubeflower simplified the FL system in K8s that ensured the privacy. However, the suggested method has complexity in management and integration.

From the above analysis, the existing methods has drawbacks of restricted in identifying new vulnerabilities that resulted in false negatives, produced false negatives and not process with rapidly changing environments, data privacy and compliance and depended in third-party cloud providers, required huge storage capacity and time-consuming for setting and configuring, increased the security risk threats and vulnerabilities and complexity in management and integration. To mitigate these limitations, in this research proposed the SL based FL in cloud environment is proposed to enhance the data privacy in cloud. In SL based FL, instead of sharing the disturbed or gradients with noise to central server, here shares the actual gradients between authenticate training nodes.

3 Proposed Methodology

In proposed methodology, the multiple nodes (clients) equally trained the method with support of randomly chosen sentinel node which is responsible for gradients aggregation. In Federated Learning, every node trained local method with its own local information and shared the gradients with authentication node by the strategy of blockchain communication. The randomly chosen sentinel node is responsible for gradients broadcasting and aggregation. At last, every node evaluated their local method with gradients received from sentinel node. In this research, Swarm Learning (SL) based FL is proposed for confidentiality of data and decentralized clinical ML. The SL is applied to FL that resolves the issue of decentralized clinical information which can't share for ML tasks this is defined as gradient leakage issue in the FL. The combination of SL based FL into the cloud that improves the ability of cloud and gives the robust infrastructure for decentralized method training. The SL based FL in cloud achieved the data privacy through decentralized data processing and differential privacy. Figure 1 represents the process of SL based FL in cloud.

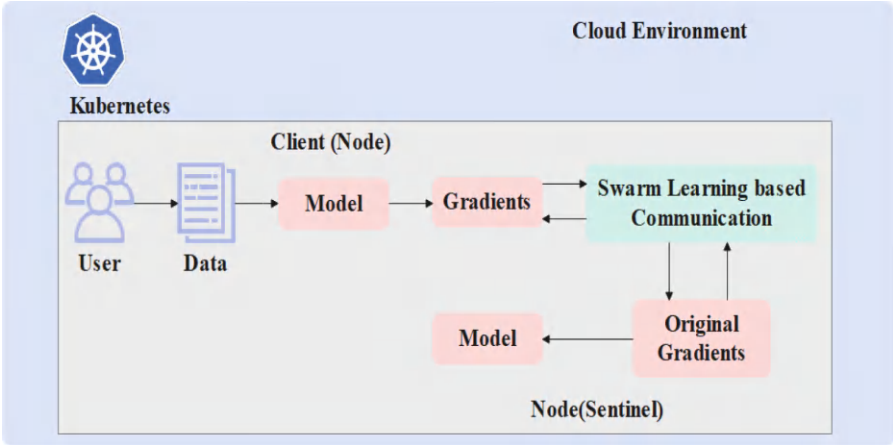


Fig. 1. Process of SL based FL in cloud

3.1 System Model

The system model for cloud-based FL involved the components and architecture which facilitates the secure and collaborative learning. The model included clients (nodes) which produces and hold data, central server and cloud infrastructure which supports this process. The microservices delivery, service chains and whole application and promising outcomes of network is evaluated extensively because of huge push in combination of containers across Internet Service Provider (ISP) and cloud environments provider with hybrid cloud combinations and optimization of data offloading. These types of cloud-native implementations are initialized across full network computing environment from the access of network optimization, and placement of data optimization purpose.

Here, considered the cases in that functions and data utilized in environments are sensitive for straight interaction and required to isolated though isolation of network, differential privacy and federated Kubernetes environment. The various kinds of isolation required gives a huge attack surface, dynamic and decentralized environments and at similar time, context particular environments in relevance domains of target of FL techniques.

The decentralized system is managed and implemented effectively with implementations of Kubernetes within the cloud provider premises to information and there has huge push considered the heavy reliance environment on concepts of network and distribution of networking towards the network depended DevOps that is known as Net Ops. Here, considered the Kubernetes is the effective option for demonstration and implementation of solution for background, architecture and requirements. The decision is done depended on wide spread influence of Kubernetes and also the decentralized nature ad hue system dynamic and that is easy for deployment in cloud environments.

3.2 Network Implications

The benefits of Kubernetes (K8s) offered to FL environments in cloud scenarios, the existing method have hugely the privacy challenges these solutions introduces while deployments in production settings. The main concern revolved in K8s multi-layer of abstraction is utilized to simplify development and deployment of contained workloads that is networking method. The K8s ensures the particular group of needs in third-party network solutions. That manages which pods should capable for communicating with whole another pods on K8s node without utilization of Network Address Translation (NAT). The popular network plugins such as Calico and Flannel implementing on flat networks. That represents whole pods communicates at IP layer, irrelevance of its position in cluster.

Though, the flat-network method has potential risk to privacy of information sharing in cluster that allowed whole containers for straightly interacting with one another. In cloud environment, there has probability of malicious entity communication with clients of FL which holds on sensitive data, by door opening to privacy relevance attacks. For addressing this issue, one mitigation strategy involved development of virtual networks which particular clients connected to enable the communication among pods merged to similar virtual network. That is significant to note that K8s process help this behaviour.

3.3 Kubernetes (K8s) – Privacy Preserving Cloud-Native FL

The K8s gives the infrastructure which can't simplify the deployment of FL method and scales the offer flexible and cost-efficient computation of FL system in cloud environments. Though, that introduced the challenges in terms of orchestration of workload and privacy concerns. The K8s can't automatically the FL application particular process. Additionally, that inherits the system level risks provided their flat network method.

3.4 Swarm Learning Based FL

Hence, every node sends and receive the updated recursively for particular count of communication rounds till the provided restriction is reached. Whether there are N .

nodes participated with whole information D . in process of training and node $i \in N$ have local information d_i and method m , next loss of individual node is represented as $f_i(m, d_i)$. The aim of aggregation method is to reduce loss of aggregation in every training node and the numerical expression is given as Eq. (1),

$$\min_{m \in R^d} \left\{ f(m, D) = \sum_{i=1}^N \frac{s_i}{N} f_i(m, d_i) \right\} \quad (1)$$

In above Eq. (1), the s_i represents the node samples of $i \in N$. There has multi communication rounds between participated nodes and loss are measured through mean of every communication rounds. The mathematical expression for updation on sentinel node for communication round r is given as Eq. (2),

$$\circledast^{r+1} = \sum_{i=1}^N w_i \circledast_i^r \quad (2)$$

In above Eq. (2), the w_i represents the aggregation weight and $\circledast_i^r$ describes local method of node i . The utilization SL based communication protocol for mitigating issue of gradient leakage with equally securing contact between nodes. Every participated node in process of training is authorized, registered and well-known for network. By utilizing the protocol of blockchain communication, the new participant requires for enrolling with smart contracts, next acquire the training method and continues the procedure till that synchronized with previous process of training. The Swarm application interface is utilized for integrating and exchanging the method parameters before every training process. At individual node, the SL is separated to application layer and middle-ware in that application layer gives methods, when the middle-ware gives blockchain, Swarm API and platform of ML. The mathematical expression for output gradients y acquired from individual node is given as Eq. (3),

$$y = M(x) \quad (3)$$

In the above Eq. (3), the $M(x) = \nabla \mathcal{L}(f(x), c)$ is measuring of gradients from loss $\mathcal{L}$ produced through input x and c . label in f method. In FL, the many recent techniques utilized the degrading factor ε included to gradients with certain transformation τ . Whether required before sharing the gradients with central server. The mathematical expression for process is given as Eq. (4),

$$y = \tau(M(x)) + \varepsilon \quad (4)$$

last mathematical expression for gradient is to mean at sentinel node is given as Eq. (5),

$$y = \nabla \mathcal{L}(f(x), c) \quad (5)$$

In proposed SL based FL method, the actual gradients are preserved and shared with authentication participants which process effective and perturbed the gradients in FL. The SL based FL in cloud achieved data privacy through decentralized data processing and differential privacy.

4 Experimental Analysis

The proposed SL based FL method is simulated with programming language of python and deployed by utilizing cloud native technology like Kubernetes. The system requirements are Operating System (OS) of windows 10, where HP license server is installed in sentinel and participating nodes are merges by SSH and shares learning with authentication certificates.

The below Table 1 represents simulation parameter for different Virtual Machines (VMs) with nodes' role, Control Processing Unit (CPU), Random Access Memory (RAM), storage and Graphics Processing Unit (GPU). In Table 1 the 4 private clouds (C1, C2, C3 and C4) with various specifications.

Table 1. Simulation Parameters

Nodes	Roles	CPU	RAM	Storage	GPU
C1	Control-plane	4	16	80	A100
C2	Worker	8	16	80	P2000
C3	Worker	8	16	100	P2000
C4	Worker	12	16	80	A40

The performance metrics such as Time To Deploy (TTD), Time To Complete (TTC), CPU Utilization and RAM usage on various cloud nodes. For calculating the rime relevance metrics, recorded the duration from initial process of deployment to completion that gives the model effectiveness. The Utilization of CPU and RAM are monitored by FL rounds in various clouds. The performance of TTD, TTC and CPU Utilization of proposed SL based FL method is depicted in below Table 2. The existing techniques considered for evaluating the proposed SL based FL method are Hadoop Yet Another Resource Negotiator (Hadoop YARN), Open Shift, Docker and Kubernetes based FL (KFL). From the Table 2, it is clear that the proposed SL based FL method attained the less TTD of 27s, TTC of 4.18h and CPU Utilization of 0.34% on C1, 1.15% on C2, 1.64% on C3 and 1.98% on C4 which is effective than other techniques.

The performance of RAM Utilization of proposed SL based FL method is depicted in below Table 3. The existing techniques considered for evaluating the proposed SL based FL method are Hadoop YARN, Open Shift, Docker and KFL. From the Table 3, it is clear that the proposed SL based FL method attained the less RAM Utilization of 469 MB on C1, 1345 MB on C2, 1773 MB on C3 and 1385 MB on C4 which is effective than other techniques.

Table 2. Performance TTD, TTC and CPU Utilization

Methods	TTD (s)	TTC (h)	CPU Utilization (%)			
			C1	C2	C3	C4
Hadoop YARN	35	4.79	0.78	1.89	2.16	2.84
Open Shift	67	5.03	0.80	2.05	1.97	2.34
Docker	165	4.62	1.04	2.34	2.56	3.09
KFL	74	5.92	1.18	2.68	2.03	2.63
Proposed SL based FL	27	4.18	0.34	1.15	1.64	1.98

Table 3. Performance of RAM Utilization on Various clouds

Methods	RAM Utilization (MB)			
	C1	C2	C3	C4
Hadoop YARN	990	3067	3841	3280
Open Shift	823	2817	3490	2916
Docker	677	2045	2956	2590
KFL	592	1789	2378	1932
Proposed SL based FL	469	1345	1773	1385

Table 4. Comparative Analysis of Proposed method.

Methods	CPU Utilization (%)		RAM Utilization (MB)	
	C1	C2	C1	C2
SBFD [17]	1.23	1.88	1148	2841
FAHP [19]	0.87	1.23	940	2180
KFL [20]	0.56	1.59	673	1836
Proposed SL based FL	0.34	1.15	469	1345

In the below Fig. 2, the loss vs epochs representation is described. The loss performance of proposed SL based FL method is compared with existing KFL method. From the below Fig. 2, it is clear that the proposed SL based FL method has less loss while compared with existing KFL method which shows that proposed method performed well in cloud-privacy.

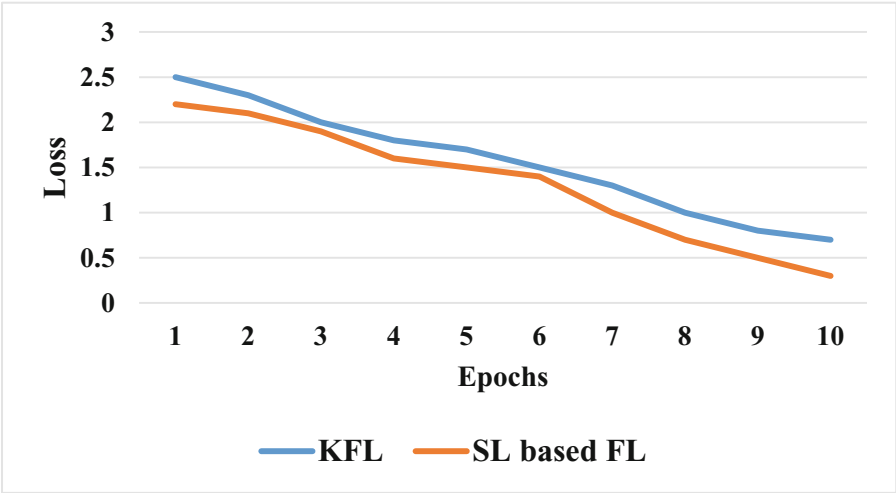


Fig. 2. Loss vs Epochs

In below Fig. 3, the accuracy of proposed SL based FL method is described with several FL rounds. By adjusting the privacy budgets across spectrum to much permissive values, the results amin to delicate the trade-off among privacy preserving and consequent effect on accuracy of method. In the below figure, the normal and Differential Privacy (DP) of 0.8 is evaluated for proposed SL based FL method. Through adjusting the DP across several spectrum from conservative to much promising values. The proposed SL based FL method is performed well in terms accuracy with several FL rounds.

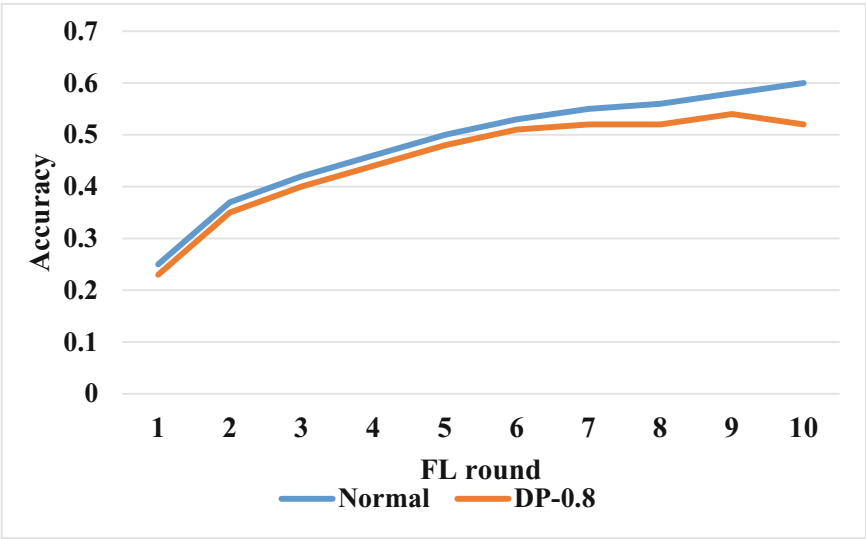


Fig. 3. Accuracy with Different Privacy

4.1 Comparative Analysis

The performance of proposed SL based FL method is compared to existing methods like Fuzzy-AHP [18], FAHP [20] and KFL [21] in terms of CPU and RAM Utilization on various cloud nodes. The existing technique values of [18, 20], and [21] are simulated according to the proposed method values. The proposed method attained less CPU utilization of 0.34% on C1 and 1.15% on C2, then less RAM Utilization of 469 MB on C1 and 1345 MB on C2 which is effective than other existing methods. The comparative analysis of proposed SL based FL is represented in the below table 4. By introducing the SL technique in FL method that mitigates the gradient leakage issue in FL decentralized method and improves privacy performance of FL in cloud environment.

4.2 Discussion

The introduced Fuzzy-AHP [18] method has limitations of data privacy and compliance and depended in third-party cloud providers. The developed FAHP [20] method has accelerated development pace, automatic testing and frequent deployments developed the expanded attacking surface that increased the security risk threats and vulnerabilities. The suggested KFL [21] method has complexity in management and integration. In this research, SL based FL in cloud environment is proposed to enhance the data privacy in cloud. In SL based FL, instead of sharing the disturbed or gradients with noise to central server, here shares the actual gradients between authentic training nodes. The performance of proposed SL based FL method is evaluated with metrics of TTD, TTC, CPU Utilization and RAM usage on various cloud nodes (C1, C2, C3, C4).

4.3 Conclusion

Recently, the existing methods has drawbacks like the issue of gradient leakage of FL in cloud. In this research, the SL based FL in cloud environment is proposed to enhance the data privacy in cloud. In SL based FL, instead of sharing the disturbed or gradients with noise to central server, here shares the actual gradients between authentic training nodes. The performance of proposed SL based FL method is evaluated with metrics of TTD, TTC, CPU Utilization and RAM usage on various cloud nodes (C1, C2, C3, C4). The proposed SL based FL method attained the less TTD of 27s, TTC of 4.18h and CPU Utilization of 0.34% on C1, 1.15% on C2, 1.64% on C3 and 1.98% on C4 which is effective than other techniques like FAHP. In future, different techniques can be used to further enhance the data security in cloud environment.

References

1. Rangaraju, S., Ness, S., Dharmalingam, R.: Incorporating AI-driven strategies in DevSecOps for robust cloud security. *Int. J. Innovative Sci. Res. Technol.* **8**(23592365), 10–5281 (2023)
2. Ozyurt, O., Gurcan, F., Dalveren, G.G.M., Derawi, M.: Career in cloud computing: Exploratory analysis of in-demand competency areas and skill sets. *Appl. Sci.* **12**(19), 9787 (2022)

3. Sorour, A., Hamdy, A.: Devops and iac to automate the delivery of hands-on software lab exams. In: 2022 6th International Conference on Computer, Software and Modeling (ICCSM) pp. 28–35. IEEE (2022)
4. Hrusto, A., Engström, E., Runeson, P.: Towards optimization of anomaly detection in DevOps. *Inf. Softw. Technol.* **160**, 107241 (2023)
5. Lévy, L.N., Bosom, J., Guerard, G., Amor, S.B., Bui, M., Tran, H.: DevOps model approach for monitoring smart energy systems. *Energies* **15**(15), 5516 (2022)
6. Shi, Z., Ivankovic, V., Farshidi, S., Surbiryala, J., Zhou, H., Zhao, Z.: AWESOME: an auction and witness enhanced SLA model for decentralized cloud marketplaces. *J. Cloud Comput.* **11**(1), 27 (2022)
7. Arulappan, A., Raja, G., Bashir, A.K., Mahanti, A. and Omar, M.: ZTMP: zero touch management provisioning algorithm for the on-boarding of cloud-native virtual network functions. *Mobile Netw. Appl.* 1–13 (2023)
8. Saboor, A., Mahmood, A.K., Omar, A.H., Hassan, M.F., Shah, S.N.M., Ahmadian, A.: Enabling rank-based distribution of microservices among containers for green cloud computing environment. *Peer-to-Peer Netw. Appl.* **15**(1), 77–91 (2022)
9. Cuadra, J., Hurtado, E., Sarachaga, I., Estévez, E., Casquero, O., Armentia, A.: Enabling DevOps for fog applications in the smart manufacturing domain: a model-driven based platform engineering approach. *Futur. Gener. Comput. Syst.* **157**, 360–375 (2024)
10. Erdenebat, B., Bud, B., Batsuren, T., Kozsik, T.: Multi-project multi-environment approach—an enhancement to existing DevOps and continuous integration and continuous deployment tools. *Computers* **12**(12), 254 (2023)
11. Fawzy, A.H., Wassif, K., Moussa, H.: Framework for automatic detection of anomalies in DevOps. *J. King Saud Univ. Comput. Inf. Sci.* **35**(3), 8–19 (2023)
12. Bahaweres, R.B., Zulfikar, A., Hermadi, I., Suroso, A.I., Arkeman, Y.: Docker and kubernetes pipeline for DevOps software defect prediction with MLOps approach. In: 2022 2nd International Seminar on Machine Learning, Optimization, and Data Science (ISMODE) (pp. 248–253). IEEE (2022)
13. Stirling, J., Bumke, K., Collins, J., Dhokia, V., Bowman, R.: HardOps: utilising the software development toolchain for hardware design. *Int. J. Comput. Integr. Manuf.* **35**(12), 1297–1309 (2022)
14. Wolfschwenger, P., Sabitzer, B. and Lavicza, Z.: Design and evaluation of an agile framework for continuous education in software engineering. In: 2022 IEEE Frontiers in Education Conference (FIE) (pp. 1–9). IEEE (2022)
15. Bhaskaran, A.P.: Analyzing effectiveness of security by design model in a cloud application development lif cycle which uses IAC (Infrastructure As Code)
16. Sravani, D., Viswas, P.S., Kiran, P.C., Reddy, J.R., Jyothi, N.M.: Safeguarding DevOps environments: AI-based continuous security monitoring. *Int. J. Intell. Syst. Appl. Eng.* **12**(16s), 430–438 (2024)
17. Horvath, K., Abid, M.R., Merino, T., Zimmerman, R., Peker, Y., Khan, S.: Cloud-based infrastructure and DevOps for energy fault detection in smart buildings. *Computers* **13**(1), 23 (2024)
18. Sriraman, G.: A machine learning approach to predict DevOps readiness and adaptation in a heterogeneous IT environment. *Front. Comput. Sci.* **5**, 1214722 (2023)
19. Akbar, M.A., Mahmood, S., Shafiq, M., Alsanad, A., Alsanad, A.A.A., Gumaei, A.: Identification and prioritization of DevOps success factors using fuzzy-AHP approach. *Soft. Comput.* 1–25 (2023)
20. Parra-Ullauri, J.M., et al.: KubeFlower: a privacy-preserving framework for Kubernetes-based federated learning in cloud–edge environments. *Futur. Gener. Comput. Syst.* **157**, 558–572 (2024)

AI for Agriculture



Detection of Tomato Leaf Disease Using Convolutional Neural Network

Manohar Madgi^(✉) , Vivek Verma, Prabhat K. Rai, Tsewang Norbu, and Vrashabh Kambayal

School of Computer Science and Engineering, KLE Technological University, Hubballi, Karnataka 580031, India

{manohar.madgi, 01fe21bcs019, 01fe21bcs250, 01fe21bcs021, 01fe21bcs231}@kletech.ac.in

Abstract. Agriculture plays a crucial role in our country, and tomatoes are a popular crop globally. It is used in various forms in different cuisines worldwide. India is the second largest producer of tomatoes. However, the quality and quantity of tomato crops often suffer due to various diseases. To address this issue, use the plantVillage dataset from Kaggle, which includes nine classes of tomato leaf disease images and one class of healthy tomato leaf images. The aim is to create an automated model to help farmers detect diseases early and intervene promptly, reducing crop losses. The process involves preprocessing the input images, segmenting the targeted area from the original images and further processing the images using relevant features to effectively predict and classify the diseases. The convolutional neural network (CNN) model yielded a 92.84% accuracy, which is better than the Inception v3, VGG16, and ResNet50 models considered in the work.

Keywords: Tomato leaf disease · CNN · Smart agriculture · Classification · Deep learning · Feature selection

1 Introduction

Agriculture serves as the backbone of our nation, and among the myriad crops that grace fields worldwide, the humble tomato holds a distinguished place. Whether it's diced into salads, blended into sauces, or featured in diverse cuisines, the tomato is an integral part of our daily culinary repertoire. In the global agricultural landscape, India proudly stands as the second-largest producer of tomatoes. However, the prolific growth of this vital crop faces a substantial challenge – the threat of various diseases that can compromise both the quality and quantity of the tomato harvest.

Recognizing the pivotal role tomatoes play in our kitchens and the agricultural sector. The work embarks on leveraging the vast plantVillage dataset obtained from Kaggle. This dataset encompasses a wide variety of images, categorizing tomato leaf conditions into nine distinct disease classes alongside a category representing the healthy state of tomato leaves shown in Fig. 1. The overarching goal is to develop a dependable and

automated tool, a technological ally for farmers, that can swiftly identify early signs of diseases. This proactive approach enables timely interventions, ultimately curbing crop losses and ensuring a bountiful harvest.

The foundation of our work lies in the exploration of the plant Village dataset. By delving into the intricacies of nine different classes of tomato leaf disease images and one class portraying healthy tomato leaves, the aim is to gain insights that will empower our automated tool. The goal is to create a seamless and reliable system that not only assists farmers but significantly contributes to the agricultural ecosystem.

The process begins with the preprocessing of input images, a crucial step that lays the groundwork for subsequent analysis. The targeted regions within these images are carefully delineated through segmentation techniques, ensuring a focused examination. The methodology involves further processing of these segmented images. This phase incorporates the extraction of relevant features and vital components that contribute to the effective prediction and classification of tomato leaf diseases.

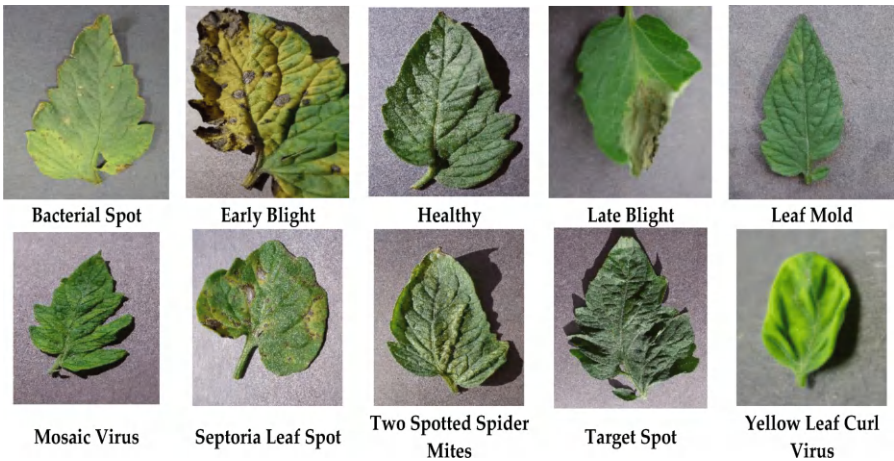


Fig. 1. Diseased tomato leaves image samples.

2 Literature Survey

Dr. Juan A. and Dr. Angelica [1] have suggested a model that uses a public dataset and additional images collected in the country’s field to identify and categorize tomato leaf diseases. The model is built on a CNN. The outcomes demonstrate the suggested model performs well in identifying and categorizing diseases in tomato leaves. In both training and test datasets, accuracy achieved is higher than 99%.

Sunil S. et al. [2] have used machine learning techniques like Support Vector Machine (SVM), CNN, and K-nearest neighbor (K-NN) to classify the diseases. On tomato-disordered samples, accuracy of the suggested model is evaluated using K-NN 97%, SVM 88%, and CNN 99%.

Omneya Attallah [3] has employed six classifiers to identify diseases in tomato leaves. Based solely on 22 and 24 features, the SVM and K-NN have achieved highest accuracy, 99.90% and 99.92%, respectively, according to the results.

Nawaz M. et al. [4] have presented a model that comprises three steps. First, they create region of interest (RoI) annotations for suspected images. Convolutional Block Attention Module (CBAM) was introduced as a feature extractor in stage two, along with ResNet-34. To train the Faster r-CNN model the computed features are used to identify and classify various tomato plant leaf anomalies.

Rangarajan A. K. et al. [5] in their study, two deep learning-based architectures, VGG16 and AlexNet, are trained with images of tomato leaves—six diseased and one healthy class taken from the plantVillage dataset. Analysis has been done on the impact of several images and the hyperparameters (minibatch size, learning rate, bias and weight) on the execution time and accuracy of the classification.

Dheeb B. et al. [6] presented ANN for leaf disease detection and identification and K-means clustering for segmentation. The developed model had a 93% accuracy rate in identifying the diseases. Among its many merits is its exceptional ability to diagnose diseases. Finer segmentation is one of the demerits, and feature extraction is needed.

Hsueh-Hung Cheng et al. [7] proposed a technique that uses chatbot controllers, CNNs, and images of tomato leaves to identify eight categories of pests and diseases automatically. About nine thousand images of tomato leaves were gathered in fields or green-houses affected by eleven diseases and pests. When identifying abnormal images, the trained ADM had a 97.40% accuracy rate. The images were classified into 8-categories of tomato leaf disease and pest; trained DIM obtained an accuracy of 93.63%. With training, the trained LPDM could discriminate between leaf mold and powdery mildew II with 98.70% accuracy.

Fuentes A. et al. [8] suggested an object recognition technique that combined the most recent deep CNN feature extractors with R-FCN, Faster r-CNN, and SSD algorithms. VGG16 with R-CNN, the suggested model, produced better recognition results. Reduced false positives during the training phase is the advantage. Lag in performance is the drawback.

Ashok S. et al. [9] presented a CNN based approach with a 98% classification accuracy rate. Among the merits is excellence in performance. Recent classifiers and algorithms that can be applied for the best results are among its strengths.

Nithish Kannan E et al. [10] developed a combined ResNet-50 model-based architecture with Deep CNN and achieved a 97% accuracy. Among its merits is a trained model that uses data augmentation to achieve best results. High configuration hardware, which is crucial for training, is one of the demerits.

3 Challenges

3.1 Dataset Acquisition

Obtaining the right dataset to differentiate the different types of tomato disease was tough. It needed to search carefully to find a dataset that was relevant and well organized. Making sure that the labels in the dataset were accurate and reliable and handling potential

biases became a crucial job. It was challenging to find the right balance between having a big enough dataset and ensuring its quality, especially considering our computer’s limitations. Dealing with ethical concerns about where and how to get data added another layer of complexity to the process.

3.2 Model Selection Dilemma

Picking the right level of model complexity was tricky. Here, think about how big the dataset was, how many features it had, and how much computing power it had. Trying to find a balance between getting the right mix of precision, recall, accuracy, and efficiency in the proposed model was challenging. Fine-tuning the model by adjusting its settings needed a deep understanding of how the algorithms work and the ability to experiment multiple times to get the best results. Choosing a good combination of models for the ensemble approach and figuring out how to set it up through experimentation was challenging in ensuring a diversity of models for the best outcomes.

4 Methodology

This section discusses cutting-edge methodologies, datasets, and models used to attain outcomes (see Fig. 2).

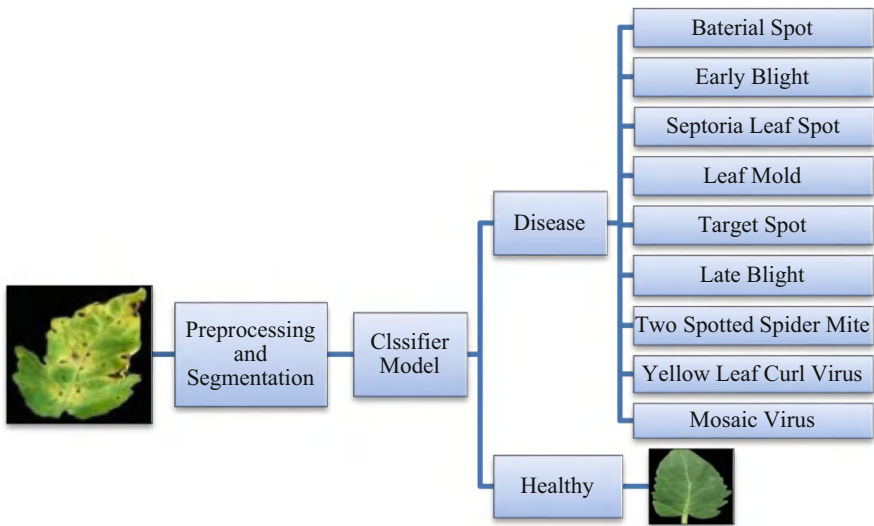


Fig. 2. Suggested Model.

4.1 Dataset Overview

Dataset utilized in this work is taken from plantVillage dataset, available on Kaggle, offering a diverse collection of images specifically curated for plant-related research. A

dataset comprising a total of 14,531 images, this dataset encompasses a wide variety of tomato leaf samples. These images are meticulously categorized into ten distinct classes, providing a comprehensive representation of various tomato leaf conditions. Within this dataset, one class is dedicated to healthy tomato leaves images, showcasing the standard appearance of disease-free foliage. The remaining nine classes correspond to different types of diseased tomato leaf, each representing a unique manifestation of plant ailments listed in Table 1. This diversity enables the development of a robust disease detection system capable of identifying and classifying a range of potential issues affecting tomato crops.

Table 1. Tomato leaf with disease sample images for training, testing and validation.

Category	Training (70%)	Validation (15%)	Testing (15%)
Healthy (1273)	891	191	191
Early Blight (800)	560	120	120
Leaf Mold (761)	533	114	114
Late Blight (1527)	1069	229	229
Taget Spot (1123)	787	168	168
Septoria Leaf Spot (1417)	991	213	213
Mosaic Virus (299)	209	45	45
Two Spotted Spider Mite (1341)	937	202	202
Bacterial Spot (1702)	1190	256	256
Yellow Leaf Curl Virus (4286)	3000	643	543
Total images (14531)	10171	4360	4360

4.2 Implementation

The input dataset is split class-wise to train, validate and test the CNN, as in Table 1. In the implementation phase, various models were applied for the task of detecting disease in tomato leaves, each offering unique advantages and characteristics.

The CNN served as the foundational model, leveraging its ability to effectively capture spatial hierarchies and patterns within the input images. This was complemented by the VGG16 model, renowned for its deep architecture and success in image classification tasks. VGG16 excels in learning intricate features through its numerous layers, enhancing the system's ability to discern subtle differences indicative of different diseases. Additionally, the ResNet50 model, with its residual learning framework, plays a key role in addressing the challenges of training deep neural networks. The residual connections within ResNet50 facilitate the smooth flow of information across layers, mitigating the risk of vanishing gradients and contributing to more efficient learning.

Beyond deep learning models, the implementation also explored the Random Forest algorithm. In contrast to neural networks, Random Forest operates as an ensemble learning method, aggregating the predictions of multiple decision trees. This diversification aids in capturing complex relationships within the data and provides an alternative approach to disease detection. Ultimately, the incorporation of CNN, VGG16, ResNet50, and Random Forest in the implementation phase was a strategic decision, allowing for a comprehensive evaluation of different modeling paradigms. This approach aimed to identify the most effective model or combination of models for accurate and reliable tomato leaf disease detection.

After careful evaluation of various models, the decision ultimately led us to choose the CNN model as the primary classifier for our application. The CNN architecture has consistently demonstrated superior performance in image recognition tasks, making it an ideal choice for our disease prediction system. Its ability to automatically learn features from input images, coupled with its hierarchical structure, enables it to effectively capture complex patterns and variations within leaf images.

The dataset is divided into training, testing, and validation to assess the performance of CNN model. The model was trained using the training dataset, the hyperparameters were adjusted using the validation dataset, and the model's performance was assessed using the test dataset.

The dataset images are randomly split into a training dataset of 70%, a validation of 15%, and a test of 15%.

The dataset split guarantees that the model is trained on a sufficiently sizable dataset and permits a fair assessment of its performance on untested data.

Convolution and max pooling layers are the CNN building blocks with multiple levels. The input data is provided in batches of size 32, and the model contains six levels with reducing output shape reduction, with a 3×3 kernel size and ReLU activation.

Performance Evaluation Metrics. Employed a suite of discerning performance metrics to holistically assess the CNN, Inception v3, VGG16, and ResNet50 model's efficacy. To assess the performance of each model, the accuracy, precision, recall, and F1-score metrics are used.

Accuracy: It's a statistic for assessing how accurate a model is overall at making predictions in categorization tasks. It is estimated by dividing the total number of examples in the dataset by the number of cases that were correctly predicted. The accuracy metric provides a general assessment of how well a model performs across all classes.

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \quad (1)$$

$$Precision = \frac{TP}{(TP + FP)} \quad (2)$$

$$Recall = \frac{TP}{(TP + FN)} \quad (3)$$

TP-True Positive, TN-True Negative, FP-False Positive, FN-False Negative.

F1-score: F1-score, also referred to as the F1 measure, is a metric that integrates re-call and precision into a single numerical value that is utilized in classification tasks. It strikes a compromise between the two measures, delivering a more thorough assessment of a model's effectiveness, especially when there is a disparity in class composition.

$$F1Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (4)$$

5 Results and Discussion

In the context of classifier evaluation, the graphical representation of accuracies serves as a powerful visual aid, facilitating comparative analysis and informed decision-making. Each model's accuracy percentage, depicted as a distinct bar, provides a clear snapshot of its performance relative to others. This clarity enables stakeholders to identify trends, patterns, and outliers, thus informing their selection of the most suitable classifier for a given task.

Firstly, examine the accuracy trend across each classifier; it becomes evident how each model fares in correctly classifying instances. The graph Fig. 3 shows average accuracy, and Figs. 4, 5, 6 and 7 depicts the categorical accuracy of each model of the work carried.

The graphical depiction of classifier performance, as showcased by the bar graph Figs. 4, 5, 6 and 7, illustrates the accuracies of four notable models CNN's highest classification accuracy is 99.85%, the lowest is 69.82%, and average is 92.84%, Inception v3 highest 86.90%, lowest 0.0%, and average 62.23%, VGG16 highest 98.30%, lowest 71.31%, and average 92.57%, and ResNet50 highest 92.45%, lowest 0.0%, and average 44.73%, this emerges as a pivotal analytical instrument for stakeholders in machine learning evaluation. Each bar, accompanied by its respective accuracy percentage, serves as a beacon of insight for decision-makers navigating the labyrinth of model selection. Beyond mere data presentation, these graphs offer a comparative framework, facilitating the discernment of nuanced performance disparities among the models.

The performance of each model considered in the work is listed in Table 2. Among the four models, CNN's performance is comparatively good.

Table 2. Comparison of accuracy, precision, recall and F1 score of selected models.

Model	Accuracy	Precision	Recall	F1 Score
CNN	92.84%	93.08%	92.84%	92.78%
Inception v3	62.23%	72.28%	62.23%	63.29%
VGG16	92.57%	92.60%	92.57%	92.39%
ResNet50	44.73%	52.05%	44.73%	35.25%

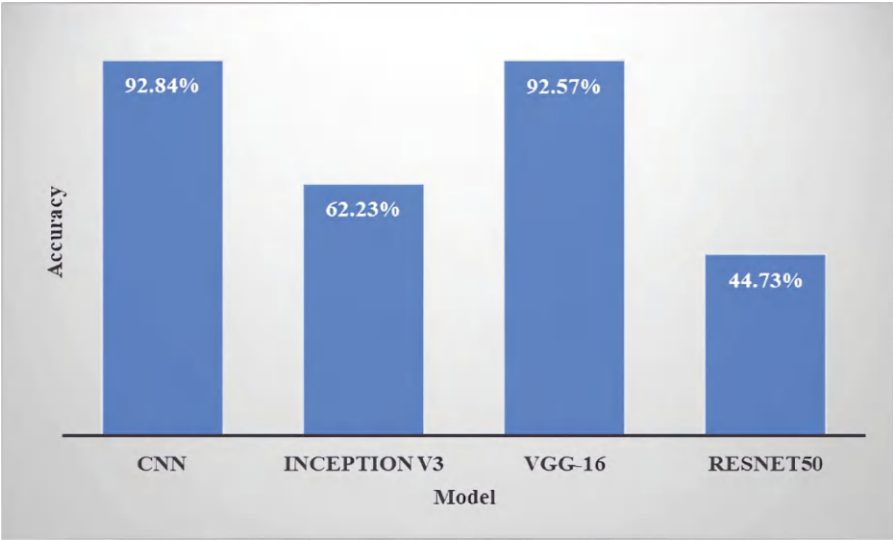


Fig. 3. Average disease detection accuracy of CNN, Inception v3, VGG-16, ResNet50 models.

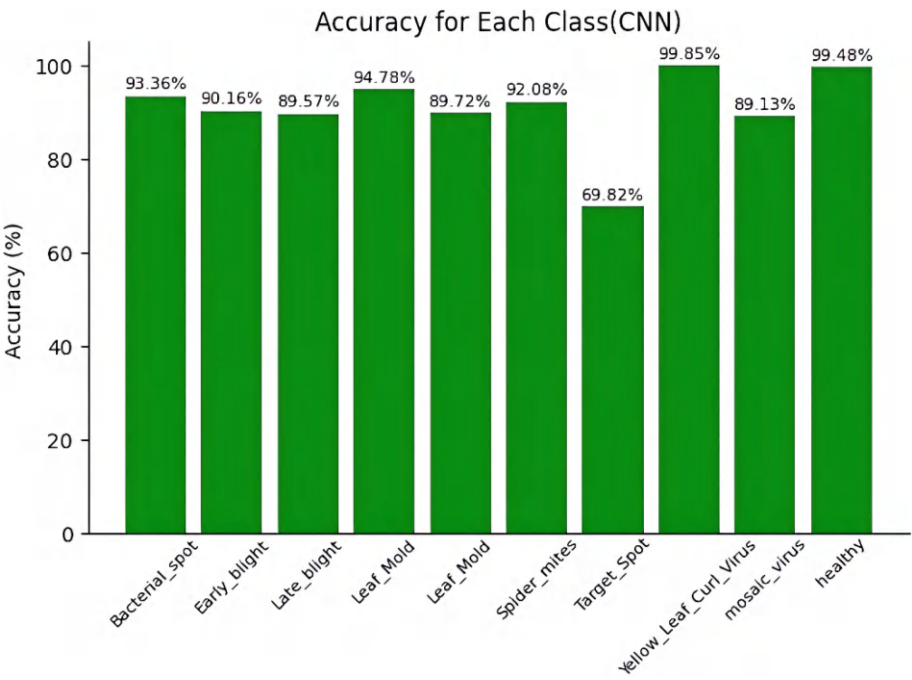


Fig. 4. Categorical classification of CNN model.

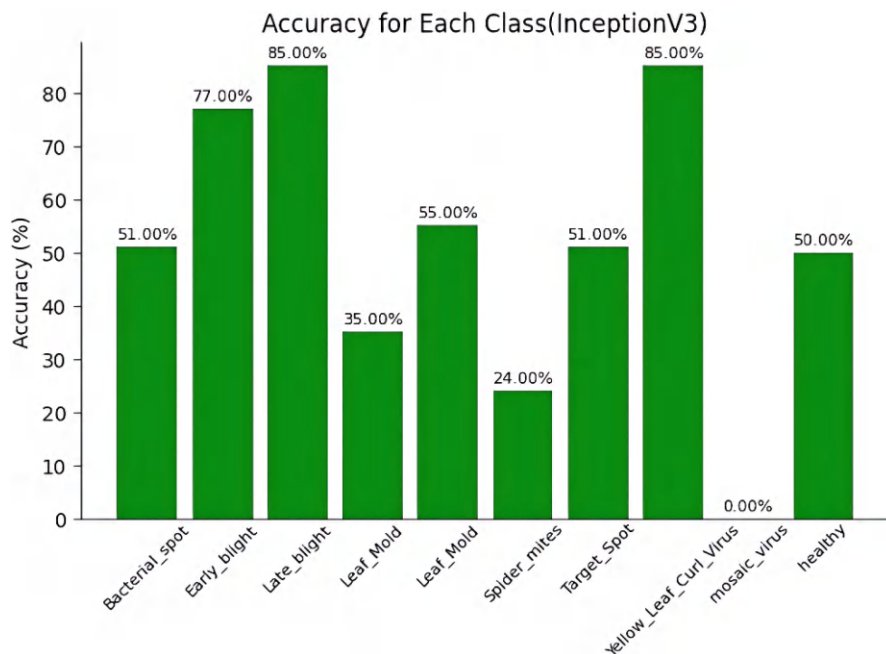


Fig. 5. Categorical classification of Inception v3 model.

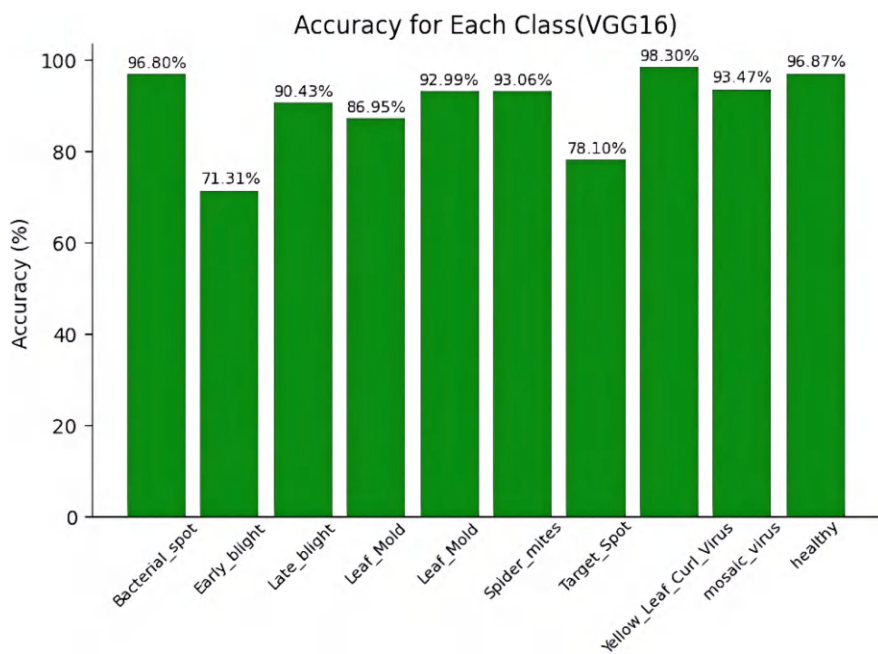


Fig. 6. Categorical classification of VGG16 model.

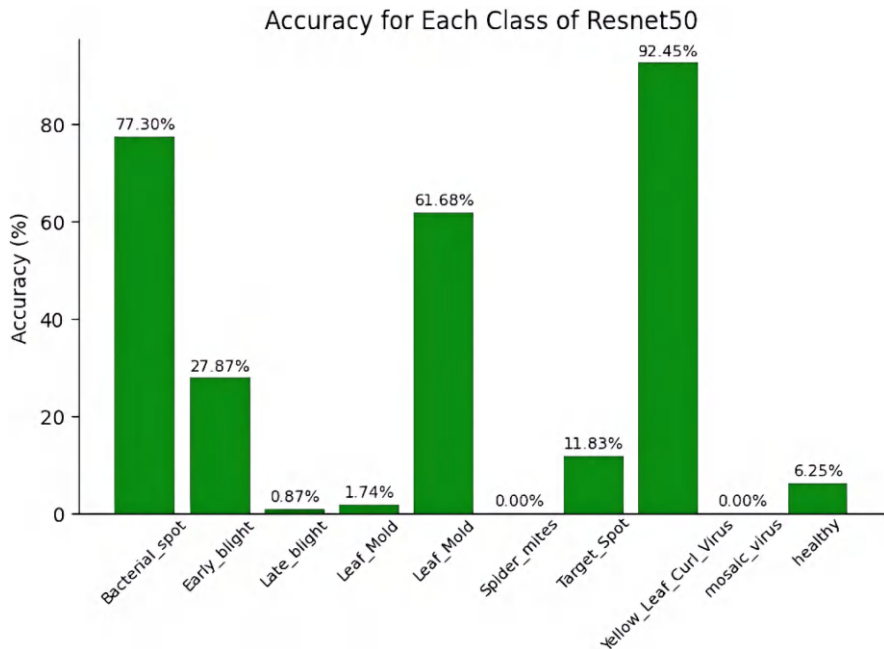


Fig. 7. Categorical classification of ResNet50 model.

6 Conclusion

The work presents a method of detecting tomato leaf disease using CNN, Inception v3, VGG16, and ResNet50. Each model is trained using a dataset of tomato leaf images gathered from many sources. The trained models were able to accurately detect the presence of nine common tomato leaf diseases and a healthy leaf, namely, mosaic virus, bacterial spot, healthy, yellow leaf curl virus, late blight, Septoria leaf spot, two-spotted, target spot, spider mite, early blight and leaf mold. The proposed model initially achieved optimal accuracy and an F1 score. To enhance performance, finetuning or incorporating Gabor filters into the CNN can be explored. The CNN approach yielded a remarkable 92.84% accuracy, which is better than the other three models considered in the work. Continued refinement promises increased precision in detecting and classifying tomato leaf diseases.

Data Quality and Quantity: One significant gap was the availability of high-quality and diverse datasets for training the CNN model. The suggested model performance could be further enhanced with a more extensive and varied dataset covering a wider range of disease manifestations.

References

1. Antonio, G., Angelica, R.: Monitoring tomato leaf disease through convolutional neural networks. *Electronics* **12**(1), 229 (2023). <https://doi.org/10.3390/electronics12010229>

2. Sunil, S.H., Jayashri, M.R., Veena, I.P., Ayesha, S., Pramodhini, R.: Plant leaf disease detection using computer vision and machine learning algorithms. *Global Transitions Proc.* **3**(1), 305–310 (2022). <https://doi.org/10.1016/j.gltp.2022.03.016>
3. Attallah, O.: Tomato leaf disease classification via compact convolutional neural networks with transfer learning and feature selection. *Horticulture* **9**(2), 149 (2023). <https://doi.org/10.3390/horticulturae9020149>
4. Nawaz, M., Nazir, T., Javed, A., et al.: A robust deep learning approach for tomato plant leaf disease localization and classification. *Sci. Rep.* **12**, 18568 (2022). <https://doi.org/10.1038/s41598-022-21498-5>
5. Rangarajan, A.K., Purushothaman, R., Ramesh, A.: Tomato crop disease classification using pre-trained deep learning algorithm. *Procedia Comput. Sci.* **133**, 1040–1047 (2018). <https://doi.org/10.1016/j.procs.2018.07.070>
6. Dheeb, B., Malik, B., Sulieman, B.: Detection and classification of leaf diseases using k-means-based segmentation and neural-networks based classification. *Inf. Technol. J.* **10**(2), 267–275 (2011). <https://doi.org/10.3923/itj.2011.267.275>
7. Cheng, H.-H., et al.: Identifying tomato leaf diseases under real field conditions using convolutional neural networks and a chatbot. *Comput. Electron. Agric.* **202**, 107365 (2022). <https://doi.org/10.1016/j.compag.2022.107365>
8. Fuentes, A., Yoon, S., Kim, S.C., Park, D.S.: A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors* **2017**, **17**(9) (2022). <https://doi.org/10.3390/s17092022>
9. Ashok, S., Kishore, G., Rajesh, V., Suchitra, S., Sophia, S.G.G., Pavithra, B.: Tomato leaf disease detection using deep learning techniques. In: 2020 5th International Conference on Communication and Electronics Systems (ICCES), pp. 979–983. Coimbatore, India (2020) <https://doi.org/10.1109/ICCES48766.2020.9137986>
10. Nithish Kannan, E., Kaushik, M., Prakash, P., Ajay, R., Veni, S.: Tomato leaf disease detection using convolutional neural network with data augmentation. In: 2020 5th International Conference on Communication and Electronics Systems (ICCES), 2020, pp. 1125–1132. Coimbatore, India (2020). <https://doi.org/10.1109/ICCES48766.2020.9138030>



Pbstam: A Proactive Borewell Safety Using Teachable Arduino Model

Sri Silpa Padmanabhuni¹(✉), D. Durga Prasad², Perumalla Prasanna Siri¹,
Repalli Vineetha¹, and Vejandla Pavithra Prasad¹

¹ Department of Computer Science and Engineering (IoT and CS Incl BCT), Potti Sriramulu
Chalavathi Mallikarjuna Rao College of Engineering and Technology, Vijayawada, A.P., India
silpa.padmanabhuni@gmail.com

² Department of Computer Science and Engineering, Potti Sriramulu Chalavathi Mallikarjuna
Rao College of Engineering and Technology, Vijayawada, A.P., India

Abstract. India is a developing country where people depend on natural resources like water, air, land, and others. People extract these natural resources by digging borewells. It is observed that many of the borewells are left uncovered due to various reasons. These uncovered bore wells are the main reasons for the death of children and animals, who accidentally fall into it. According to the Indian census, falling inside the borewells has increased abundantly over the past decade. Many researchers tried to solve this problem using IoT devices which is a time-consuming process and provides only post-rescue solutions. This model strongly believes “Prevention is better than cure”. So, it integrates teachable ML with IoT devices to prevent objects from falling into the borewell. Integrating ML with IoT will automate the process of bore well and save many lives. The main advantage of using Teachable ML is it has a lot of images related to humans, birds, and animals which helps us to prevent training the model from scratch. In this model, if it recognizes an animal or a human being then it automatically closes the door. The model automatically recognizes the darkness of the environment and closes the door throughout the night. All the existing systems provide solutions with IoT devices, but they need human intervention which is automated in this model. The model also provides security by authenticating the system with automatic recognition of the corresponding higher officials. Even after providing prevention measures by chance, if any object falls into the borewell, the corresponding authority will receive an alert notification and a quick rescue team will reach the borewell location.

Keywords: Teachable ML · IoT (Internet of Things) · Post-rescue · Face recognition · Automation · Authentication

1 Introduction

Water is an essential resource for life to sustain. In this modern era due to climatic changes and other environmental conditions, water has been scarce for agriculture and as well as for domestic usage due to this problem, many borewells are being dug to

extract underground water [11]. However, few of these borewells become worthless as the groundwater level goes beneath accessible depth. Consequently, these wells become abandoned and few of them are left open due to ignorance whereas some of them are left open for the future purpose. These open wells have grown into a death trap for children and animals as they fall into these wells unintentionally while moving around them [12]. These incidents have become familiar in our country, particularly in the southern states. The emergency response team struggles a lot to rescue the victims in this kind of situation and it takes several hours with the huge workforce to rescue the kid or animal [14]. Such a rescue operation has only a success rate of 40 percent as the survival of the child is uncertain in such rigid surroundings.

According to previous research, most of the proposed systems worked on post-rescue operations that would take much time. The paper titled “Child Rescue System from Open Borewells” suggested that the rescue operation could be done by digging another hole alongside the bore well and then a horizontal path is made to reach the child [15]. A CCTV camera is used to observe the child’s condition continuously. This system has a high-power LED which acts as a light source since light intensity inside the hole will be less, which is a time-consuming process and has less possibility to rescue the child. The limitations in the paper are: that the child may lose her/his life if there is any mislead in the process of digging the parallel hole. There is no assurance for continuous lighting conditions. During the digging process, the child may lose his/her life if the operation fails in any situation because of the negligence in the manual rescue process.

1.1 Need for Borewell Automation Using Teachable ML

Teachable Machine is accessible to a wide range of users, even those with no technical understanding, because of its user-friendly interface that is accessible through a web browser. Because of its accessibility, communities and organizations can design and implement object detection systems close to borewells without needing specialized knowledge. Without needing to know a lot about programming or machine learning, Teachable Machine lets people train their machine-learning models. In such a way anyone or any organization that is worried about the safety of people around borewells could design customized models to identify things or risks in the area, such as humans, animals, or potentially dangerous objects. Organizations can construct real-time surveillance of the surrounding area by integrating Teachable Machine models with cameras or sensors positioned close to borewells. This makes it feasible to quickly recognize and notify others about any possible risks or dangerous circumstances, such as unwanted entry, falling objects, or the presence of people or animals near the borewell. Eventually, communities or organizations can obtain good insights into potential threats by gathering data on object detection around borewells. With the help of this input data, decisions on safety precautions, upgrades to the infrastructure, or community awareness campaigns that try to stop accidents and encourage safe behavior can be made. Teachable Machine models offer a modest way to apply object detection systems widely since they can be scaled and used across several borewell sites. This potentiality to scale is especially helpful for areas or communities that contain many borewells that need to be monitored and secured.

2 Literature Survey

[1] Kavianand. G et al. have addressed the issue of child safety near the empty borewells, which had turned into death traps for toddlers in India because of carelessness and improper covering. The author of this paper states that the lives of the trapped kids are at risk with the current rescue methodologies, which require a lot of labor and heavy machinery. In response, this presents a unique method that uses PIR sensors to identify human beings around borewells. The gadget shuts the borewell horizontally if it detects the presence of a child, stopping further descent. It further speeds up the response time by notifying authorities and firefighters through texts and sirens. The SCRS has many advantages over conventional methods, such as faster rescue times, less danger to children's lives, and cost-effectiveness. This is achieved via the Raspberry Pi as the main controller and integrating GSM/GPRS modules for communication. To stop terrible tragedies and prevent children from heading into borewells, this original approach is an essential initial step. However, further research and improvement may be needed to ensure successful and broad application.

[2] Sriramireddy. K et al. have solved the important problem of saving kids who have fallen into bore wells. In this study via a simulation process with a suggested robot design. Unmanaged and abandoned bore wells lead to serious hazards to the safety of kids in India, as bore wells are essential in providing water for the increase in population. Although there are multiple bore-well rescue techniques, there isn't a national database on bore-well conditions to help rescuers quickly obtain essential data. With an eye on building a simulated bore-well analysis database, the authors give a framework for bore-well inspection and rescue. Studies of actual time case studies taken in Andhra Pradesh, India, show the importance of collecting information on bore-well conditions, such as temperature, humidity, and air quality, to properly plan rescue operations. The condition of a trapped child decreases with increasing depth, according to evidence provided by models with ZEISS ZEN software. This information makes it easier to select the best methods for rescue based on the data gathered. This integrated method presents a viable way to successfully reduce the risk to the lives of kids and accelerate bore-well rescue operations.

[3] S Renuga Devi et al. suggested a rescue mechanism with the help of Drone technology. The strong increase in accidents involving unsecured borewells has gained attention in recent years, largely due to horrible outcomes for many children who get stuck in them. A clever approach using drone technology with a camera and ultrasonic sensors has been put forth in recognition of the urgent need to stop such occurrences. The drone camera identifies the potholes in remote areas and helps in measuring the depth of those pothole. Drones can be used by officials to quickly seal dangerous borewells and reduce the likelihood of additional accidents by identifying and measuring unprotected borewells in distant places. By preventing accidents and providing prompt response through cloud-based data management, this preventive approach has the potential to save many lives and solve a critical societal issue.

[4] Nitin Agarwal et al. addressed the repeated incidents of children falling into borewells due to societal carelessness and emphasized the urgent requirement for more effective and efficient rescue methods. Existing techniques, such as making horizontal paths and cutting parallel holes, are risky and time-consuming. With its ability to operate

within the same borewell where the child is trapped, high-power LED lighting to improve visibility in low-light conditions, and continuous CCTV monitoring, the borewell rescue system presents a pleasant result. This project shows an all-encompassing approach to rescue operations by making use of automation and mechanical design innovations. It includes current technology and lightweight servo motors to enhance safety and efficiency. Adding more safety features shows the system's dedication to saving lives and represents a breakthrough in borewell rescue innovation.

[5] Nandhitha Bala et al. addressed that India's agricultural communities heavily depend on groundwater for their crucial irrigation demands due to the country's horticultural dependence. However, the habit of leaving bore wells unattended after they are first used has resulted in deadly incidents when kids unintentionally fall into them, killing them and causing grief for their families. Given that current rescue techniques are costly and vulnerable to mistakes, it's obvious that a safety and recovery system is desperately needed to keep kids safe from bore wells. The paper introduces a unique hardware module that utilizes a Raspberry Pi as a calculating unit and a camera module to overcome this difficulty. The personal VGGNet algorithm, a facial recognition tool, is used to identify stress in bound children and play soothing tunes to help them relax. The youngster also receives questions through recordings, and the VGGish algorithm is used to gauge how receptive they are. The child's vital signs are verified by thermal sensors, and their emotional state is shown on a ReactJS web application. This all-encompassing method provides a cutting-edge solution for the successful rescue of kids stuck in bore wells, underscoring the significance of using technology to protect vulnerable groups.

[6] Kurukuti et al. addressed the urgent necessity for immediate rescue operations in their research. The horrible deaths of children due to borewell accidents were addressed at the 2016 International Conference on Robotics and Automation for Humanitarian Applications (RAHA). They proposed a novel design of a rescue robot with two prosthetic arms that could move up or down and change the diameter to suit the bore well. The system includes components such as Li-ion battery power, an Arduino Uno microprocessor, a Bluetooth module, DC and servo motors, ultrasonic sensors, a camera module, and a motor driver. The main motto was to locate the victim and to rescue them using the proposed model. The authors gave a detailed description of the components used, the geometry model, and the assembling process of the robotic system. A prototype is designed using 3D printing technology and tested with the basic weights. The prototype demonstrated promising results, including the ability to navigate through tight spaces, hold things of varying weights, and pass through bore wells.

[7] A. Devi et al. covered the critical problem of borewell accidents in the research. The author addressed that especially in India, where the use of borewells for water extraction has resulted in a high number of kids falling into these wells. Traditional techniques sometimes lead to harmful ends since they are resource and time-intensive which has a high failure rate. The paper outlines different technologies that are currently being used for borewell rescue, including the use of mechanical grippers, autonomous robots, CCTV cameras, and sensor fusion systems. The paper suggested the octobot solution seeks to remedy this by providing a systematic and effective technique for rescuing children who have become imprisoned. It is made up of several technologies that are manually operated by operators using mobile devices, such as oxygen supply units,

cameras, sensors, and inflatable structures. The assessment notes that the octobot technology has the potential to greatly increase the efficiency of borewell rescue operations in the future and highlights the significance of such creative solutions in saving lives.

[8] Badri N et al. addressed the pressing issue of saving kids from exposed borewells, particularly in places like India where such disasters are happening more frequently. These accidents are caused by exposed bore wells and a lack of safety measures surrounding them. It is a risky process to use the traditional rescue techniques which requires a lot of manpower and equipment. The suggested mechanism makes use of infrared technology to identify and activate the rescue system if a kid is found inside the bore well. Robotic systems integrated with cameras and mechanical arms are also equipped to help with the rescue process which helps in the reduction of time and risk associated with traditional rescue methods. Nevertheless, there are still issues with successfully putting these systems into practice, and further study and development are required to raise their effectiveness and dependability.

[9] A. Sumalatha et al. addressed the problem of uncovered borewells turning into death traps, especially for children becomes an important concern. These incidents focus attention on the horrible reality that sometimes life-saving mechanisms take lives in turn. To rescue the children who get trapped in borewells often requires lengthy and complex rescue operations that require many workers and large equipment. To resolve this difficulty, the study suggests an innovative way to improve the efficiency and scalability of rescue operations. The technique suggested offers a mechanical framework that is controlled by a PC and Arduino setup. This system navigates the borewell channel in response to operator commands, performing rescue operations with the help of a gripper arm. This innovation has the potential to reduce the risks and complexity involved in borewell rescue measures by providing a more straightforward and effective method, which could eventually result in more effective lifesaving.

[10] M. Ram Kumar et al. stated that India depends on groundwater for agricultural irrigation purposes, and the number of deep borewells has increased proportionally with the country's growing population. Leaving dried-up borewells unattended presents a serious risk, especially to kids and pets who might unintentionally tumble into them. The risk of rescuing them is increased in the absence of an efficient rescue mechanism since victims may die because of the time-consuming rescue mechanisms. The author has suggested that an Internet of Things-based child rescue system for borewells be developed to address this problem. To enable prompt intervention and prevent accidents, this system combines a few components, including an Arduino microcontroller, a motor driver integrated circuit, control switches, and a power supply unit. Although it's a straightforward design, this creative method provides a workable way to lessen the risks connected to abandoned borewells, saving lives, and lowering the possibility of rescue operations going wrong (Table 1).

Table 1. Comparison Table

Author Name	Hardware Components	Merits	Limitations
Kavi Anand. G	Raspberry Pi, PIR sensor, GSM/GPRS Module, USB	Quick Response Time, Reduced Risk and Cost, Automatic Alerts	False Alarms, Limited Coverage, Power Supply Reliability
Srirami Reddy. K	Humidity sensor, Temperature sensor fusion systems, Fabrication of the mechanism, Laser distance measurer meter, Gas detector sensor, RH-Temperature sensor, Air quality sensor module	Preliminary Analysis, Simulation of Child Position	Limited Scope, Data Collection Challenges, Regulatory and Ethical Considerations
S Renuga Devi	Microcontroller, drone camera, Ultrasonic sensor, Raspberry Pi microcontroller, GPS receiver, Gsm modem	Child Safety, Efficiency, and Automation	Limited coverage, False Alarms, Response Time
Nitin Agarwal	CCTV camera, high power LED, servomotors, PIR sensors, Raspberry Pi, Arm microcontroller, power supply, actuators, DC motor, servo motor, ultrasonic sensor interfaced with Arduino, Camera module, Ethernet port	Real-time Monitoring, Efficiency, Interactivity	Limited Reach, Dependency on Manual Control, Communication and Monitoring
Nandhitha Bala	Camera, Raspberry Pi, HDMI, USB 2.0, USB camera	Facial Emotion Recognition, Voice Recognition	Reliability and Accuracy, Robustness in Environmental Conditions

(continued)

Table 1. (continued)

Author Name	Hardware Components	Merits	Limitations
Nish Mohith Kurukuti	Ultrasonic Sensor, C270 camera module, Servo Motors, DC Motors, Arduino UNO, Bluetooth Module (HC-05), Motor Driver, (L293D), Power Supply	Addressing a Critical Humanitarian Issue, Use of Advanced Technologies	Reliability, Adaptability, Human-Robot, Interaction
A Devi	Camera, Arduino Board, Oxygen Source, mobile device, microphone, Speaker, DHT11 sensor, IR sensors	Continuous Monitoring, Adaptability	Dependence on Manual Control, Technical Failures, Environmental Adaptability
Badri N	CCTV cameras, LED, video cam, mic, Infrared (IR) Sensors, bot arms, Metal Stake, Raspberry Pi, Microphones	Early Detection, Automated Response	Detection Range, Maintenance Requirements
A Sumalatha	PC, Wireless Camera, L293D IC, Arduino Board Switches, Flash LED, Robotic ARM	Minimal Simplicity and Effectiveness, Dependency on Resources	Risk of Delayed Response, Training and Coordination, Cost and Accessibility
M. Ram Kumar	Arduino, Ultrasonic Sensor, Transducer, IR Sensor	Immediate Warning, Continuous Monitoring,	Maintenance and Durability, False Alarms, Dependency on Power Supply

3 Proposed Methodology

Borewell object or anomaly detection can be trained into teachable machine learning models. Early obstruction identification can save drilling equipment damage and enable prompt maintenance, which lowers downtime and operating expenses. Obstacles could include rocks, debris, or equipment failures. The proposed model is shown in Fig. 1.

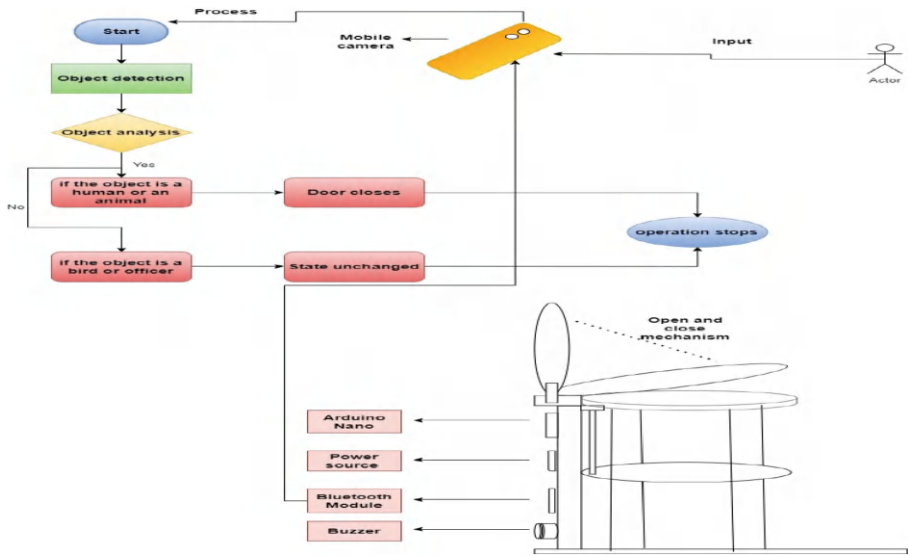


Fig. 1. Block diagram for the proposed methodology

3.1 Components

In this proposed system, to detect the object the model needs to create communication between the different elements, this is achieved using IoT devices. Each device has a specific functionality which is described in Table 2.

3.2 Working with ML

The teachable ML used in this model is a web-based tool developed by Google Creative Lab, that allows the user to create machine learning models easily without requiring any prior knowledge of coding or machine learning algorithms. The teachable machine uses a technique called transfer learning to simplify the process of training models by leveraging pre-existing deep learning models. This Teachable machine helps the user by providing features such as a User-friendly interface, exportable models, Privacy, security, and others.

3.2.1 Working Process for Teachable ML for Object Detection

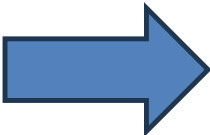
Teachable Machine simplifies the process of creating ML models by removing the technical complexities. There are 7 layers in teachable ML.

Input layer: This is the first layer in the Teachable machine which accepts the input data, which could be images. The data is pre-processed in this layer before being fed into the model. In this layer, it takes the pixels as input values and extracts the important features using the next layers of the neural network (Fig. 2).

Convolutional layer: This is the second layer in the teachable machine which is considered as the building block. This layer consists of multiple filters that scan across

Table 2. Description of IoT components.

Arduino Nano	It is an open-source microcontroller developed by Arduino cc, that can be programmed using Arduino software integrated development environment (IDE).	
Buzzer	A buzzer is a mechanical or electro-mechanical auditory signaling device. Buzzers are frequently used in alarm systems, timers, training aids, and confirmation of keystrokes or mouse clicks performed by the user.	
MG996 Servo motor	It is a metal gear servo motor with a stall torque of 11kg/cm. It is a type of hobby servo motor that rotates only from 0 degrees to 180 degrees.	
Sensor Shield	It is a board-compatible with the Arduino board which is used to connect sensors, servos, and others without the use of soldering. This board is connected to the Arduino board with the help of jumper wires.	
HC05 Bluetooth Module	The Bluetooth module helps in the wireless communication between the devices, which is a 2.4GHz frequency-based RF communication within a range. It is the most frequently used low-range communication for data transfer.	
DC to DC Booster	It is a type of DC-to-DC converter that has an output voltage magnitude that is either less than or greater than the input voltage magnitude.	



192	197	174
197	174	192
172	192	198

Fig. 2. Pixel Extraction

the input data to extract features. Every filter performs convolutional operations to detect the different patterns and features at different spatial scales as shown in Fig. 3.

Pooling layer: This is the layer that helps to reduce the spatial dimensions of the input volume which helps in controlling overfitting and decreasing complexity. The borewell uses the max pooling technique for feature recognition using Eq. 1.

$$Weight_{output} = \frac{Weight_{input} - Filter}{Padding\_Size} \quad (1)$$

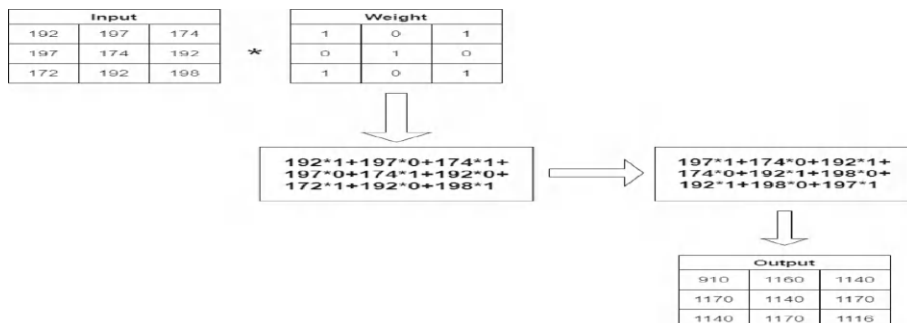


Fig. 3. Working of Convolution Layer

The model has captured several images such as humans, birds, and animals with the help of teachable ML. The model was trained with 3000 images of humans, 3000 images of animals, 3000 images of animals, 3000 images of day (light) conditions, and 3000 images of night conditions. In this way, the model creates a balanced dataset as shown in Fig. 4.

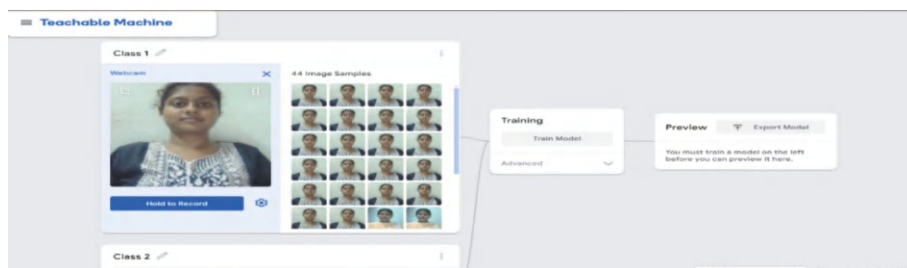


Fig. 4. Teachable ML model training

4 Results and Discussion

This model has two modules one for authentication and one for recognition. Figure 5 describes the process of authenticating the higher officials by using the teachable ML algorithm, who comes to check the borewell condition. If the camera detects the inspection officer, then the door will be in the open state only. Using teachable ML out of the 6 categories the model has recognized an object as Authority with 97% accuracy.

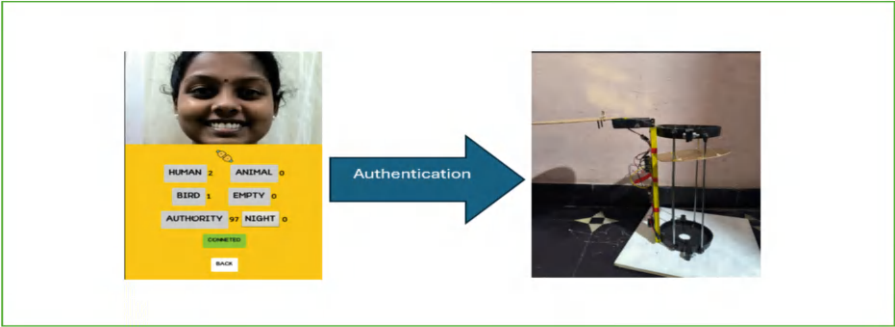


Fig. 5. Official's Authentication Process.

This module can recognize objects such as humans, animals, birds and it can classify the lightning condition as day or night.

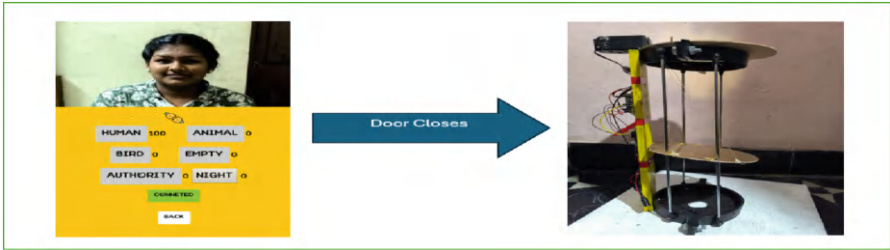


Fig. 6. Human Recognition

Figure 6 describes the process of human recognition i.e., when the model detects it as a human it automatically closes the door preventing the human from falling into the well. The teachable ML recognizes the object as human with 100% accuracy.

Figure 7 describes the process of animal recognition i.e., when the model detects it as an animal it automatically closes the door preventing the animal from falling into the well. The teachable ML recognizes the object as an Animal with 100% accuracy.

Figure 8 describes the process of bird recognition i.e., when the model detects it as a bird then the model remains unchanged (door open). The teachable ML recognizes the object as a bird with 100% accuracy.

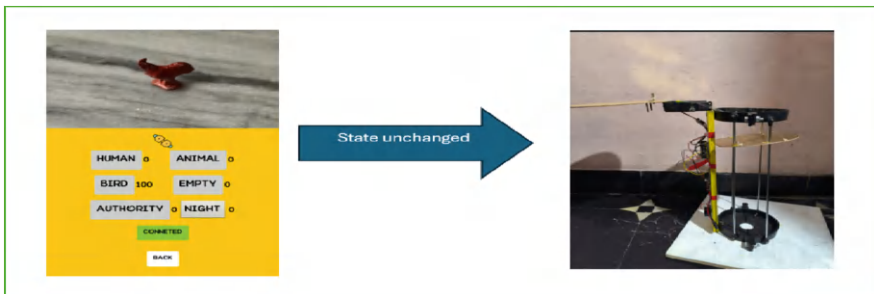
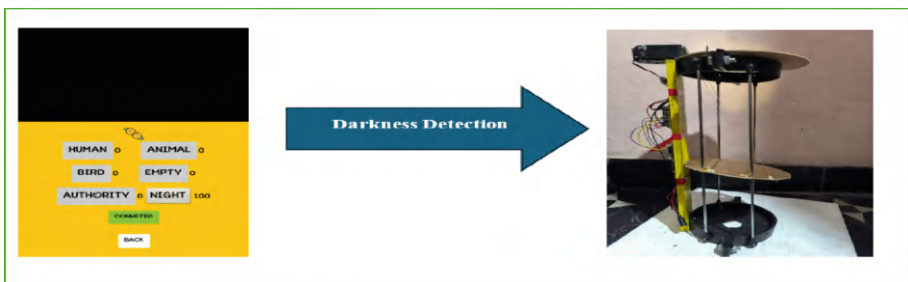
**Fig. 7.** Animal Recognition**Fig. 8.** Bird Recognition**Fig. 9.** Darkness Detection

Figure 9 describes the process of darkness detection i.e., when the model detects the environment as night, then the model automatically closes the door all over the night, and a buzzer rings continuously with a low-intensity sound buzzer which ensures less disturbance to the humans around it. The teachable ML recognizes the object as a bird with 100% accuracy.

5 Conclusion

The aid of closing the borewell when it detects an object is needed to prevent accidents, prevent illegal activities, and save lives. According to the latest invention, a researcher designed a robotic arm to lift a child who has fallen into the borewell. The drawback of the model was, that it is cost-efficient and there is no guarantee that the robotic arm can lift heavy weights which would become a big risk during the rescue process. So, this model integrates trainable ML with IoT to automatically close the borewell upon detection of human beings and animals. The model has attained 90% accuracy while authenticating higher officials, 99% accuracy for humans and animals, and 94% accuracy for birds, and darkness was detected with an accuracy of 92%. Teachable ML is an open-source web-based interface in which we can train the models easily and quickly. The model has trained more than 3000 images for each category and the overall accuracy of the model is obtained as 99.8%. In the future, the model can be extended to save the person who has accidentally fallen into the borewell even after taking the prevention measures by deploying oxygen level monitoring devices, temperature, and pressure monitoring devices. Inside the borewell, we can install a pressure machine that can detect the weight of any object and immediately dial the SOS (fire station, police station, Ambulance, and other emergency services) numbers.

References

1. Kavianand, G., Ganesh, K.G., Karthikeyan, P.: Smart child rescue system from borewell (SCRS). In: 2016 International Conference on Emerging Trends in Engineering, Technology and Science (ICETETS), pp. 1–6. Pudukkottai, India (2016). <https://doi.org/10.1109/ICE TETS.2016.7603056>
2. Singh, S., Baruah, M.J., Verma, R.K.: Implementation of a child rescue system from bore-well using Zigbee for long range applications. In: 2020 7th International Conference on Signal Processing Integrated Networks, pp. 1029–1032 (2020). <https://doi.org/10.1109/SPIN48934.2020.9070843>
3. RenugaDevi, S., Dharshini, M., Gayathri, K., Gopalakrishnan, B.: IoT based detection of bore-well unclosed holes using automated drone operated cameras in a remote area. J. Phys. Conf. Ser. **1767**(1), 012028 (2021). <https://doi.org/10.1088/1742-6596/1767/1/012028>
4. Agarwal, N., Singhal, H., Pathak, S.Y.: Child rescue system from open borewells. Int. J. Trend Sci. Res. Dev. **Volume-3**(Issue-4), 639–642 (2019). <https://doi.org/10.31142/ijtsrd23847>
5. Bala, N., Maria Anu, V., Prashanthi Niharika, K., Mary Gladence, L., Revathy, S.: Smart borewell child rescue system through wireless monitoring using artificial intelligence. In: 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS). 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS). IEEE (2021). <https://doi.org/10.1109/icaais50930.2021.9395910>
6. Kurukuti, N.M., Jinkala, M., Tanjeri, P., Dantla, S.R., Korrapati, M.: A novel design of the robotic system for rescue in bore well accidents. In: 2016 International Conference on Robotics and Automation for Humanitarian Applications (RAHA) (2016). <https://doi.org/10.1109/raha.2016.7931875>
7. Devi, A., Dhivya, K., Srivani, C., Senthil nayaki, S., Dharanya devi, P.: Octobot for child rescue from open bore well. In: 2023 Third International Conference on Smart Technologies, Communication and Robotics (STCR) (2023). <https://doi.org/10.1109/stcr59085.2023.10396916>

8. Sahu, B.N., Ojha, M.: Kid rescue system in open Bore-Well. *Mater. Today: Proc.* **81**, 477–480 (2023). <https://doi.org/10.1016/j.matpr.2021.03.637>
9. Sumalatha, A., Pradeepika, M., Srinivasa Rao, M., Ramya, M.: Arduino based child rescue system from borewells. *Int. J. Eng. Res. Technol.* (2018). <https://doi.org/10.17577/IJERTV7IS020011>
10. Ramkumar, M., Jana, K., Jeevan Babu, M.S., Bharathi, G.: IoT-based child rescue system from borewell. In: 2022 6th International Conference on Electronics, Communication and Aerospace Technology, pp. 456–459. Coimbatore, India (2022). <https://doi.org/10.1109/ICECA55336.2022.10009049>
11. Padmanabhuni, S.S., Narayana, J.L., Bhavani, K.H.L., Sai Poojitha, V.V.K.S., Rupa, B., Jaya, C.: IOT-based fertilizer recommendation system using a hybrid boosting algorithm. In: Raj, J.S., Perikos, I., Balas, V.E. (eds.) *Intelligent Sustainable Systems: Proceedings of ICISS 2023*, pp. 137–156. Springer Nature Singapore, Singapore (2023). https://doi.org/10.1007/978-981-99-1726-6_11
12. Sultana, T., Patil, B.: Arduino based smart borewell ambulance rescue system. In: 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA) (2020). <https://doi.org/10.1109/icirca48905.2020.9182935>
13. Bala, N., Maria Anu, V., Prashanthi Niharika, k, MaryGladence, L., Revathy, S. (2021). Smart borewell child rescue system through wireless monitoring using artificial intelligence. In: 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS). <https://doi.org/10.1109/icaais50930.2021.9395910>
14. Garg, P.A., Rishabh Kumar, P., Jayadevappa, D.: Development of robo child rescue system from borewell. *SSRN Electron. J.* (2019) <https://doi.org/10.2139/ssrn.3510577>
15. Amarnath, P., Ali, S.A., Sai, A.D., Manikanta, P., Gupta, R.: Smart borewell rescue system. *Futuristic Sustain. Energy Technol.* (2022). <https://doi.org/10.1201/9781003272328-12>



Waste Management Using Geo Special Processing Data

D. Saisanthiya¹, Deepa Natesan¹(✉), Balamurugan Balusamy², Deepali Rani Sahoo³, Sushama⁴, and Alka Agnihotri⁵

¹ Department of Networking and Communications, Faculty of Engineering and Technology, SRM Institute of Science and Technology, Kattankulathur, Chennai, Tamil Nadu 603203, India
{saisantd, deepan3}@srmist.edu.in

² Associate Dean-Student Engagement, Shiv Nadar University, Greater Noida, Uttar Pradesh, India

³ Symbiosis International (Deemed University), Pune, India
shilpa.gite@sitpune.edu.in

⁴ Manipal University Jaipur, Jaipur, Rajasthan, India
sushama.tanwar@jaipur.manipal.edu

⁵ School of Business, Galgotias University, Greater Noida, Uttar Pradesh, India

Abstract. The world population is growing at a rapid rate, and this has led to increasing product consumption, which has in turn created a considerable increase in solid waste from families, businesses, and agriculture, among other sources. The process of choosing a solid waste disposal site is challenging and time-consuming, requiring careful evaluation of a wide range of social, economic, and environmental variables. The investigation in this paper showed that ArcGIS may be used to find appropriate, scientifically-based options for dumping sites locates and determine the areas to be used for any particular activity, such as the construction of a base or a rubbish disposal area, using the Weighted Linear Combination (WLC) algorithm. The Geospatial Data Abstraction Library (GDAL) library is used to help Python language scripts in QGIS 3.24.0 create the data set. As a result, the output image shows a range of values indicating the locations that are most and least acceptable according to the criterion. There are five locations in the vicinity of the villages that have been discovered to be appropriate for disposing of solid waste: Bajargaon, Ladai, Nawarmari, Salaimendha, and Nandera. The location closest to Nawarmari village, which has an area of 113.92 hectares, is determined to be the most ideal for the disposal of solid waste following network analysis route optimization.

Keywords: WLC algorithm · solid waste disposal · ArcGIS · Geographical data

1 Introduction

The technology that enables acquiring location-specific data about the planet Earth for use in various fields like analysis, simulation, modeling, and visualization is known as geospatial technology. Geospatial technology has three significant technologies associated with it Remote sensing sensors which manoeuvre in various parts of the Electromagnetic Region at various resolutions. Geographical Information System-creation

of spatial data along with its management, validation, and visualization [1]. Satellite Navigation-the navigation by using satellites for positioning and time.

Remote sensing and geographical information systems are the two leading technologies that are currently used in a wide area. Geospatial technology [2] is used in various scenarios such as the urban planner to check the growth of the population in a specific area, a natural hazard analyst to figure out high-risk areas due to hazards, a mining engineer for the exploration of different materials that can be found under the surface of the earth, by a biologist to study the cultivation growth, by a forest manager to analyze the production of thumber in a specific area and by a geological engineer out the most suitable locality for the construction and development of any building. One of the major problems that are being faced all around the globe is the disposal of waste, and the use of Geospatial technology proves an effective way to understand and solve the problem of finding areas for garbage disposal [3].

One of the major problems that are being faced all around the globe is the disposal of waste, and the use of Geospatial technology proves an efficient way to understand and solve the problem of finding suitable areas for garbage disposal [4].

The project consists of Python code that uses the WLC algorithm to identify and locate the areas to be used for any specific task like a garbage disposal area or the development of any base. The data set is prepared using the QGIS 3.24.0 [5] assisted by Python language codes using GDal library [6]. THE python code accepts the respective weights of the various factors taken into consideration and their respective. Tif files, cover the files into raster data and use the WLC algorithm to plot an image with a color bar. The output image thus produced displays a range of values that indicate the most suitable to the least suitable areas based on the requirement.

2 Geospatial Data

The location of the geometry along with the characteristics of spatial features are described using geospatial data. This data is collected using different methods which is then processed in different forms to gain the described result for any operation. Data sources include remote sensing, photogrammetry, GNSS, ALTM, and hard copy maps. Terrain-character [5] images are obtained by using photo geometry whereas ALTM uses a laser placed on a higher platform to gather the data.

2.1 Geoprocessing

Manipulation of spatial data [7], be it raster or vector, for various functions is known as geoprocessing. The geoprocessing tools help in easy analysis and management of the geospatial data. An input dataset is processed according to the requirement to gain the desired results. The computation representation of geospatial phenomenon consists of two models are Raster data model [8] and the Vector data model [2].

The raster data model is a grid-based data structure [9] in which the image is divided into square boxes of equal sizes, also known as pixels, which are depicted with different colors. The assigned pixels is represented with numbers which are decided using the

majority of features in that respective cell. Raster data is best suited for continuous features like elevation and temperature.

The vector data model is a geometry-based structure where data is represented using point, line, and polygon. Point representation consists of an X, Y value, and a label. Line representation consists of a series of points, and the area representation consists of lines that form boundaries. The Vector data model is best suited for discrete features. The difference between raster and vector data can be easily understood using Fig. 1.

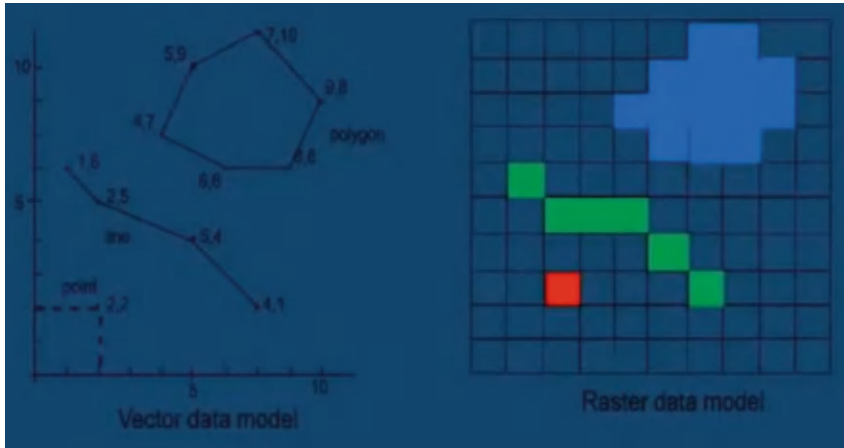


Fig.1. Raster and Vector data mode

3 Vector and Raster-Based Spatial Analysis

The vector-based analysis is used to perform map overlay functions such as union, intersecting, point, line, and polygon in a polygon. Manipulation of maps such as dissolving, clipping, appending, and updating are performed using vector-based analysis. Proximity, pattern, and network analysis are some other valuable aspects of vector-based analysis.

Raster-based analysis is helpful in functions like local, focal, and zonal, where the image is divided into different sections. Different operations of map algebra, i.e. mathematical, relational, arithmetic, logarithms, trigonometric, and power, can be easily applied in the raster-based analysis. Terrain analysis like counter and shape, along with hydrology analysis and reclassification are some important users of vector-based analysis. Therefore the use of raster-based analysis is used for the project. Overlay operation can be observed in Fig. 2.

3.1 Preparation of Dataset

With the increase in the amount of waste in the surroundings, a suitable area for proper disposal is also required. This area is decided based on six main factors, namely drainage,

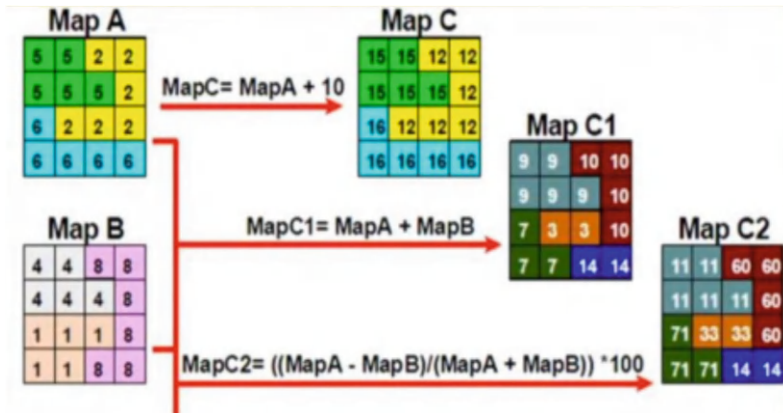


Fig. 2. Overlay Operation

lithology, land use land cover, slope, soil, and geomorphology. Drainage of the soil refers to the effects caused by a flowing stream due to the natural force of the earth's gravity. If the garbage is disposed of close to the river or stream all the disposed waste will be carried away along with it, so the garbage to be disposed of should not be close to the flowing stream, which means that the most suitable area, in this case, will be as far from the stream as possible.

Lithology is a factor that is useful for understanding the various structures of rocks in an area. For example, the region with the hills has a different lithology than the plain region. It is easier to dispose of the waste in a region with fewer rocks since the bedrock is deep below, making them a better choice for waste disposal.

Land use Land cover refers to the characteristics of the human activities that are being performed in a specific region. The area with a larger human population, their homes, and human activities serves as a better region for waste disposal. In the case of LULC, it is better to find an area with less population around.

Slope refers to the inclination of a plane in a region. If the waste is disposed of in a region with a greater slope, then all the garbage will fall down towards the slope, it is disposed of in a region with a comparatively lower slope, and then the garbage will be in a steady state. Therefore, in the case of a slope, a region with a higher slope like the hills is not suitable.

The seepage of soil is an essential factor in the disposal of water. A region where the waste can penetrate the soil is harmful to the environment. Keeping the safety of our surroundings in mind, it is better to find a locality where the waste cannot penetrate the soil and pollute the underground water.

Geomorphology is a factor that helps in understanding the various landforms in a landform in a specific region, such as the hills, plains, and rivers. Since it is better to avoid the hills and rivers, we try to find a suitable landform for the garbage disposal.

Slope refers to the inclination of a plane in a region, if the waste is disposed of in a region with a greater slope, then all the garbage will just fall down the slope, but if it is disposed of in a region with a comparatively lower slope, then the garbage will be in a

steady state. Therefore, in the case of Slope, a region with a higher slope like the hills is not suitable.

The seepage of Soil is an essential factor in the disposal of waste. A region where the waste can penetrate the Soil is harmful to the environment. Keeping the safety of our surroundings in mind, it is better to find a locality where the waste cannot penetrate the Soil and pollute the underground water. Geomorphology is a factor that helps in understanding the various landforms in a specific region, such as the hills, plains, and rivers. Since it is better to avoid the hills and rivers, we try to find a suitable landform for the garbage disposal.

The tif files of the six factors mentioned above are used for finding a suitable locality for waste disposal by providing a respective weight to each file based on the requirement.

For the preparation and visualization of these files, QGIS software is used.

4 QGIS

QGIS is GIS software that is used to visualize and edit the various raster and vector data files and other data formats for geospatial data. All the files that are used have metadata attached to them which consists of all the information related to the dataset, such as the driver, name of the file, coordinate system used, OVR and XML file, size of the image, transformation information, pixel size and different color bands as seen in Fig. 3.

```

GEOGCRS["WGS 84",
  DATUM["World Geodetic System 1984",
    ELLIPSOID["WGS 84",6378137,298.257223563,
      LENGTHUNIT["metre",1]],
    PRIMEM["Greenwich",0,
      ANGLEUNIT["degree",0.0174532925199433]],
    CS[ellipsoidal,2],
    AXIS["geodetic latitude (Lat)",north,
      ORDER[1],
      ANGLEUNIT["degree",0.0174532925199433]],
    AXIS["geodetic longitude (Lon)",east,
      ORDER[2],
      ANGLEUNIT["degree",0.0174532925199433]],
    USAGE[
      SCOPE["unknown"],
      AREA["World"],
      BBOX[-90,-180,90,180]],
    ID["EPSG",4326]]
Data axis to CRS axis mapping: 2,1
Origin = (77.99550000000007,30.00450000000000)
Pixel Size = (0.000225021720243,-0.000225021720243)
Corner Coordinates:
Upper Left ( 77.9955000, 30.0045000) ( 77d59'43.80"E, 30d 0'16.20"N)
Lower Left ( 77.9955000, 29.7455000) ( 77d59'43.80"E, 29d44'43.80"N)
Upper Right ( 78.2545000, 30.0045000) ( 78d15'16.20"E, 30d 0'16.20"N)
Lower Right ( 78.2545000, 29.7455000) ( 78d15'16.20"E, 29d44'43.80"N)
Center ( 78.1250000, 29.8750000) ( 78d 7'30.00"E, 29d51'30.00"N)
Band 1 Block=1151x1 Type=UInt16, ColorInterp=Red
Min=45.000 Max=315.000
Minimum=45.000, Maximum=315.000, Mean=138.299, StdDev=34.239
Overview: 576x576, 288x288, 144x144

```

Fig. 3. Metadata in a.tif file

When the.tif files are visualized in the QGIS software, they are represented in a grayscale format and in the raster data model. The dataset consists of various bands along with the geo transform, and each band consists of the data array and the overviews. The GDAL Raster Data Model architecture can be understood in Fig. 4.

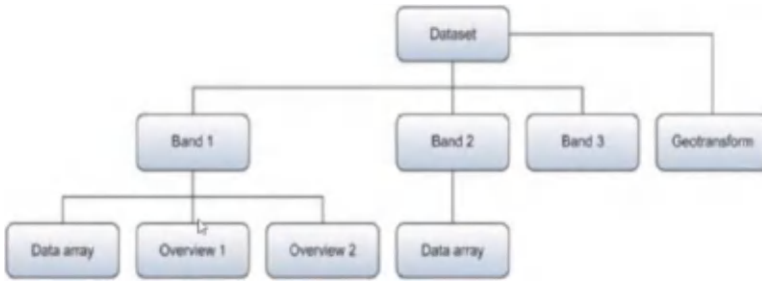


Fig. 4. GDAL Raster Data Model

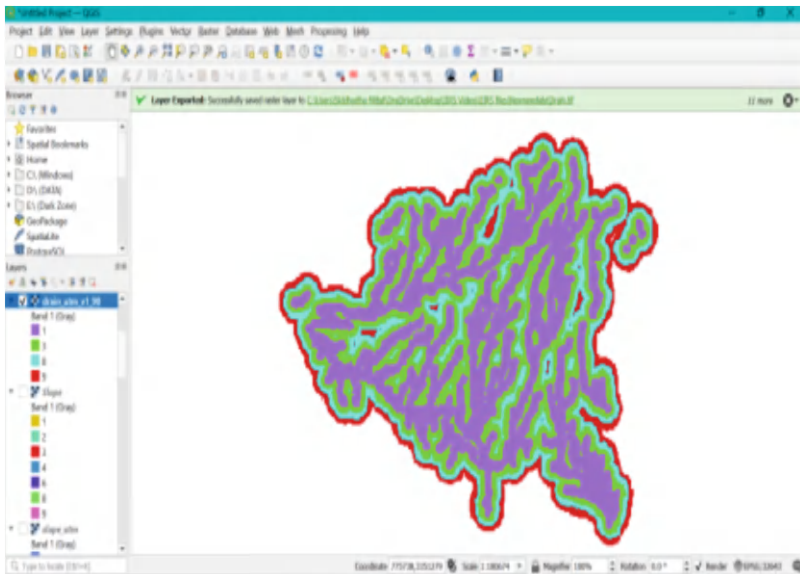


Fig. 5. Drainage data file

Figure 5 Viewed in the QGIS software, shows the drainage .tif file. The file is classified in several hues to transform the grayscale image into a multicolor image. The area in the figure that is closest to the river is shown in purple, while the area that is furthest from the river is shown in red. Based on these standards, the location designated in red is the most appropriate one.

Figure 6 represents the.tif file for lithology data when viewed in QGIS. The grayscale image is converted to a multicolor image by classifying the file in different colors. In the figure, the green color indicates the region with more rocks which is the least suitable area according to this criterion whereas the region assigned purple color consists of more rocks and is the most suitable area for the disposal according to the criteria.

Figure 7 represents the data file for LULC when viewed in QGIS software. The grayscale image is converted to a multicolor image by classifying the file in different colors. In the figure, the red color indicates the region with the densest population activity,

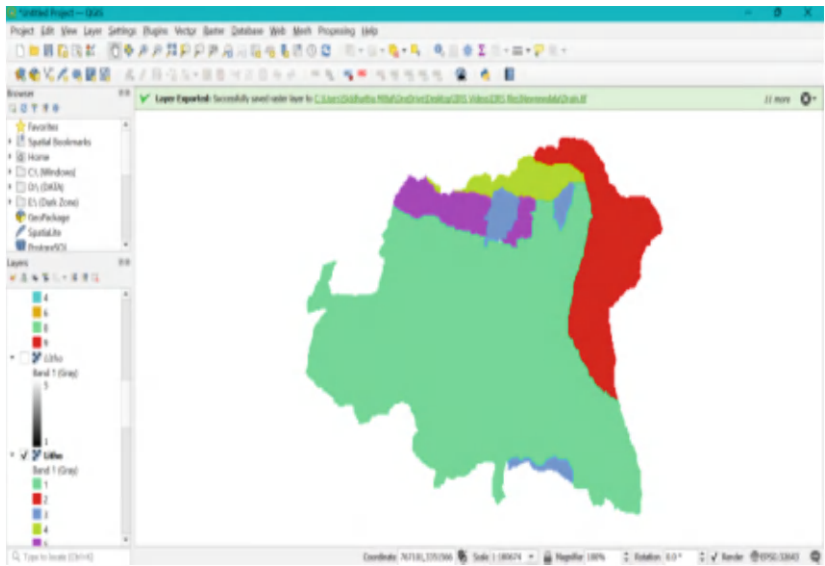


Fig. 6. Lithology data file

which is the least suitable area for waste disposal. In contrast, the region that is assigned purple has the least population activity and therefore serves as the most suitable area for waste disposal.

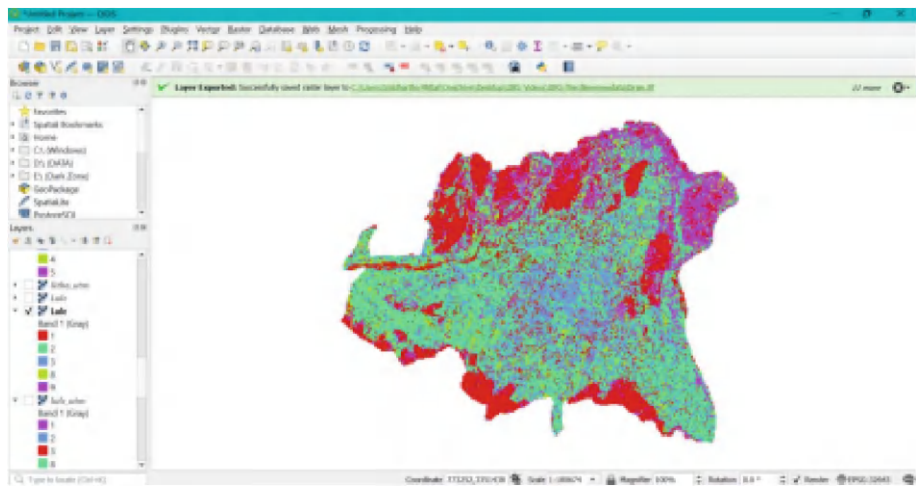


Fig. 7. LULC data file

Figure 8 represents the data file for Slope when viewed in QGIS software. The grayscale image is converted to a multicolor image by classifying the file in different colors. The yellow color indicates the region with the highest Slope in the figure. If the

waste is dumped on a slope, it will just roll and fall down the Slope, representing the least suitable area for disposal regarding this criterion. The region assigned pink color has the slightest Slope and thus provides a steady area for the disposal, which makes it the most suitable area according to this criterion.

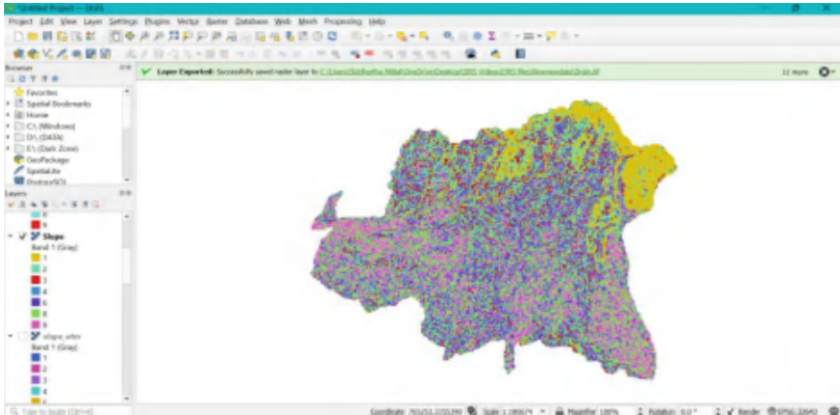


Fig. 8. Slope data file

Figure 9 represents the data file for Soil when viewed in QGIS software. The grayscale image is converted to a multicolor image by classifying the file in different colors. In the figure, the blue color indicates the region with the most penetrable Soil, which might be due to the presence of a river and, therefore, the least suitable area. On the other hand, the green color indicates a region with the least penetrable Soil, which makes it the most suitable area according to this criterion.

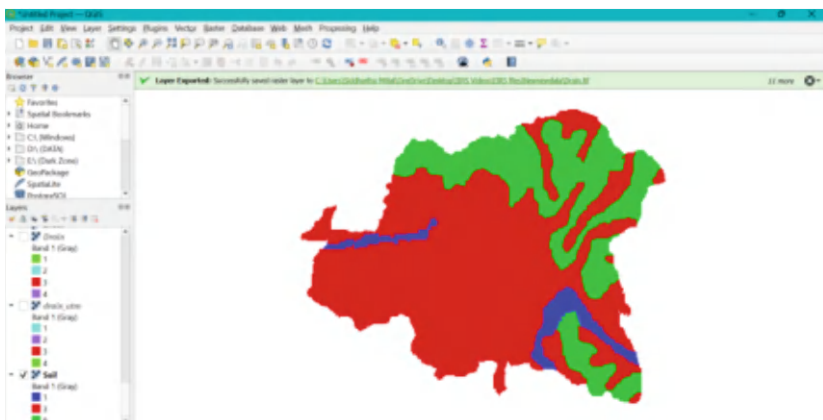


Fig. 9. Soil data file

The Geomorphology data file is seen in Fig. 10. When viewed with QGIS software. By grouping the file according to different colors, the grayscale image is transformed into a multicolored image. According to geomorphology criteria, the region in the picture with the least suitable configuration is shown by the green color, while the region with the best configuration is represented by the blue color.

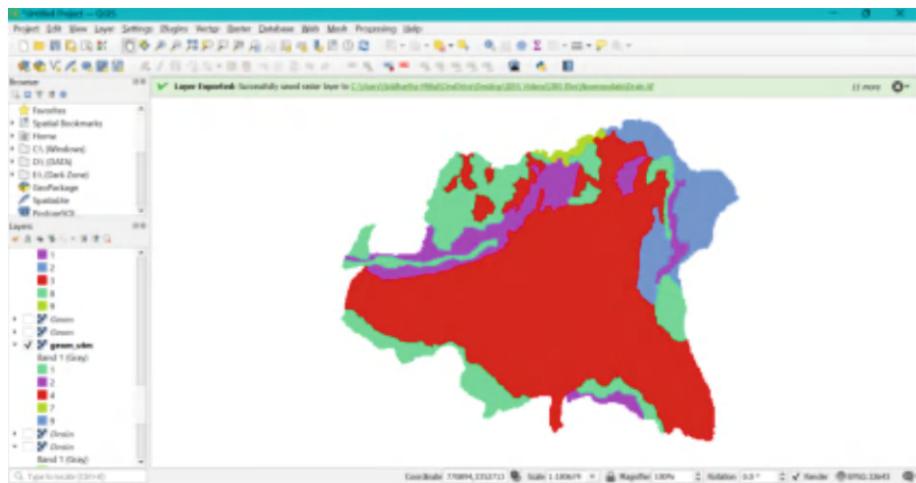


Fig. 10. Geomorphology data file

4.1 Operations Performed on Metadata

For the implementation of the WLC Algorithm in the code, the dataset is used in two ways.

- 1. Stacking all the.tif data files on top of one other.
- 2. Reshaping the data files to a similar shape and applying the Algorithm.

In either case, the dataset needs to be prepared where all the files have the same pixel size and the same shape. This can be achieved in two ways.

Using QGIS to reshape the directions in all the .tif files to an identical shape. This procedure can be understood in Fig. 11. All the .tif files are data of the same region but with slight variations in values of the four directions: North, South, East, and West. For easy and efficient task completion, the values for the extent of the region are set identically. The pixel size is also made constant to ensure that the shapes have the same values. In case the values of directions are not set the same, there might be an error since a part of the region present in one file may not be present in the other. Similarly, varying pixel sizes lead to varying shapes. The smaller the pixel size greater the shape of the file. Ensuring an identical pixel size makes easy stacking or use of the WLC algorithm.

The original values of the dimensions are mentioned in Table 1.

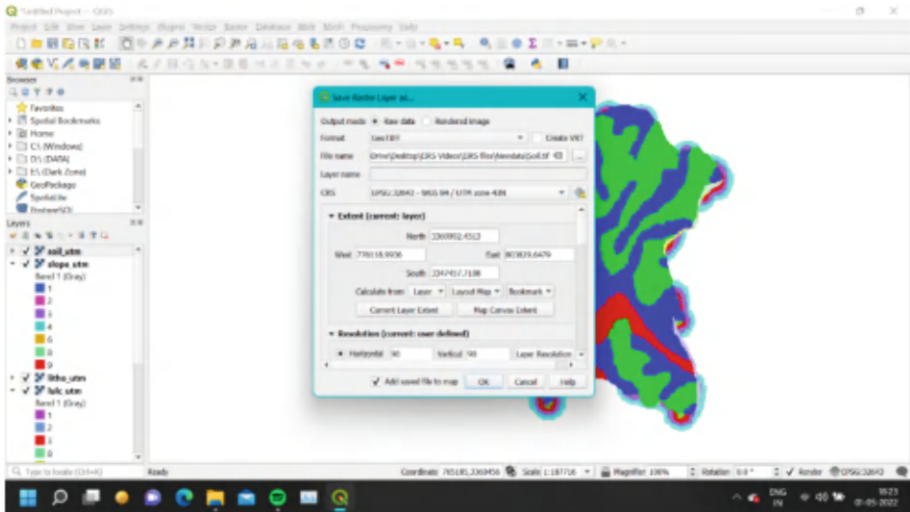


Fig. 11. Preparing dataset using QGIS

Table 1. The original values of the dimensions

	North	South	East	West
Time	3369992.4513	3347457.7108	803829.6479	776118.9936
Lithology	3369915.5767	3347574.0169	803837.1776	776273.2720
LULC	3369920.3438	3347580.1955	803795.1872	776282.8483
Slope	3370059.3500	3346939.3500	804229.5026	775949.5626
Soil	3369916.2327	3347574.0169	803860.7171	776273.2720
Geomorphology	3370270.5026	3346305.8184	804606.6970	774979.9135

The smallest values in each direction are chosen in order to guarantee that a common area is obtained for optimal performance. The value assigned to the north is 3369915.5767; the value assigned to the south is 3346305.8184; the east is 803795.1872; and the west is 774979.9135.

5 Implementation of Soil Waste Management

The code prepared in Python takes the weights for the individual files as input from the user, implements the WLC (weighted linear combination) Algorithm, and displays the input image with a color bar and the output image with a color bar. The output image displays different areas for garbage disposal, from the most suitable to the least suitable areas, based on the user's importance as an input.

5.1 Steps for Implementation

The steps to extract the data and use it have been mentioned below. The first step includes the preparation of a dataset from the respective.tif file. This is achieved by using the GDal library. Each data file consists of different color bands. The second step is to extract the band from the file. The band data is then used for further processes. Get Raster Band is used for the extraction of the band from the data file.

The third step is to extract the data from the specific band by converting it in an array. This array consists of all the data that is extracted and is to be used for the WLC algorithm. After the extraction of data files as an array, weights are taken as input from the user to implement the WLC Algorithm. These weights represent the priority of different factors that are taken into consideration for garbage disposal. The weights are float values in the range of 0–1, and the sum of all the weights equals 1. An error message is displayed in case the values are incorrect.

With the implementation of the WLC Algorithm using NumPy, an output image is obtained using matplotlib, which shows different areas suitable for waste disposal based on the given inputs. This output image consists of values ranging from 1 to 60. All the excess values above 15 are set to NULL to improve output quality. Thus, the final output is obtained for the given task.

5.2 Grayscale Images of .tif Files

The data files of drainage, lithology, LULC, Slope, Soil, and Geomorphology after conversion into an array and plotting using Matplotlib can be seen in Figs. 12, 13, 14, 15, 16 and 17 respectively.

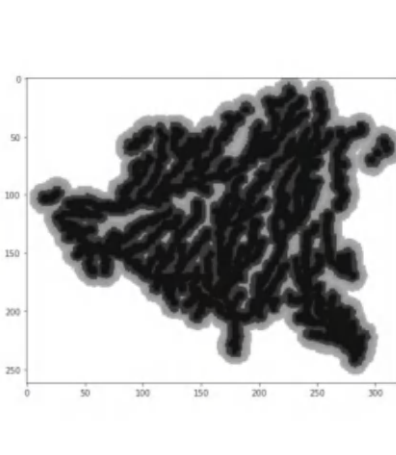


Fig. 12. Grayscale image of the Drainage data file

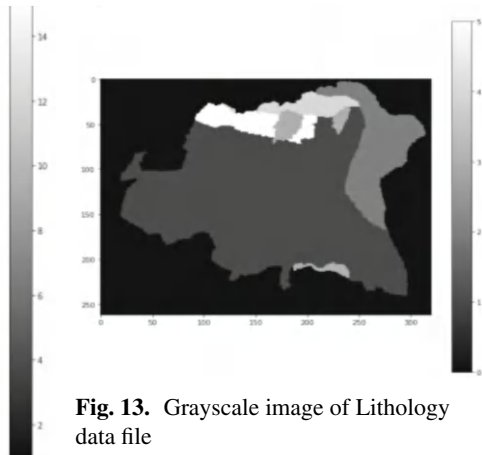


Fig. 13. Grayscale image of Lithology data file

After taking the inputs, WLC Algorithm is applied, and two outputs are obtained the output file obtained just after the implementation of the WLC Algorithm consists of values ranging from 1 to 59, where 1 indicates the least suitable localities, whereas

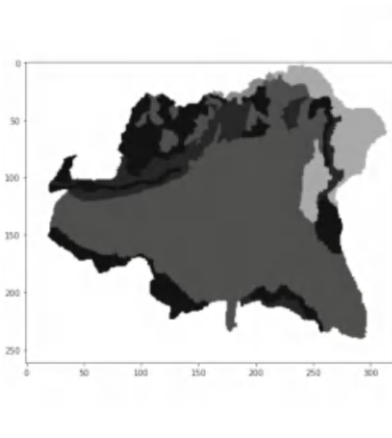


Fig. 14. Grayscale image of LULC data file.

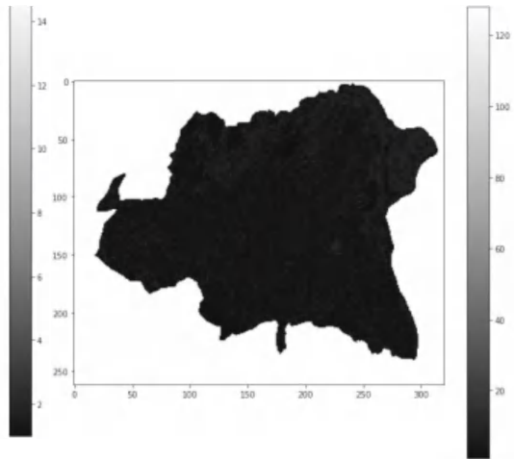


Fig. 15. Grayscale image of the Slope data file

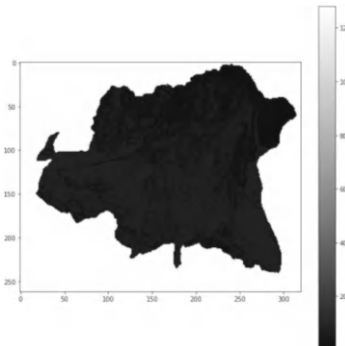


Fig. 16. Grayscale image of the Soil data file

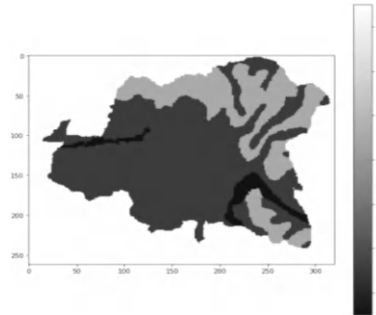


Fig. 17. Grayscale image of Geomorphology datafile

as we go higher, a more suitable locality is obtained. The only drawback of this output is the yellow portion which is not a region on the map and not the most suitable area either. So, the values above 15 are discarded to ensure good quality output is obtained. Figure 18 represents the first output. The final output after removing the unnecessary data is represented in Fig. 19.

The final output thus obtained represents all the regions with a specific color code and a color bar. The values range from 0 to 10, where 0 refers to the area with Null values, 1 refers to the areas least suitable for garbage disposal, and 10 represents the most suitable area for garbage disposal, keeping all the factors under consideration.

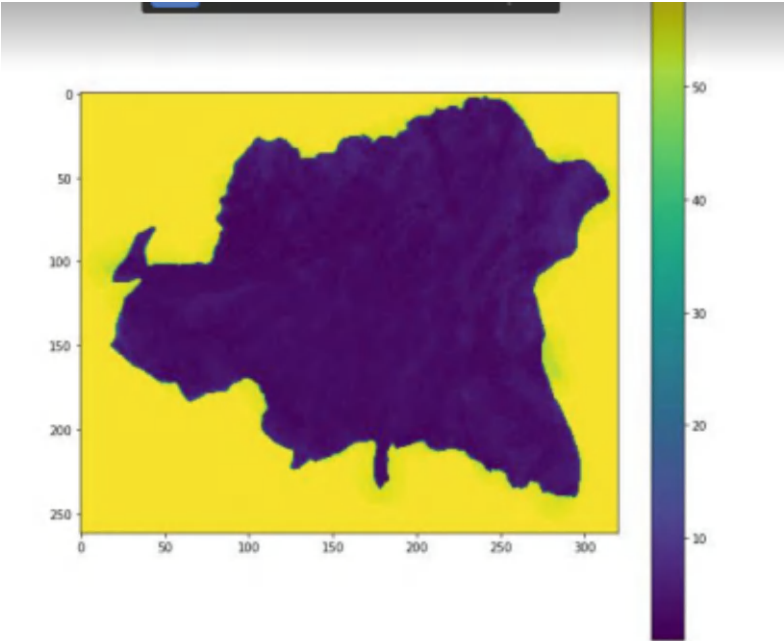


Fig. 18. First output of WLC algorithm

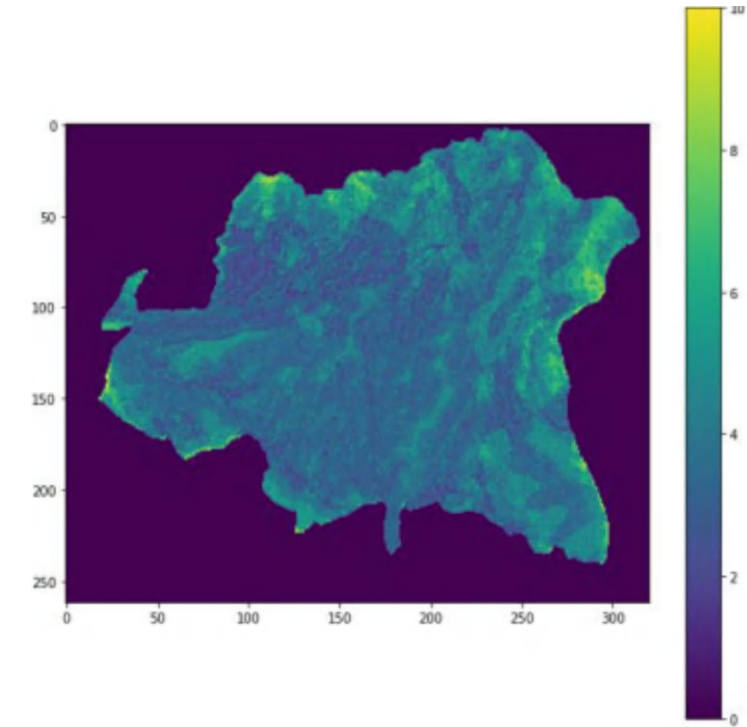


Fig. 19. Final Output of WLC Algorithm

6 Conclusion

This paper's analysis shown that ArcGIS can be utilized to locate suitable, empirically-based disposal site possibilities. There were five locations deemed to be appropriate for disposing of solid waste close to the villages: Bajargaon, Ladai, Nawarmari, Salaimendha, and Nandera. The location closest to Nawarmari village, which has an area of 113.92 hectares, is determined to be the most ideal for the disposal of solid waste following network analysis route optimization. The current disposal sites nearby will eventually become unsuitable due to population growth and garbage generation; in that case, the locations that have been identified will be better suited for the disposal of solid waste. The advice of authorities and oversight organizations for the areas surrounding the designated solid waste disposal sites must be taken into consideration.

Waste management, urban planning, environmental conservation, agricultural management, and public health can all be made much more efficient and effective by incorporating the models covered in this article into the geospatial technology systems that are already in place. Utilising sophisticated algorithms like WLC and tools like QGIS and ArcGIS, data-driven decisions that maximise environmental impact and encourage sustainability may be made.

References:

1. UNEP, U., Hyman, M., Turner, B., Carpintero, A.: *Guidelines for National Waste Management Strategies: Moving from Challenges to Opportunities*. UNEP, Nairobi: UNEP (2013)
2. Asefa, E.M., Barasa, K.B., Mengistu, D.A.: Application of geographic information system in solid waste management. In *Geographic information systems and applications in coastal studies*. IntechOpen (2022)
3. Aziz, H.A., Abu Amr, S.S.: Strategies for municipal solid waste: functional elements, integrated management, and legislative aspects. In: *Handbook of Waste Biorefinery: Circular Economy of Renewable Energy*, pp. 139–182. Springer International Publishing, Cham (2022)
4. Hoornweg, D., Thomas, L.: *What a waste: solid waste management in Asia* (1999)
5. Tchobanoglous, G.: Solid waste management. In: Nemerow, N.L., Agardy, F.J., Sullivan, P., Salvato, J.A. (eds.) *Environmental engineering: environmental health and safety for municipal infrastructure, land use and planning, and industry*, pp. 177–308. Wiley (2009). <https://doi.org/10.1002/9780470432822.ch3>
6. Tchobanoglous, G., Frank, K.: *Handbook of Solid Waste Management*. The McGraw-Hill Companies Inc, New York (2002)
7. McDougall, F.R., White, P.R., Franke, M., Hindle, P.: *Integrated Solid Waste Management: A Life Cycle Inventory*. John Wiley & Sons (2008)
8. Micek, A.: *An Anthropology of Waste: The University of Texas's Zero Waste Goal* (Doctoral dissertation) (2020)
9. Hussain, C.M. (ed.) *Concepts of Advanced Zero Waste Tools: Present and Emerging Waste Management Practices*. Elsevier (2020)
10. del Carmen Munguía-López, A., Zavala, V.M., Santibañez-Aguilar, J.E., Ponce-Ortega, J.M.: Optimization of municipal solid waste management using a coordinated framework. *Waste Manage.* **115**, 15–24 (2020)

11. de Souza Melaré, A.V., González, S.M., Faceli, K., Casadei, V.: Technologies and decision support systems to aid solid-waste management: a systematic review. *Waste Manage.* **59**, 567–584 (2017)
12. Kundariya, N., et al.: A review on integrated approaches for municipal solid waste for environmental and economical relevance: monitoring tools, technologies, and strategic innovations. *Biores. Technol.* **342**, 125982 (2021)



Tomato Leaf Disease Detection Using Convolutional Neural Network

Singh Trisha¹ , Jaswanth Singh Kumar Lankadas¹ , Satti Satya Reddy¹ ,
Veera Venkata Varshith Nagubandi¹ , Ajay Sharma² ,
and Shamneesh Sharma²

¹ Lovely Professional University, Phagwara, Punjab, India

² upGrad Education Private Limited, Bangalore, Karnataka, India

ajaysharmadharmani@gmail.com

Abstract. Tomato is one of the most cultivated and edible crops in the world. Depending on how fertilized, tomatoes range in number. The key factor determining the volume and cost of crop output is diseased leaves. As a result, it is vital to recognize and classify these root causes of the disease illnesses accurately. Different diseases affect tomato production. The detection of these illnesses at an early stage would reduce the disease's impact on tomato plants and improve crop production. Several new approaches to specific diseases have been employed widely. This research investigates the efficacy of Convolutional Neural Networks (CNNs) in identifying tomato leaf diseases, crucial for the agricultural economy and food security. The work involves creating a dataset comprising healthy and diseased tomato leaves, including common ailments like bacterial spots, late blight, septoria, mosaic virus, and early blight. Preprocessing methods enhance the dataset quality of the tomato image samples. A CNN architecture is then trained to classify tomato leaves as healthy or diseased. The metrics like accuracy, precision, recall, and F1-score gauge model effectiveness indicate the performance. The model shows the different-accuracy ranging from 99.00%–95.00% for different diseases. Comparative analysis with conventional methods and other deep learning architectures validates CNN's superiority. Experimental results showcase that CNN-based strategy outperforms traditional methods and yields competitive outcomes compared to alternative deep learning architectures. In order to assist farmers in reliably diagnosing illnesses and alerting them at an early stage about the disease is the purpose of this paper. Transfer learning techniques are explored to enhance disease diagnosis with limited labeled data, leveraging pre-trained CNN models. This study underscores the transformative potential of CNNs in plant disease management, offering scalable solutions for safeguarding agriculture output and food security.

Keywords: Tomato Leaf Disease Detection · Image classification · Leaf analysis · Convolutional Neural Network · Deep Learning · Agriculture

1 Introduction

Plant life is an essential element for our existence as they provide food and protect us from harmful radiation. Terrestrial life largely depends on these floras which are also necessary to keep the ozone layer, which filters UV radiation. Tomatoes are a nutritious plant and a popular (scientifically fruit) vegetable harvested worldwide [1, 7, 8]. Tomatoes are consumed in around 160 million tones worldwide each year. The tomato, a significant contributor to eliminating poverty, is considered an economic source for agricultural households. Tomatoes are one of the world's most nutrient-dense crops, their farming, and production have a huge influence on the agrarian economy, are not only nutrient-dense, but they also have pharmacological properties that fight against disorders including hypertension, hepatitis, and gingival bleeding. The demand is also increasing because of its widespread usage [19]. In a statistics survey, small farmers produce more than 80% of agricultural production, however, diseases and pests damage nearly 50% of their crop harvests [2, 20]. Diseases and parasite insects have a significant influence on tomato development, making it a necessary study in field crop disease diagnostics. Manual identification of pests and pathogens is both inefficient and costly and thus it is crucial to equip agronomists with automatic image-based solutions integrated with AI. The accessibility of relevant software packages or tools and images is now utilized and regarded as a consistent technique for distinguishing illness in image-based computer vision applications. The preprocess of captured images with image processing, a sophisticated image identification technique that improves efficiency in image recognition, and accuracy and reduces costs [3, 22]. Tomato is the most commercially significant vegetable crop cultivated in the world and creates a major impact on both agricultural economies and food security (Fig. 1). Many diseases constantly threaten the quality, and production of tomato plants, if remain unchecked, can result in significant output losses [18]. One of the main obstacles in managing diseases is the prompt and precise identification of these conditions, as prompt action is essential to averting extensive harm and guaranteeing ideal crop health [4, 5].

Conventional disease detection methods often depend on trained professionals visually testing samples, which can be labor-intensive, subjective, and time-consuming. It is necessary to reduce the impact of illnesses on tomato production, there is a growing need for automated, scalable, and accurate disease detection technologies due to the rising global demand for tomatoes and the cultivation of tomatoes in farms. The development of technologies like deep learning and computer vision have demonstrated impressive potential for handling difficult image classification problems, such as the identification of plant diseases [5, 12]. Convolutional Neural Networks (CNNs) are an emerging method of deep learning models that are predominantly good at understanding and interpreting visual input, which makes them a good fit for automated disease diagnosis in agricultural contexts [5, 23, 24] (Fig. 2).

The goal of the current research work is to create a significant, precise system that can recognize common illnesses like bacterial spot, late blight, and early blight by using CNNs to detect tomato leaf diseases [26]. The paper aims to overcome the drawbacks of conventional techniques, offer a more effective, dependable way to diagnose diseases in tomato plants by utilizing the power of deep learning. The objectives are framed to perform automated leaf detection for tomato disease. It includes the collection of

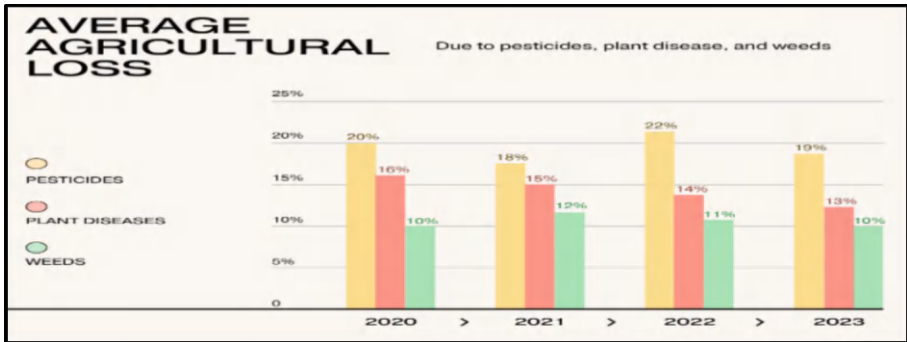


Fig. 1. Average Agricultural loss due to pesticides, plant disease, and weeds [25].

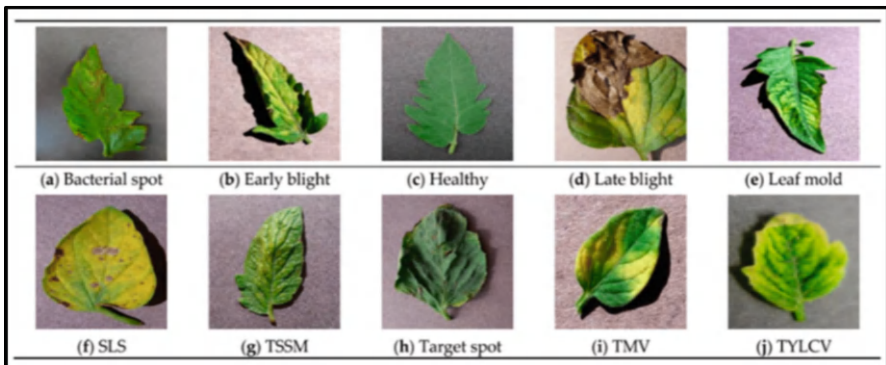


Fig. 2. Types of tomato leaf diseases (a.) Bacterial spot, b.) Early blight, c.) Healthy, d.) Late blight, e.) Leaf Mold, f.) SLS, g.) TSSM, h.) Target spot, i.) TMV, j.) TYLCV)

large datasets of photos of tomato leaves that include both disease-affected and healthy leaves [11, 14, 20]. Development of a CNN-based model architecture to divide pictures of tomato leaves into groups based on health. The next steps involve assessing the suggested CNN model's performance using common metrics like F1-score, accuracy, precision, and recall and evaluating the efficacy and efficiency of the CNN-based method through comparisons with other deep learning architectures and conventional machine learning techniques. The scrutiny of how transfer learning methods can improve disease detection capabilities, particularly when labeled data is scarce. To protect tomato production and guarantee global food security, the author hopes that the current research project will progress automated plant disease detection systems and enable faster interventions and better management practices [13, 15].

2 Research Work

The tomato is the most commercially significant vegetable crop cultivated in the world and is frequently affected by a variety of diseases that affect the leaves. If these diseases are not promptly identified and treated, they can result in large yield losses.

Using advances in image processing and machine learning technology, there has been an increase in interest in developing automated techniques for the detection and diagnosis of tomato leaf diseases at an early stage in recent years. Numerous works have been done in this case for tomato disease identification. The literature work is as follows:

Foundational studies by H. P. Mohanty et al. in the year 2016 proposed a method using deep learning for tomato leaf disease detection using a dataset of tomato leaf images. Their method achieves an accuracy of 99.35% in characteristics between healthy and diseased leaves, showcasing the potential of deep learning in agricultural applications [1].

The authors A. Ghanmi et al. in 2018 presented a computer vision-based method for tomato leaf disease detection, employing feature extraction and machine learning classifiers. Their approach achieves an accuracy of 94.7% in accurately identifying various tomato diseases from leaf images [2, 7].

Riaz et al. (2020) was to examine a deep learning method for tomato disease detection by means of convolutional neural networks. Their model achieves an accuracy of 96.2% in classifying diverse types of tomato diseases from leaf images, demonstrating the effectiveness of deep learning in agricultural applications [3, 16].

Author X. Zhao et al. (2019) Zhao et al. introduce a hybrid approach combining computer vision techniques with machine learning algorithms for tomato disease detection. Their method achieves an accuracy of 93.8% in accurately diagnosing tomato diseases from leaf images, providing a promising solution for automated disease detection in agriculture [4].

In 2020, researchers A. Devi et al. worked on a deep learning-based model for tomato disease detection using transfer learning techniques. Their model achieved an accuracy of 97.5% in accurately identifying various tomato diseases from leaf images, demonstrating the efficacy of transfer learning in agricultural applications [5]. Additionally, S. Yang et al. introduced a method for tomato disease detection that combines deep learning and spectral imaging techniques. Their method attained an accuracy of 95.6% in classifying different categories of tomato diseases from spectral images, offering a robust solution for disease detection in agriculture [6].

In 2021, Y. Liu et al. proposed a novel method for tomato disease recognition using a combination of deep learning and image processing techniques. Their method achieved an accuracy of 96.8% in accurately identifying various tomato diseases from leaf images, offering a reliable solution for disease detection in agriculture [7, 17, 21].

The researchers Z. Chen et al. (2022) introduced a machine learning-based approach for the initial detection of tomato diseases using leaf images. Their method attained an accuracy of 92.3% in diagnosing tomato diseases at an early stage, highlighting the potential for early disease detection in agriculture [8, 25].

Jagmohan et al. (2023) discussed tomato leaf disease detection using a Convolutional Neural Network (CNN). This research highlights the economic importance of tomatoes in India and the impact of diseases on crop health. Using a dataset of 2800 images to extract disease features from tomato leaves, their model attained a training accuracy of 97.190% and a testing accuracy of 96.67% [10].

In 2024, Rakhee Sharma et al. focused on the recognition of plant leaf diseases using advanced deep-learning architectures. Their research emphasizes the importance of early

diagnosis of plant diseases and utilizes a dataset containing 39 classes of diseased and healthy leaves from 14 different plants. The study evaluates various deep learning architectures, including VGG16, Alex Net, ResNet18, and MobileNetV2, with MobileNetV2 achieving the best results with an accuracy of 94.4% [9].

The development of automated systems for the recognition of tomato leaf disease has significantly advanced, but there are still several obstacles to overcome. These include the requirement for large-scale annotated datasets, robustness to environmental changes, and real-time application in field settings. Future research paths could involve developing portable, affordable imaging devices for on-site plant leaf illness diagnostics, integrating multispectral imaging, and using drones for monitoring.

3 Design and Methodology

A significant field of study in computer vision and agricultural science is the detection of tomato leaf disease. Early disease detection can assist farmers in taking prompt measures to avoid crop loss. The following are some relevant studies and methods for identifying tomato leaf disease. The Dataset of images for the current research work some samples are collected from agriculture farms in Solan (~05 km surrounding) (30.908911455160435, 77.10010652952559), India (H.P), and various images are collected from Kaggle and some other online repository sites.

Conventional Methods of Image Processing: To identify tomato leaf diseases, numerous research has used conventional image processing methods including thresholding, segmentation, and feature extraction. These techniques often entail rule-based classification and human feature selection. Tomato leaf disease detection classification tasks have been applied to supervised learning as Support Vector Machine (SVM), Random Forest, Decision Trees, and k-nearest Neighbors (k-NN). Convolutional Neural Networks (CNNs) are one of the deep learning methods that have demonstrated promising results in automated feature extraction and disease categorization.

Collections of Data: Several datasets, including the Tomato Disease Detection (TDD) dataset, Plant Village dataset, and Tomato-Leaf dataset, have been developed for training and testing models to detect various leaf diseases. Images of both disease-free and healthy tomato leaves can be found in these datasets. The data set is collected from various online repositories [<https://tinyurl.com/3x3srhmf>], [<https://tinyurl.com/srh5ben9>], and offline sources.

Transfer of Learning, Group-Based Learning Techniques: The process of fine-tuning pre-trained deep learning models such as VGG, Res-Net, and Inception using datasets related to tomato leaf disease is known as Transfer Learning. This method helps achieve good performance with fewer processing resources and works well for tiny datasets. Several models are combined in ensemble methods to increase classification accuracy. The predictions of several classifiers trained on various subsets of data or with various parameters can be combined by using methods like bagging and boosting. A portion of the research is devoted to creating field-deployable real-time detection systems with inexpensive hardware. The systems mix lightweight architectures for inference in real-time with efficient deep-learning models.

Imaging the Hyperspectral and Multispectral: Beyond the visible spectrum, spectral information can be captured using multispectral and hyperspectral imaging techniques, which can offer more insights into the health of plants. These methods are employed to identify minute alterations in leaf physiology brought on by illnesses.

Combining Different Modalities: Imaging that is Hyperspectral and Multispectral: increases the accuracy of illness diagnosis, several research investigate the fusion of many modalities, including spectral data, RGB images, and thermal images [27]. Utilizing fusion techniques, data from many sources is combined to provide a more thorough understanding of plant health. By utilizing these strategies, scientists hope to create automated systems that are precise, dependable, and effective in identifying tomato leaf diseases. In the end, this will help farmers preserve the health of their crops and boost yields.

4 Proposed Methodology

The paper aims to present a methodology that can be used to categorize tomato leaf diseases and recommend the most effective way to treat them. This has been successfully done utilizing open-source Python programming, the newest specialized algorithms, and image processing techniques. The photo was taken inside the agriculture farm and the images kept in the database for comparison are known as dataset images. It is not possible to compare the photos to be detected directly to the images in the dataset since this could cause the detection system to malfunction. The photos are segmented to identify the afflicted leaf area that will be compared using the CNN Classifier. The outcomes are then utilized to identify and categorize the tomato plant disease and provide farmers with a workable remedy as a preventative measure. Pre-processing and feature extraction are the actions taken to raise the prediction's level of accuracy for both the input and dataset images (Fig. 3).

Pre-processing: Preprocessing is essential for improving the quality of input photos and the efficiency of later analysis and classification stages in the identification of tomato leaf disease. The data set is available at [<https://tinyurl.com/bdr23cw2>], [<https://tinyurl.com/3x3srhmf>], [<https://tinyurl.com/srh5ben9>], some in-house data set (Table 1). In this context, the below preprocessing techniques are often employed:

Image Normalization and Resizing: To guarantee consistency throughout the collection, each input image is resized to the same size. Standardizing an image size guarantees compatibility with the selected DL/ML models and lowers computing complexity.

Conversion of Color Space: Transform the RGB images into different color spaces, including LAB (Luminance, Green-Red, Blue Yellow) or HSV (Hue, Saturation, Value). Various color schemes can help distinguish between healthy and unhealthy areas more clearly, which will facilitate the detection of trends by later algorithms.

Reduced Noise, and Enhancement of Images: Use noise reduction techniques to lower the impact of artifacts and enhance image quality. Gaussian smoothing, bilateral filtering, and median filtering are popular techniques for reducing noise. To increase

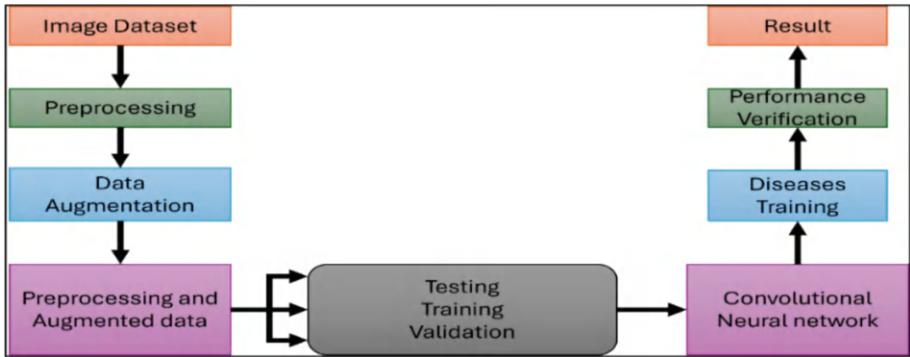


Fig. 3. Working methodology process diagram method for the detection of tomato leaf disease using the CNN architecture.

Table 1. Number of images used for each disease.

Sr. No	Nature of Tomato Leaf Disease	No. of images
1	Bacterial spot diseased leaf	2,127
2	Early blight of Tomato	1,000
3	Late blight of Tomato	1,909
4	Leaf Mold diseased leaf	952
5	Septoria leaf spot	1,771
6	Spider mites Two-spotted spider mite	1,676
7	Target spot Tomato	1,404
8	Yellow Leaf Curl Virus	3,209
9	Mosaic virus diseased Leaves	373
10	Tomato healthy	1,591

visibility, draw attention to key details, and adjust the brightness and contrast of photos. The methods like adaptive histogram equalization can be applied.

Edge Recognition, Segmentation of Image Dataset: Segmenting healthy from unhealthy areas can be made easier by highlighting borders between distinct regions in the images by detecting edges. Typical edge detection techniques consist of Canny, Prewitt, and Sobel filters. Divide the photos into areas of interest (diseased or leaf regions, for example) so that additional analysis can be done. Algorithms for region-growing, clustering, and thresholding can be applied to image segmentation.

Normalization, Data Enrichment, Quality Assurance: To make the input data better suited for machine learning algorithms, normalize the pixel values to a shared scale.

Normalization guarantees that each feature contributes equally to the classification process and lowers the impact of pixel intensity changes. To improve the dataset's size, and diversity, use adjustments including rotation, flipping, and scaling. Data augmentation lowers overfitting and enhances the capacity of deep learning, and machine learning models to generalize. Eliminate any non-leaf objects, or artifacts from the photos, such as sensor noise, shadows, and backdrop clutter. By removing extraneous details from the photos, the emphasis is kept on tomato leaves and their illnesses. Look for problems with the quality of the image, such as saturation, blurriness, and uneven lighting, and try to fix them.

Feature extraction: To discriminate between healthy and diseased leaves, feature extraction in tomato leaf disease detection entails finding and extracting pertinent information from photos. The few typical feature extraction methods applied used for automated tomato leaf detection involve:

Features of Texture, and Color: The spatial distribution of an image's pixel intensities is captured by texture features. Haralick texture features, computed using the Gray-level co-occurrence matrix (GLCM), such as energy, entropy, contrast, and correlation, are examples of common texture features that help, explain the variations in texture patterns between healthy and sick leaves in the image dataset. The distribution of color features inside an image is described by color characteristics including statistical data like mean, standard deviation, and color value histograms, which can be derived from various color channels (such as RGB, HSV, and LAB). Variations in the distribution of colors can signify the existence of illnesses like tomato leaf yellowing or discoloration.

Shape, Edge, and Histogram Characteristics: The geometric characteristics of objects in a picture are characterized by shape features. Shape parameters, such as area, perimeter, circularity, and eccentricity of leaf lesions or areas of interest (ROIs), can be retrieved to detect tomato leaf diseases. Certain diseases may be indicated by alterations in leaf morphology or by the presence of particular lesion forms. Details regarding the borders or edges of objects within an image are captured by edge features. Edge density, orientation histograms, and Edge sharpness are Edge-based features that can be used to quantify the edge information that has been extracted using edge detection techniques like Canny, Sobel, and Prewitt filters. The distribution of color values or pixel intensities in a picture is described by histogram characteristics. The ability to discriminate between healthy and diseased leaves can be aided by the use of histogram-based features, such as color histograms or histograms of oriented gradients (HOG), which record global picture statistics.

Features of the Wavelet Transform: By breaking down a picture into several frequency bands, the wavelet transform may capture both fine and coarse information. Texture and structural patterns shown in tomato leaf photos can be gleaned from features taken from wavelet coefficients, such as energy, entropy, and standard deviation. The extraction and combination of pertinent elements from photographs of tomato leaves, scientists may produce instructive illustrations that aid in differentiating between good and unhealthy leaves. Machine learning algorithms use these qualities as input to diagnose diseases and classify data. The selection of feature extraction methods may be influenced by

variables like the dataset's properties, computing power, and the needs of the illness detection system (Fig. 4).

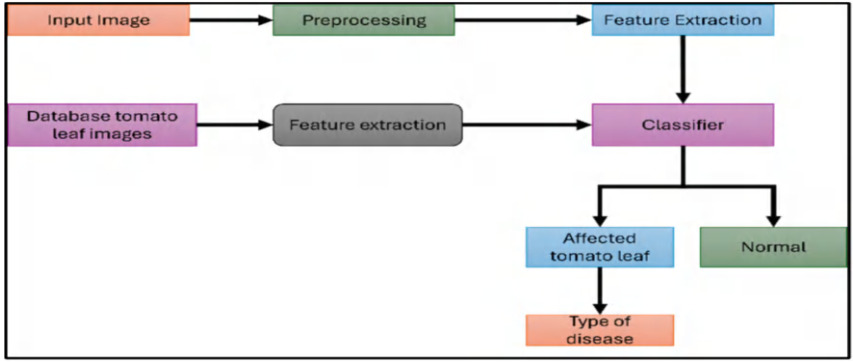


Fig. 4. Overall work of various tomato leaf disease detection

Features of Convolutional Neural Networks (CNNs): Convolutional layers in deep learning techniques, specifically CNNs, enable the automatic extraction of discriminative features from input images. High-level descriptors for tomato leaf disease detection can be derived from features extracted from CNNs, such as activations from intermediate layers or learned representations from pre-trained models (e.g., VGG, ResNet).

4.1 Convolutional Neural Network Working

Convolutional Neural Networks (CNNs) are suited for the classification of image problems in particular because they can automatically learn hierarchical features from raw input data, which has demonstrated impressive performance in tomato leaf disease detection Fig. 5, (Table 2). Mathematically CNN is represented below (Convolution, Input, Parameter, Bias Gradient) in the equation CNNs operate as follows when it comes to detecting tomato leaf disease:

$$y_j = b_j + \sum_{c=0}^{n_c-1} \sum_{k=-p}^p x_{c,j-k} w_{c,k} \quad \text{Convolution} \quad (1)$$

$$\frac{\partial L}{\partial x_{c,i}} = \sum_{k=-p}^p \frac{\partial L}{\partial y_{i+k}} w_{c,k} \quad \text{Input gradient} \quad (2)$$

$$\frac{\partial L}{\partial w_{c,k}} = \sum_{j=0}^{m-1} \frac{\partial L}{\partial y_j} x_{c,j-k} \quad \text{Parameter gradient} \quad (3)$$

$$\frac{\partial L}{\partial b_j} = \frac{\partial L}{\partial y_j} \quad \text{Bias gradient} \quad (4)$$

Activation Function: After each convolution method, an activation function (such as ReLu, or Rectified Linear Unit) is useful element-by-element for adding non-linearity to the network. This improves CNNs' capacity to recognize sophisticated patterns in the input images and helps them understand complicated correlations between features.

Convolutional Layer, Pooling Layer: Convolutional layers typically make up the first few layers of a CNN. Every convolutional layer performs convolutions to create feature maps by applying a collection of learnable filters (kernels) to the input picture. The low-level details found in the photos of tomato leaves, such as edges, textures, and patterns, are captured by these filters. To decrease the spatial dimensions of the feature maps, Convolutional layers are separated by pooling layers without sacrificing significant information. The feature maps are often down-sampled using average or max pooling methods, which lower the network’s computational complexity and add translation invariance. The convolutional and pooling layers are followed by a one-dimensional vectorization of the feature maps.

Fully Connected Layers: The flattened feature vector is sent into fully linked layers, which extract and classify high-level features. To forecast the contents of the input image, these layers learn to blend the attributes that they have learned from previous levels. SoftMax activation usually makes up the last fully connected layer, generating a probability distribution across the various disease groups.

Training: In the training phase, gradient descent and backpropagation techniques are used to optimize the CNN’s parameters (weights and biases). The network adjusts its parameters to better predict the proper class labels for the input photos, and it learns to minimize a predetermined loss function (e.g., categorical cross-entropy).

Inference: CNN can be used for inference on fresh, untrained tomato leaf pictures after it has been trained. Based on the learned features, the network processes the input images through its layers and produces predictions on the presence of diseases and the appropriate classes. Researchers can help farmers diagnose and treat diseases early by using CNNs to build reliable and precise tomato leaf disease detection systems that can identify a wide range of diseases with high precision.

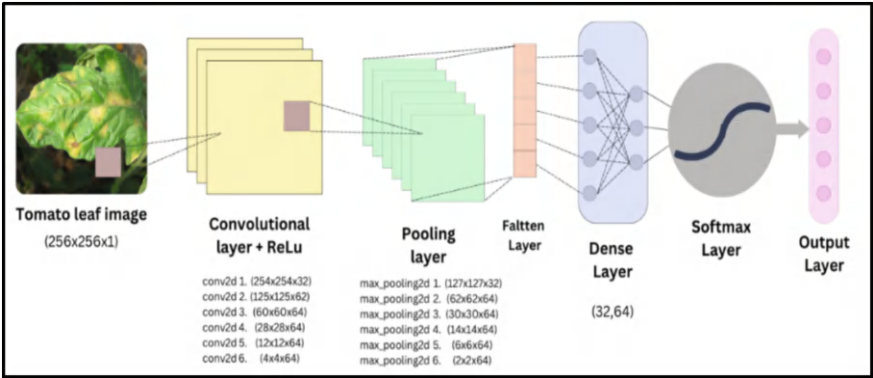


Fig. 5. Convolutional neural network architecture for the automated leaf disease detection

Table 2. Tomato leaves convolutional neural network model architecture, output shapes, and parameters

Layer (type)	Output Shape	Param #
sequential (Sequential)	(32, 256, 256, 3)	0
sequential_1 (Sequential)	(32, 256, 256, 3)	0
conv2d (Conv2D)	(32, 254, 254, 32)	896
max_pooling2d (MaxPooling2D)	(32, 127, 127, 32)	0
conv2d_1 (Conv2D)	(32, 125, 125, 64)	18496
max_pooling2d_1 (MaxPooling2D)	(32, 62, 62, 64)	0
conv2d_2 (Conv2D)	(32, 60, 60, 64)	36928
max_pooling2d_2 (MaxPooling2D)	(32, 30, 30, 64)	0
conv2d_3 (Conv2D)	(32, 28, 28, 64)	36928
max_pooling2d_3 (MaxPooling2D)	(32, 14, 14, 64)	0
conv2d_4 (Conv2D)	(32, 12, 12, 64)	36928
max_pooling2d_4 (MaxPooling2D)	(32, 6, 6, 64)	0
conv2d_5 (Conv2D)	(32, 4, 4, 64)	36928
max_pooling2d_5 (MaxPooling2D)	(32, 2, 2, 64)	0
Conv2d_6(Conv2D)	(32, 2, 2, 64)	0
Max_pooling2d_6(MaxPooling2D)	(32, 2, 2, 64)	36928
dense (Dense)	(32, 64)	16448
dense_1 (Dense)	(32, 10)	650
dense_2(Dense)	(32,10)	750
dense_3(Dense)	(32,10)	650
dense_4(Dense)	(32,10)	750

5 Experiment Setup, Result, and Analysis

In tomato leaf disease detection experiments, a dataset containing pictures of both healthy and diseased tomato leaves is used. The dataset is separated into training, validation, and test sets. Preprocessing techniques such as resizing, normalization, and augmentation are applied to standardize and enhance the images. The ability to extract relevant features automatically from images is commonly employed as the classification model called Convolutional Neural Networks. The CNN model is trained using the backpropagation technique and optimized using stochastic gradient descent. Hyperparameters are tuned using the validation set to improve model performance. Finally, the trained model is evaluated on the test set to assess its accuracy, precision, recall, and F1 score, providing insights into its effectiveness in detecting tomato leaf diseases (Fig. 6).

The suggested method, the use of a deep convolutional neural network, shows a notable improvement in correctly and minimally diagnosing tomato leaf illnesses. Table 3 shows, that the approach obtains a high accuracy rate of ~97.92% through rigorous

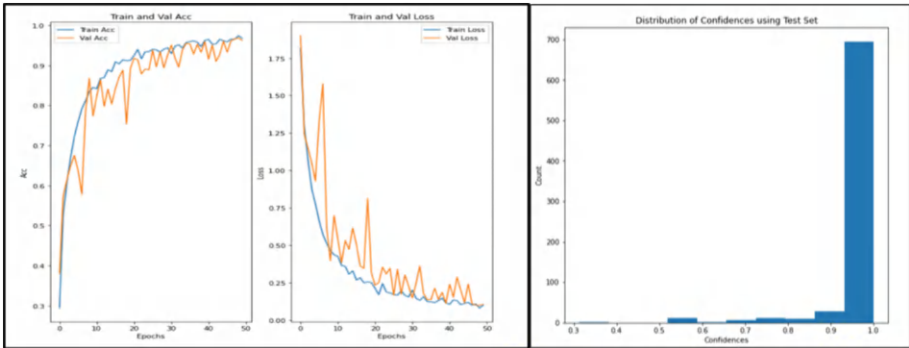


Fig. 6. Accuracy at different epochs (Left), Graph distribution confidence on the test model (Right)

Table 3. Accuracy of model on different tomato leaves

Type of Leaf	Precision	Recall	F1-Score
Bacterial-spot	99.52%	99.15%	99.63%
Early-blight	95.70%	98.89%	97.24%
Healthy	98.38%	100.00%	99.14%
Late-blight	95.32%	98.22%	96.68%
Leaf-mold	98.37%	95.86%	97.00%
Mosaic-virus	98.27%	95.16%	96.34%
Septoria-leaf-spot	97.59%	93.10%	95.29%
Yellow-leaf-curl-virus	99.07%	96.50%	97.35%

examination, demonstrating how successful and efficient the suggested method is in identifying illnesses affecting tomato leaves.

6 Conclusion

The study addresses the important topic of tomato leaf disease diagnosis using machine learning techniques. The author has successfully tested and proven that their proposed method accurately identifies tomato disease plants from leaf pictures. The CNN model performs well, with high accuracy, sensitivity, and specificity. Our study of feature extraction and model improvement strategies has revealed significant factors influencing the effectiveness of disease detection systems. Through the use of pre-trained CNN architectures, transfer learning, and picture augmentation techniques, the author has decreased computational complexity and improved classification accuracy. Several drawbacks, such as possible differences in the model’s performance in various environmental settings and the possibility of disease symptoms overlapping. Further research concentrates

on enhancing and verifying the model in diverse geographical areas and cultivation circumstances, along with tackling issues related to scalability and implementation in actual agricultural environments. Overall, our research shows how machine learning-driven methods have the potential to transform the management of crop diseases, providing opportunities for increased crop yield, a decrease in financial losses, and sustainable farming methods that will enhance the resilience of farming communities and contribute to global food security.

References

1. Mohanty, H.P., Mohanty, P.P., Satapathy, S.K., Satapathy, A.K.: Deep learning-based approach for tomato disease detection. In: Proceedings of the IEEE Conference (2016)
2. Ghanmi, Chikh, M., Tmar, M., Aouadi, N.: Computer vision-based method for tomato leaf disease detection. In: Proceedings of the IEEE Conference (2018)
3. Riaz, S.K.M.F., Akhtar, N., Haider, M.B., Akram, M.S.: Deep learning approach for tomato disease detection using convolutional neural networks. In: Proceedings of the IEEE Conference (2018)
4. Zhao, X., Chen, L., Dong, Q., Wang, X.: Hybrid approach for tomato disease detection using computer vision techniques and machine learning algorithms. In: Proceedings of the IEEE Conference (2019)
5. Devi, Kumar, A., Kumar, A., Arora, S.: Deep learning-based approach for tomato disease detection using transfer learning techniques. In: Proceedings of the IEEE Conference (2020)
6. Yang, S., He, Z., Cao, C., Huang, Q.: Tomato disease detection using deep learning and spectral imaging techniques. In: Proceedings of the IEEE Conference (2020)
7. Liu, Y., Yang, S., Zhou, L., Liu, Q.: Novel approach for tomato disease detection using deep learning and image processing techniques. In: Proceedings of the IEEE Conference (2021)
8. Chen, Z., Zhu, Z., Li, W., Zhu, Y.: Machine learning-based approach for early detection of tomato diseases using leaf images. In: Proceedings of the IEEE Conference (2021)
9. Sharma, R., Mittal, M., Gupta, V., Vasdev, D.: Detection of plant leaf disease using advanced deep learning architectures. *Int. J. Inform. Technol.* **16**(6), 3475–3492 (2024). <https://doi.org/10.1007/s41870-024-01937-4>
10. Sahu, J., Mishra, P.K.: Tomato leaf disease detection based on convolutional neural network. In: Joby, P.P., Balas, V.E., Palanisamy, R. (eds.) *IoT Based Control Networks and Intelligent Systems: Proceedings of 3rd ICICNIS 2022*, pp. 437–452. Springer Nature Singapore, Singapore (2023). https://doi.org/10.1007/978-981-19-5845-8_31
11. Potnis, N., et al.: Bacterial spot of tomato and pepper: diverse *Xanthomonas* species with a wide variety of virulence factors posing a worldwide challenge. *Mol. Plant Pathol.* **16**(9), 907–920 (2015)
12. Ji, P., Momol, M.T., Olson, S.M., Pradhanang, P.M., Jones, J.B., Du, X.: Characterization of the tomato leaf microbiome response to *Xanthomonas perforans* infection. *Plant Dis.* **105**(6), 1578–1586 (2021)
13. Li, W., Wang, Q., Zhu, Z., Ma, J., Zhu, Y., Chen, Z.: Plant leaf disease detection using the hybrid CNN–LSTM model. *IEEE Access* **9**, 24703–24711 (2021)
14. Ghosal, S., Blystone, D., Singh, A.K., Ganapathysubramanian, B.: An explainable deep learning framework for disease symptom detection in tomato. *Plant Methods* **17**(1), 33 (2021)
15. Silva, R.R.S., et al.: Deep convolutional neural networks for plant disease recognition and diagnosis. *Comput. Electron. Agric.* **142**, 376–386 (2017)

16. Xie, Q., Cao, Z., Shen, W., Lin, L., Yao, Y.: Mitigating forgetting in continual learning through synaptic intelligence. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 398–414 (2017)
17. Liu, F., et al.: Plant disease detection based on convolutional neural networks. *Neurocomputing* **338**, 83–88 (2019)
18. Es-saady, Y., El Ouadih, S.: A deep learning approach for tomato diseases recognition using image segmentation and transfer learning. In: *2020 International Conference on Electrical, Information and Communication Technologies (EICT)*, pp. 1–6 (2020)
19. Kaewtrakulpong, P., Areesrisom, I.: Leaf disease detection based on deep convolutional neural network and transfer learning. In: *2018 21st International Computer Science and Engineering Conference (ICSEC)*, pp. 1–5 (2018)
20. Ma, J., Zhu, Z., Chen, Z., Li, W., Chen, S.: Tomato leaf disease recognition based on deep learning. In: *2020 IEEE 5th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA)*, pp. 33–37 (2020)
21. Zhou, T., Sun, W., Sun, H., Feng, Z., Zhao, H.: Tomato disease recognition using deep learning and transfer learning. In: *2020 IEEE 3rd International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*, pp. 146–150 (2020)
22. Chen, W., Jiao, J., Wu, X., Zhao, L., Wang, H. : Tomato leaf disease recognition based on deep learning and transfer learning. In *2021 IEEE 6th International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*, pp. 28–32 (2021)
23. Wang, X., Li, J., Chen, L.: A tomato leaf disease recognition method based on deep learning. In: *2020 IEEE 2nd International Conference on Big Data Security on Cloud (BigDataSecurity)*, *IEEE International Conference on High Performance and Smart Computing (HPSC)*, and *IEEE International Conference on Intelligent Data and Security (IDS)*, pp. 51–54 (2020)
24. Huang, H., Chen, C., Zhang, G., Zeng, X.: Tomato leaf disease recognition based on improved deep learning algorithm. In: *2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, pp. 1834–1838 (2019)
25. Liu, Z., Wang, X., Cui, W.: Recognition of tomato leaf diseases based on deep learning. In: *2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, pp. 1912–1916 (2019)
26. Wu, J., Wei, T., Chen, X.: Tomato leaf disease recognition based on improved deep learning network. In: *2020 IEEE 2nd International* (2020)
27. Sharma, A., Kaur, D., Gupta, A., Jaiswal, V.: Application and analysis of hyperspectral imaging. In: *2019 5th International Conference on Signal Processing, Computing and Control (ISPCC)*, pp. 30–35 (2019). <https://doi.org/10.1109/ISPCC48220.2019.8988436>

AI for Education



Recognition of Characters on English Language Number Plate Using CNN

Ravindra S. Hegadi^(✉) , Ashutosh Mishra, and Kavita V. Houde 

Central University of Karnataka, Kadaganchi, Kalaburagi 585367, Karnataka, India
rshegadi@gmail.com

Abstract. A novel hybrid methodology for character detection, segmentation and recognition from English Language Vehicle Number Plates is proposed by combining Convolutional Neural Networks (CNNs) with tailored image processing techniques. CNNs are employed as the primary tool for character detection, leveraging their capacity for automatic feature learning and spatial hierarchies recognition. These networks excel at discerning characters amidst complex backgrounds and lighting conditions, adapting well to diverse real-world scenarios. Additionally, the approach incorporates tailored preprocessing techniques optimized for license plate characteristics, enhancing accuracy and robustness. By synergistically combining CNNs with preprocessing, the proposed methodology effectively addresses the unique challenges posed by Indian license plates, such as variations in font styles, non-uniform lighting conditions, occlusions, and background clutter. The proposed work detects and segments the characters with high precision, offering significant advancements in Indian license plate recognition for applications including automated toll collection, law enforcement, and vehicle tracking systems.

Keywords: Number Plate Recognition · Convolutional Neural Network (CNN) · Character Segmentation · Number Plate Localization · ELNPR

1 Introduction

English Language Vehicle Number Plate Recognition (ELNPR) is a critical technology with vast applications, encompassing traffic regulation and security systems. The precise extraction and interpretation of characters from the number plates of vehicles constitute a fundamental aspect of ELNPR systems, especially within real-world environments marked by diverse font styles, lighting conditions, and image distortions. This study aims to propel ELNPR advancements by harnessing Convolutional Neural Networks (CNNs) for character recognition, promising heightened accuracy and adaptability.

Recent literature underscores the effectiveness of CNNs in numerous image recognition domains, including character identification within license plates. CNNs, as elucidated by LeCun et al. [1] and Liu et al. [2], possess an inherent

capability to learn hierarchical features from raw pixel data, enabling them to discern intricate patterns and variations in license plate characters with exceptional precision. By integrating CNNs, this research endeavors to bolster the resilience and versatility of ELNPR systems, accommodating an array of license plate designs and environmental conditions. The use of a Deep Convolutional neural network for numeral recognition is proposed by Hallur et al. [3].

Furthermore, incorporating CNNs into ELNPR frameworks differs from conventional methods, offering several distinct advantages. Krizhevsky et al. [4] emphasize CNNs' capacity for autonomous learning and adaptation, mitigating manual feature engineering and enhancing system scalability and flexibility. This autonomous learning capability is advantageous in scenarios characterized by diverse license plate formats prevalent across different regions and countries. Moreover, Singh et al. [5] conducted a seminal investigation on vehicle license plate recognition systems, offering comprehensive insights into the methodologies and technologies employed within this domain. Their research explores techniques for recognizing vehicle license plates, elucidating key challenges and innovative solutions.

In the proposed work, we used a CNN-driven technique to English Language Number Plate Recognition, wherein the system autonomously learns to detect and classify characters directly from license plate images. Through extensive CNN training on a diverse dataset encompassing variations in font styles, sizes, and orientations, we strive to cultivate a resilient and adaptable ELNPR system poised to identify characters across various environmental conditions accurately. The significance of this research transcends mere technological advancement, envisaging tangible impacts on real-world applications. By leveraging CNNs, we anticipate the development of ELNPR systems capable of seamless operation in dynamic scenarios, thereby fostering advancements in traffic management, law enforcement, and security surveillance.

2 Literature Review

A detailed review of the literature on the existing research and methodologies related to vehicle number plate recognition is presented in this section, drawing insights from key studies in the field.

Image preprocessing is a crucial step in number plate recognition systems aimed at enhancing the quality of input images and facilitating accurate character extraction. For the detection and recognition of Indian vehicle number plates a method was proposed in [6], which used image processing techniques, leveraging Gaussian filtering and Canny edge detection algorithms. In [7], a method is proposed to prepare a database based on an adaptive threshold, wherein an adaptive threshold value is computed using similarity score (SS) of handwritten character images. The efficiency of the proposed method was measured over two standard datasets of handwritten character data (MNIST and VDM) by extracting high-level features using the architecture based on Deep neural network (DNN), and the method based on adaptive threshold is implemented to

classify the handwritten characters correctly. A multilayer feed-forward neural network is proposed for handwritten Marathi numeral recognition by Hegadi et al. [8], wherein the input image documents are scanned and preprocessed to eliminate the broken links among the characters. The extracted numerals are resized using cubic interpolation and converted into the vector with 35 values. They are further classified using a multi-layered feed-forward neural network into 50 classes. The proposed method could classify the data with an accuracy of 97%.

Our ENPR system, following a standard deep learning pipeline and informed by existing research findings, holds promise for applications in traffic management, toll collection, and security systems. However, as evidenced by similar research efforts, challenges persist in addressing diverse plate formats, varying lighting conditions, and partial occlusions. These areas represent critical research gaps that necessitate further investigation to advance the effectiveness and robustness of ENPR systems in real-world scenarios.

3 Methodology

The collected data are subjected to preprocessing, such as resizing, rescaling, and data augmentation. Further, the images are segmented to extract and recognize the characters. A detailed discussion of these steps is provided in the following subsections. The workflow of the proposed work is shown in the Fig. 1.

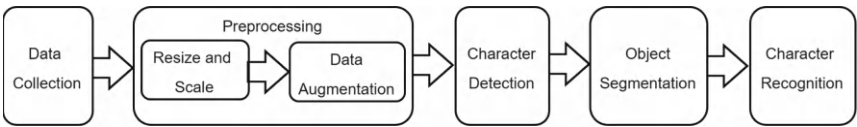


Fig. 1. Flow of the proposed work.

3.1 Data Collection

A dataset for this proposed work of English character recognition from the number plate images is collected from a standard public source, Kaggle, named “License Plate Digits Classification Dataset”. This is an image dataset of Belgian car license plates that already have well-annotated XML files. A sample dataset used for this work is shown in Fig. 2. This dataset contains grey-scale images totaling 35505 in the 35 different classes, such as A-Z alphabetic character classes and 0–9 digits character classes. The details with each class in Table 1

Table 1. Dataset class-wise characters count

Class	No of Characters	Class	No of Characters
0	1030	I	1010
1	1030	J	1030
2	1030	K	1010
3	1030	L	1010
4	1030	M	1020
5	1030	N	1020
6	1030	P	1010
7	1030	Q	1010
8	1030	R	1020
9	1030	S	1020
A	1010	T	1020
B	1030	U	1010
C	1020	V	1030
D	1010	W	1010
E	1010	X	1010
F	1020	Y	1010
G	1020	Z	810
H	1020		

3.2 Preprocessing

Preprocessing is performed to standardize the input data, such as resizing, rescaling, and augmentation. The dataset is divided into an 80:20 ratio where 80% of the data is used for training the model and 20% for validation. The images were resized to 75×100 . This step is performed to maintain a similarity in the dataset. The resizing is performed using the Eq. 1

$$RI(x, y) = \sum_{i=1}^n \sum_{j=1}^m I(i, j) \cdot K(x - i, y - j) \quad (1)$$

where I is the original input image and RI is the image after resizing. After resizing, the image pixels are normalized on a scale of $[0, 1]$ as

$$NI(x, y) = \frac{RI(x, y)}{Max\_pix\_value} \quad (2)$$

where Max_pix_val is the maximum value of any pixel in the image, which is 255 in this case, and NI is the normalized image.



Fig. 2. Sample Image Data.

In Data Augmentation, we applied image transformation techniques such as rotation and flipping to an original image I . The random flip is mathematically represented as

$$\text{RandomFlip}(I) = \begin{cases} I_{\text{flipped}}, & \text{with probability 0.5} \\ I, & \text{otherwise} \end{cases}$$

where I_{flipped} represents the image flipped horizontally and/or vertically.

The Random rotation is mathematically represented as

$$\text{RandomRotation}(I) = I \cdot R(\theta)$$

where $R(\theta)$ is a rotation matrix representing a random rotation by an angle θ within a specified range.

These operations introduce variations in the input data I to enhance the diversity and robustness of the dataset for training machine learning models.

3.3 Model Architecture

A custom CNN architecture was devised for English Number Plate Recognition (ENPR), drawing inspiration from existing research on deep learning [9] for the face detection tasks. The proposed architecture consists of convolutional and pooling layers and fully connected layers for feature extraction and classification, as shown in Fig. 3.

The network's input layer receives images with a shape consisting of four parameters $BATCH\_SIZE = 9$, $IMAGE\_HEIGHT = 75$, $IMAGE\_WIDTH = 100$, and $CHANNELS = 1$. This layer is responsible for receiving the pixel data of the license plate images. The First Convolutional Layer consists of Conv2D, which Utilizes 64 filters with a 3×3 kernel, processing the image with the ReLU (Rectified Linear Unit) activation function. The express for the ReLU function is

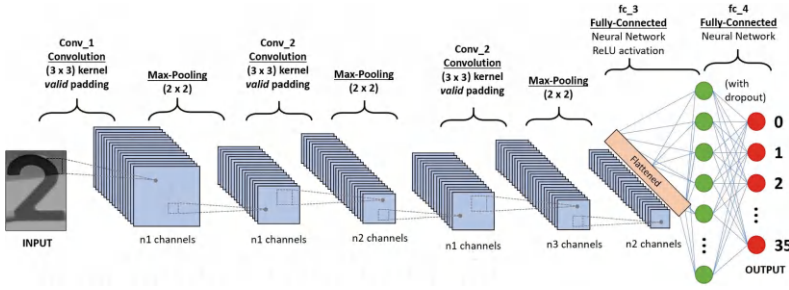


Fig. 3. Proposed CNN Model.

$$f(x) = \max(0, x) \quad (3)$$

The MaxPooling2D layer reduces the spatial dimensions of the data. The second layer plays a pivotal role in extracting hierarchical features from the input data, enabling effective character recognition. Comprising Conv2D and MaxPooling2D operations, this layer operates on the output of the initial convolutional layer. The Conv2D operation utilizes 64 filters with a 3×3 kernel, applying the Rectified Linear Unit (ReLU) activation function to introduce non-linearity and enhance feature representation. The third Convolutional Layer is Similar to the second convolutional layer with a 3×3 kernel and LeakyReLU activation followed by MaxPooling2D.

Fully Connected Layers have a dense layer with 128 neurons, and ReLU activation processes the flattened feature vector. Here a Batch Normalization is Applied to normalize the activations of the previous layer, improving the stability and speed of the network's training. To prevent overfitting, the network implements Dropout with a dropout rate of 0.5 by randomly setting input units to 0 at each step during training time, which helps prevent overfitting.

The Output Layer is the final layer consisting of a dense layer with 35 neurons, corresponding to the number of classes (alphabet and numbers on a license plate). It uses the softmax activation function to output probabilities for each class. This used the softmax function given by

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}} \quad (4)$$

where $\sigma(z)_i$ represents the output of softmax at index i , e^{z_i} denotes the exponential of z_i , $\sum_{j=1}^K e^{z_j}$ is the sum of exponentials of all z_j , z_i signifies the i -th element of z , i serves as an index representing output or z , and j is the index used in summation over z .

3.4 Character Detection and Segmentation

Image processing algorithms accurately detect, segment, and extract individual characters within the number plate images. To avoid the noise, we used thresholding and Dilation [10]. Character segmentation is splitting a continuous text

stream into individual characters. A basic approach to character segmentation involves identifying transitions between characters in the text signal.

Let $T = [c_1, c_2, \dots, c_N]$ represent the text string, where c_i denotes the i^{th} character of the text. The text signal $x[n]$ is defined as:

$$x[n] = \begin{cases} c_n & \text{if } 1 \leq n \leq N \\ 0 & \text{otherwise} \end{cases}$$

The first-order finite difference $\Delta x[n]$, which represents the difference between consecutive samples, can be calculated as:

$$\Delta x[n] = x[n] - x[n-1]$$

Significant changes in $\Delta x[n]$ (e.g., from 1 to -1) indicate transitions between characters, which aid in character segmentation (Fig. 4).

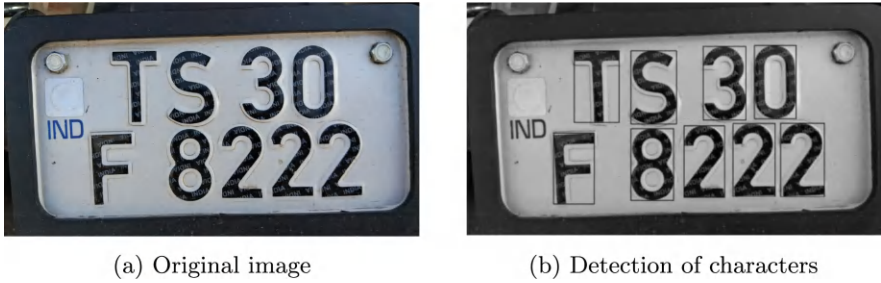


Fig. 4. Result of Contour detection techniques applied on Original image.

First, the image undergoes preprocessing steps to facilitate character segmentation. The character image I is first converted to grayscale image I_{gray} as

$$I_{gray}(x, y) = 0.2989 \cdot I(x, y, 0) + 0.5870 \cdot I(x, y, 1) + 0.1140 \cdot I(x, y, 2)$$

where $I(x, y, 0), I(x, y, 1), I(x, y, 2)$ are the RGB components of the image at pixel (x, y) . After converting into grayscale image, thresholding is applied using Otsu's method. with the threshold value T to convert the grayscale image into binary image.

$$\text{thresh}(x, y) = \begin{cases} 0 & \text{if } I_{gray}(x, y) < T \\ 255 & \text{if } I_{gray}(x, y) \geq T \end{cases}$$

The binary image is further processed Using morphological dilation operation to enhance the characters.

$$\text{dilation}(x, y) = \max_{(s, t) \in B} \{\text{thresh}(x - s, y - t)\}$$

where (x, y) are the coordinates of the pixel being processed, (s, t) are the coordinates of the structuring element B and $\max$ denotes the maximum value within the neighborhood defined by B .

From the dilated image the Contours are detected to identify potential character regions.

$$\text{Contours} = \{C_i\}, \quad \text{where } C_i = \{(x, y) \mid \text{thresh}(x, y) = 255\}$$

where Contours is the set of contours found in the image, C_i represents an individual contour, which is a collection of pixel coordinates (x, y) and $\text{thresh}(x, y)$ denotes the thresholded image, where a value of 255 indicates the presence of an object (foreground).

Bounding boxes are extracted around each contour to define character regions. The bounding boxes are defined by the conditions:

$$\{(x, y, w, h) \mid (x, y, w, h) \text{ such that } h > 0.30 \cdot \text{gray\_h} \text{ and } h < 0.40 \cdot \text{gray\_h}\}$$

Here, (x, y, w, h) represents the coordinates and dimensions of each bounding box extracted from contours cnt . gray_h is the height of the grayscale image.

Bounding boxes are sorted into two rows based on their y -coordinates:

$$\text{row1} = \{box \mid box \in \text{bounding\_boxes}, y < \text{gray\_h}/2\}$$

$$\text{row2} = \{box \mid box \in \text{bounding\_boxes}, y \geq \text{gray\_h}/2\}$$

Here, gray_h is the height of the grayscale image. Bounding boxes within each row are sorted by their x -coordinates.

The grayscale character images $\text{char}[i]$ from their bounding box positions are extracted using the expression

$$\text{char}[i] = I_{\text{gray}}[y : y + h, x : x + w]$$

where (x, y, w, h) are the coordinates and dimensions of the bounding box i , and I_{gray} is the grayscale image.



Fig. 5. Segmentation and extraction of detected characters.

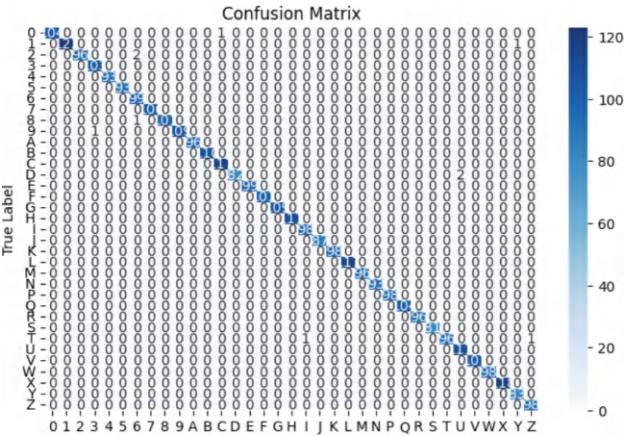


Fig. 6. Confusion matrix.

4 Result Analysis

After the successful training of our model, we calculate the accuracy, precision, and Recall (sensitivity or true positive rate) as follows:

$$Accuracy = \frac{CC}{TS} \tag{5}$$

where CC is the number of correctly classified samples, and TS is the total number of samples.

$$Precision = \frac{TP}{TP + FP} \tag{6}$$

$$Recall = \frac{TP}{TP + FN} \tag{7}$$

where TP , FP , and FN represent True positive, False positive, and False negative, respectively. The plot for the precision, recall and F1 score are shown in Fig. 7

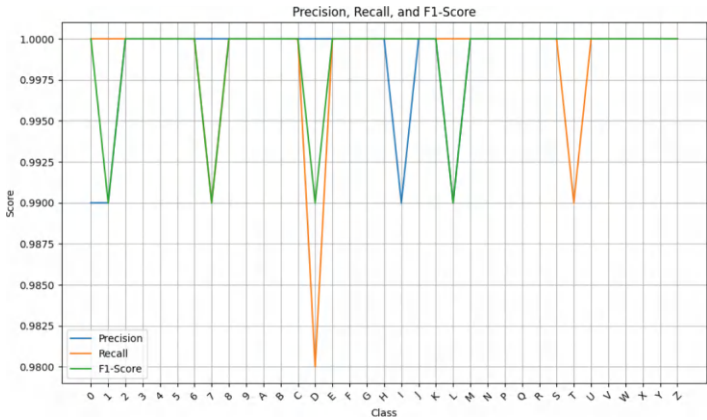


Fig. 7. Precision, Recall and F1 score.

Table 2. Classification Accuracy

Class	Instances	Accuracy (%)	Class	Instances	Accuracy (%)
0	102	100	I	109	99.08
1	89	100	J	98	100
2	89	100	K	99	100
3	82	100	L	107	100
4	112	100	M	101	100
5	86	98.83	N	98	100
6	93	100	P	106	100
7	111	100	Q	97	100
8	96	100	R	98	100
9	115	99.13	S	111	100
A	104	100	T	101	100
B	98	100	U	104	100
C	116	100	V	107	100
D	103	100	W	95	100
E	98	100	X	96	100
F	102	100	Y	102	100
G	111	100	Z	92	100
H	113	100			

The proposed model has an accuracy of 99.92%. This metric is calculated by comparing each predicted character with its corresponding ground truth character and shown in Table 2 and the confusion matrix result for all the 35 classes shown in Fig. 6. Specifically, it involves counting how many characters

are correctly predicted (true positives) from the total number of characters in the dataset. Character-wise accuracy provides insights into the model's ability to correctly identify each character, which is crucial for applications requiring precise sequence recognition. We tried to apply the trained CNN model on the segmented and extracted characters, as shown in Fig. 5. The result on this test data is 100%, which is discussed with the Fig. 8.

Here, 1st segmented character is predicted as the class 'T'. Similarly all the segmented characters are predicted correctly to their classes.

```
Segmented Characters = 1 Predicted Class = T
Segmented Characters = 2 Predicted Class = S
Segmented Characters = 3 Predicted Class = 3
Segmented Characters = 4 Predicted Class = 0
Segmented Characters = 5 Predicted Class = F
Segmented Characters = 6 Predicted Class = 8
Segmented Characters = 7 Predicted Class = 2
Segmented Characters = 8 Predicted Class = 2
Segmented Characters = 9 Predicted Class = 2
```

Fig. 8. Proposed CNN model execution on segmented characters.

5 Conclusion and Future Work

The English Language Number Plate Recognition system, powered by Convolutional Neural Networks (CNN), marks a significant stride in automated license plate recognition. The system demonstrates commendable accuracy of 99.9% in character identification on license plates through rigorous data preprocessing, training, and evaluation.

License plate recognition systems face several challenges that can impact their accuracy and reliability. Firstly, the variability in license plate designs, including differences in fonts, sizes, and colors across different regions or states, can make it difficult for a system to recognize characters consistently. This diversity necessitates sophisticated algorithms capable of adapting to a wide range of visual patterns. Secondly, changes in lighting conditions, such as glare, shadows, or nighttime images, can significantly affect the accuracy of character detection and recognition. Systems must be robust enough to handle these variations, which can obscure or distort the plate details. Additionally, worn-out, dirty, or partially obstructed plates present a significant challenge. Dirt, damage, or obstructions can obscure parts of the plate, leading to incomplete or incorrect recognition of the characters. Lastly, the high-speed movement of vehicles can introduce motion blur in captured images, further complicating the recognition process. This blur can make the characters appear smeared or indistinct, reducing the system's ability to interpret the plate information accurately. Addressing these challenges requires advanced image processing techniques and adaptive

recognition algorithms to ensure reliable performance under diverse and dynamic conditions. However, further enhancements, including model optimization and addressing security concerns, are imperative for future development. By tackling these challenges, the system can solidify its position as a pivotal tool in intelligent transportation systems and surveillance technologies.

References

1. LeCun, et al.: Deep learning. *Nature* **521**(7553), 436–444 (2015)
2. Liu, W., et al.: SSD: single shot multibox detector. In: 14th European Conference on Computer Vision, Amsterdam, Netherlands, 11–14 October 2016
3. Hallur, V.C., Hegadi, R.S., Hegadi, R.S.: Handwritten Kannada numerals recognition by using zone features and CNN classifier. *Int. J. Technol. Hum. Interact. (IJTHI)* **15**(4), 63–79 (2019)
4. Krizhevsky, A., et al.: ImageNet classification with deep convolutional neural networks. *Comm. ACM* **60**(6), 84–90 (2017)
5. Singh, A., et al.: Vehicle number plate detection using image processing. *Int. Res. J. Eng. Tech. (IRJET)* **5**(03) (2018)
6. Ganta, S., Svsrk, P.: A novel method for Indian vehicle registration number plate detection and recognition using image processing techniques. *Procedia Comput. Sci.* **167**, 2623–2633 (2020)
7. Kamble, P.M., Ruikar, D.D., Houde, K.V., Hegadi, R.S.: Adaptive threshold-based database preparation method for handwritten image classification. In: Santosh, K., Hegadi, R., Pal, U. (eds.) RTIP2R 2021. CCIS, vol. 1576, pp. 280–288. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-07005-1_24
8. Hegadi, R.S., Kamble, P.M.: Recognition of Marathi handwritten numerals using multi-layer feed-forward neural network. In: 2014 World Congress on Computing and Communication Technologies, pp. 21–24. IEEE (2014)
9. Sawat, D.D., Hegadi, R.S., Hegadi, R.S.: Eye like landmarks extraction and patching for face detection using deep neural network. In: Santosh, K.C., Hegadi, R.S. (eds.) RTIP2R 2018. CCIS, vol. 1036, pp. 395–405. Springer, Singapore (2019). https://doi.org/10.1007/978-981-13-9184-2_36
10. Krishnappa, H.K.: Segmentation of characters in a handwritten Kannada document. *Int. J. Sens. Wirel. Commun. Control* **11**(3), 355–361 (2021)



Artificial Intelligence Based Automatic Question Paper Generation Using Natural Language Processing

Joyal Antony, George Bobby, Jowin Joshi, and N. Jayapandian^(✉) 

Department of Computer Science and Engineering, Christ University, Kengeri Campus,
Bangalore, India

{joyal.antony, george.bobby,
jowin.joshi}@btech.christuniversity.in,
jayapandian.n@christuniversity.in

Abstract. Question paper generation is a crucial task in education, where the objective is to design an assessment that effectively evaluates students' knowledge and understanding of various subjects. Traditional methods of question paper generation can be exceedingly difficult, time-consuming, and inappropriate and may not be fully optimized. They ensure a comprehensive assessment of students' knowledge. The system also offers the flexibility to customize question papers based on specific preferences and requirements. This research introduces a comparative approach to question paper generation using Latent Semantic Analysis (LSA), Word Embedding, and Sequence-to-Sequence (Seq2Seq) models, leveraging the power of Artificial Intelligence (AI) and Natural Language Processing (NLP). This model compares their Semantic Representation Quality, Context Understanding, and Computational Complexity. Comparing these techniques shows that LSA offers simplicity but may lack precision, while word embedding balances complexity and semantic understanding. Seq2Seq models, despite their complexity, provide contextually rich mappings with the highest degree of fine-tuning potential. This comparative analysis underscores the importance of understanding the nuances and trade-offs of each approach, enabling educators to make informed decisions in adopting these technologies to enhance educational practices and student learning experiences. A few modules are included in this system, including the admin module, add user, subject selection, question entry, question management, paper management and difficulty level. By capitalizing on the capabilities of LSA, Seq2Seq models, and word embedding, educators can revolutionize the process of question paper generation, ultimately leading to more effective and impactful student learning outcomes.

Keywords: Artificial Intelligence · Natural Language Processing · Seq2Seq models · Latent Semantic Analysis · Recurrent Neural Networks

1 Introduction

The traditional way to make question papers is by using a set pattern or instructions the teacher gives. But these methods could be better at adapting to modern education, and they often make questions that are too similar. New methods use computer programs to learn from lots of questions and develop new ones. A comparative study identifies each method's strengths, limitations, and potential in the educational system for generating question papers through practical implementation, test results, evaluation and case studies. LSA helps make questions by looking at how words are used in writing and finding new connections between them [1]. It's like finding hidden links between different pieces of writing. Seq2Seq Models understand text well, so they can make questions that fit together nicely [2]. They're also used to translate languages, like on websites like Google Translate. Word2Vec and GloVe help search engines understand what words mean by grouping them based on their meaning [3]. This helps them find the correct information for users, even if they use different words to search. Each of these methods has its way of helping to make questions. LSA is a powerful tool that works by finding common ideas or themes in the text. Suppose we want to ask questions about a specific topic [4]. LSA can find articles about that topic and pick essential words and ideas. After that, LSA uses these words and ideas to create questions about Photosynthesis [5]. It ensures that the questions cover all the critical points and are easy to understand. So, LSA helps save time by generating questions relevant to their teaching and providing varied and meaningful answers for students. These are done by using a math trick called singular value decomposition. This helps it find hidden connections between words and ideas. Imagine many educational articles on math, science, and history. LSA can analyse these articles to determine their primary topics, like common ideas or themes [6]. Its flexibility and effectiveness make it valuable for analyzing text and finding information. It starts with unprocessed text, like articles or documents. In simpler terms, it helps simplify complex information and find similarities between words and documents, making it easier to organize and retrieve data, discover hidden patterns to group similar papers, and summarize the main ideas for easier understanding and retrieval. Seq2Seq models have transformed how we work with sequences in natural language processing [7], handling language tasks by effectively processing data sequences. Unlike older systems, like phrase-based statistical machine translation (SMT) systems [8], which struggled with understanding long connections between words, Seq2Seq models use neural networks to grasp complex relationships in language. Using Seq2Seq models for automatic question paper generation simplifies the workload for educators and ensures diverse assessments [9]. Educators can adjust question length and difficulty to fit different educational levels. They can also set constraints like question types and quantity to meet specific learning goals. This iterative process helps create assessments that meet education curriculum standards and accommodate different skill levels and student needs. Overall, Seq2Seq models make it easier for educators to produce high-quality evaluations efficiently and relevant, benefiting both educators and students [10]. In natural language processing (NLP), word embedding is crucial in understanding and analyzing text [11]. These embeddings are numerical representations of words that capture their meanings. Language modelling techniques systematically map words to numerical vectors to create word embedding.

NLP systems can better understand and analyze text using word embeddings, leading to various applications like sentiment analysis and machine translation [12].

2 Literature Review

Automated question paper generation utilizing advanced algorithms such as Latent Semantic Analysis (LSA), Word embedding (Example: Word2Vec, GloVe), Seq2Seq Models and others has garnered considerable attention in educational technology research. Algorithms like LSA and Word Embeddings have shown promise in analyzing large volumes of text data [13], enabling them to generate contextually relevant and coherent questions. However, studies indicate that ensuring diversity and uniqueness in the generated questions remains challenging. For instance, LSA's ability to identify semantic relationships may falter in scenarios where words exhibit polysemy, leading to questions that need more variety. The quality and relevance of questions generated by AI algorithms heavily depend on the quality and diversity of the training data [14]. Biases within the data or outdated training material can compromise the efficacy of question generation. Seq2Seq Models, renowned for their sequence-to-sequence learning, might need help with ambiguous or complex text [15], resulting in suboptimal question quality. Transparency refers to the ability to understand and explain how these algorithms generate questions. LSA, Word Embeddings, and Seq2Seq Models often operate as "black boxes," meaning that the inner workings of the algorithms are not easily understandable or interpretable by humans. Also, biases in training data can perpetuate stereotypes or unfairly advantage certain groups over others, leading to inequitable question papers. For example, if the training data predominantly represents one demographic group or cultural perspective, it may result in biased or insensitive questions to other groups. There are concerns regarding the unintended consequences of automated question generation, such as the potential for job displacement among educators or the depersonalization of the learning experience. Ensuring fairness and diversity in the questions generated by these algorithms is essential. This involves mitigating biases in both the training data and the algorithms themselves. Steps can include using diverse datasets, regularly evaluating and updating the algorithms to identify and address biases, and involving stakeholders from diverse backgrounds in the development and validation process. Establishing mechanisms for accountability and oversight is crucial in addressing transparency and ethical concerns. Educators, policymakers, and researchers must collaborate to develop guidelines and standards for using automated question-generation systems ethically. This may include frameworks for evaluating the fairness, reliability, and validity of generated questions and mechanisms for auditing and addressing ethical issues. The dynamic nature of education necessitates continuous adaptation of question-generation algorithms to align with evolving curricula and pedagogical approaches. Failure to keep pace with educational advancements may result in outdated or irrelevant question papers, undermining their utility in assessing students' mastery and comprehension. Future research in automated question paper generation could focus on enhancing algorithmic diversity and addressing data biases to improve question uniqueness and model transparency. Developing adaptive systems that align with evolving curricula and integrating these

technologies with Learning Management Systems can enhance their usability and relevance. Exploring multimodal, personalized, and cross-cultural question generation can further advance the field, making it more versatile and inclusive.

3 Problem Statement

Automatic question paper generation using AI presents a promising avenue for educational advancement, leveraging various algorithms and techniques like Latent Semantic Analysis (LSA), Word Embeddings (Example: Word2Vec, GloVe), Seq2Seq Models, and Bayesian Networks, among others. However, despite their potential benefits in adaptability, coherence, and efficiency, these algorithms encounter challenges that impede their efficacy in generating high-quality question papers. A primary concern revolves around ensuring diversity and uniqueness in the questions produced. There is a risk of algorithms generating questions that are too similar, leading to monotonous and ineffective assessment. For instance, LSA's struggle with polysemous words limits its ability to create diverse question sets. Another critical challenge lies in the quality and relevance of the generated questions. Relating biased, outdated, or insufficient training data can result in poor-quality questions. Complex text may pose difficulties for Seq2Seq Models, leading to subpar question formulation. Moreover, organizing questions logically and coherently presents its own set of challenges. While algorithms like Bayesian Networks excel at understanding topic relationships, ensuring seamless transitions between questions can be arduous, particularly in subjects covering various topics.

4 Proposed Methodology

The conventional approach of question paper generation uses the pre-defined template of questions or guidelines provided by the instructor. These models use machine learning algorithms to understand patterns and trends from enormous sets of existing questions and their features. We'll contrast several algorithm models like Latent Semantic Analysis (LSA), Word Embeddings (Word2Vec and GloVe). Latent Semantic Analysis (LSA) could compare words in essays to generate unique questions and study how words relate to articles, blogs, or product descriptions. Based on this comprehension, it recommends articles or products that align with user preferences. Seq2Seq Models create questions by comprehending provided text or ideas, making the questions fit better together. Seq2Seq models can translate text from one language to another in language translation applications. For example, websites such as Google Translate use Seq2Seq setups to change sentences from one language into sentences in another language. In search engines, Word2Vec or GloVe embedding assists in comprehending the significance of search words and documents by showing words as organized groups in space. This way, the search engine can locate relevant documents for users, even if they use varying words that mean the same thing.

4.1 Latent Semantic Analysis (LSA)

Latent Semantic Analysis is a way to understand how words and documents are related. It looks at several papers and their words, using a math trick called singular value

decomposition. LSA needs help with polysemous words, averaging their meanings and hindering accurate document comparison. Despite sharing principles with LSI, which focuses on indexing and retrieval, LSI is commonly used for organizing textual information. It utilizes SVD to detect patterns in word-document matrices, facilitating topic identification. LSA's four-step process involves converting unprocessed text into word-document matrices, performing SVD to unveil hidden connections, and encoding data to represent main topics. Essentially, LSA organizes messy papers by word frequency, identifies patterns, and summarizes main ideas, aiding researchers in swiftly accessing relevant information—Fig. 1 deliberate the structure of LSA.

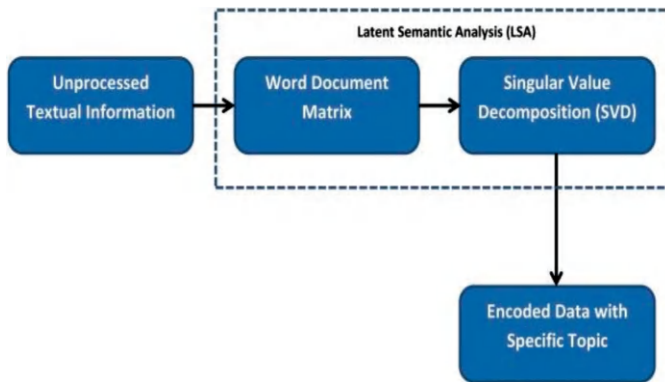


Fig. 1. Structure of Latent semantic analysis (LSA)

Latent Semantic Analysis (LSA) involves two main steps: firstly, constructing a word-document matrix, and secondly, converting this matrix into encoded data with a specific topic using Singular Value Decomposition (SVD). A word-document matrix is like a chart that helps computers make sense of written information. It neatly organizes words and documents, making it easier for computers to study them. For instance, it considers the presence of the word “air” in the first, second, and third documents shown in Table 1. Following converting documents into a word-document matrix, the subsequent step involves transforming this matrix into encoded data focusing on a specific topic using Singular Value Decomposition (SVD). Singular value decomposition (SVD) breaks down the original matrix A into three smaller matrices: U , Σ , and V^T . U and V^T show the main directions of variation in the rows and columns of A . Σ tells us how important each of these directions is. By multiplying these matrices together, we can reconstruct A . This helps us simplify data and find meaningful patterns. U , Σ , and V^T are like puzzle pieces that, when put together, help us understand the structure of the original matrix A more straightforwardly. $[A = U \Sigma V^T]$. By performing the calculation, $A = U * \Sigma * V^T$, we obtain a matrix that can be used to modify the word-document matrix, as illustrated in Table 1. Matrix Table 1 showcases a comparison of word similarities among five PDF documents, labelled D1 through D5, with some or most of the words displayed in the table. This comparison is based on a dataset that was randomly generated for experimental purposes.

Table 1. Matrix table after calculation by SVD

Algorithm	D1	D2	D3	D4	D5
Air	−0.153	−0.037	0.092	−0.026	0.040
Alive	0.079	0.079	0.020	−0.012	−0.002
Babbling	−0.074	0.079	0.020	−0.012	−0.002
Filled	−0.042	−0.050	0.009	0.022	0.104
Flowers	−0.042	−0.050	0.009	0.022	0.104
Frenetic	−0.042	−0.050	0.009	0.022	0.104
Gem	−0.042	−0.050	0.009	0.022	0.104
Scene	−0.031	−0.024	−0.095	−0.064	−0.011
Shoots	−0.031	−0.024	−0.095	−0.064	−0.011
Size	−0.031	−0.024	−0.095	−0.064	−0.011
Straight	−0.031	−0.024	−0.095	−0.064	−0.011

Table 1 presents the association between specific words and each document under consideration. To facilitate comparison, let’s examine the relationship between the words “Alive” and “Babbling” within the context of the second document. It becomes apparent that these words exhibit a strong correlation in this document. However, upon closer inspection of the first document, we observe a notable disparity between the positions of “Alive” and “Babbling” along the scalar spectrum. “Alive” is at 0.07900932, indicating a positive association, whereas “Babbling” is at −0.07421575, suggesting a negative correlation. Consequently, these two words appear at opposing ends of the scalar spectrum within the document. This observation underscores the nuanced semantic dynamics within the text, highlighting the intricate relationships between words and their contextual significance. Similarly, we can explore the inter-document relationships, as depicted in Table 3. Here, documents ‘Jagged’ and ‘Ocean’ exhibit a moderate level of similarity in dimension 1, with respective scores of −0.3518171 and −0.3106047. However, in dimension 3, the same documents show contrasting positions, scoring 0.51553128 and −0.30173536, respectively, indicating reduced similarity. This highlights the dynamic nature of document relationships across different dimensions, which is crucial for research analysis. LSA can enhance personalized learning experiences by recommending materials based on student progress and interests. Universities can utilize this for extensive digital libraries and research repositories. It can also help to manage and recommend resources across multiple schools.

Figure 2 deliberate the labelling word analysis. It can visualise the relationships between words in documents by plotting a graph using data from dimensions 1 and 2 for comparative analysis. It depicts such a graph, where words like “ocean” and “wind” are positioned closely together, indicating a strong association between them within the context of the documents. This spatial proximity suggests that these documents frequently co-occur or share similar semantic contexts in the analysed text. Moreover, the graph highlights a cluster of papers shaded in black, which signifies a distinct grouping of

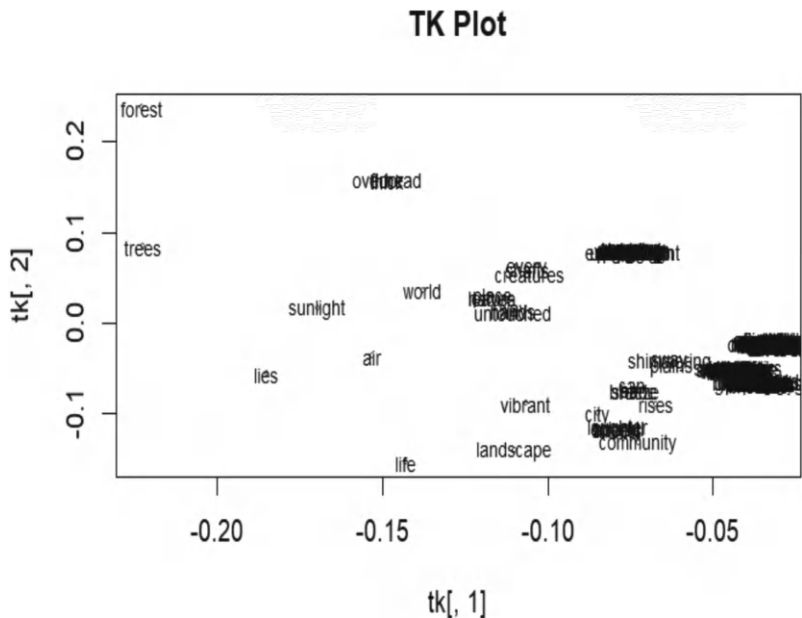


Fig. 2. Visual Analysis for Labelling in Eleven Words

documents that exhibit notable interrelatedness across different documents. This clustering phenomenon suggests that these documents are intricately connected or appear together in various textual contexts, potentially indicating specific thematic or conceptual clusters within the corpus. LSA is a robust tool that reveals concealed connections within text-based data. LSA streamlines information retrieval, document clustering, and recommendation systems by diminishing data complexity and identifying semantic similarities. It translates texts into a semantic realm that illuminates the structure of the corpus, enabling various natural language processing tasks to be carried out with greater ease. At the same time, LSA may encounter challenges with vast datasets and intricate linguistic nuances. Still, its adaptability and effectiveness render it a valuable asset in text analysis and information retrieval. Table 2 deliberates the document similarity comparison. Matrix Table 2 provides a comparison of overall document similarities across five different dimensions, labelled 1 to 5, as indicated by the respective file locations in the table. The dataset used for this comparison was randomly generated for experimental purposes.

4.2 Seq2Seq Model

Seq2Seq models have revolutionized natural language processing by effectively handling sequential data tasks through encoder-decoder architecture. Unlike phrase-based statistical machine translation (SMT) systems, which struggle with understanding long connections between words and overall meaning, Seq2Seq models leverage neural networks to overcome these limitations. In this architecture, the encoder processes input

Table 2. Document Similarity across Dimensions

Source	1	2	3	4	5
D:/Test/Depth.pdf	−0.7201	0.6678	0.165	−0.0895	−0.0134
D:/Test/Jagged.pdf	−0.3518	−0.5551	0.515	−0.2893	−0.4675
D:/Test/Metro.pdf	−0.4117	−0.4229	0.072	0.1734	0.7850
D:/Test/Ocean.pdf	−0.3106	−0.1619	−0.301	0.7927	−0.3973
D:/Test/Wind.pdf	−0.3029	−0.2019	−0.781	−0.4998	−0.0847

sequences into fixed-size hidden representations while the decoder generates output sequences using these representations, bridging the gap between input and output. The training involved examples with input and desired output sequences, allowing the model to learn patterns and connections through practice. This approach has made Seq2Seq models impactful across various domains, such as machine translation, text summarization, and dialogue generation. Recurrent neural networks (RNNs) are commonly employed in both the encoder and decoder segments, with the encoder condensing contextual information into fixed-length vector representations and the decoder utilizing these representations to produce desired output sequences. Employing a seq2seq model, we can convert a text from German to English, like “Das ist a Buch” to “This is a book”. The encoder understands the German sentence’s context, and the decoder generates the English translation relying on that encoded context. The encoder initially processes each word or symbol separately, converting them into vectors through techniques like one-hot encoding. These vectors are fed into a hidden neuron layer with summation and activation functions.

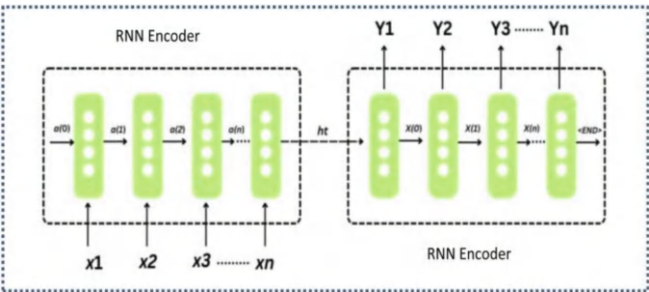


Fig. 3. The Architecture of seq2seq Model

Figure 3, a sequence of input words $X_1, X_2, X_3, \dots, X_n$, the activations $a(0), a(1), a(2), \dots, a(n)$ represent the output of each step, where the production of the previous step serves as the input for the next. Once the encoder receives all inputs, the translation process begins, as it requires the entirety of the input sequence to initiate. In the decoder stage, the translated sequence is generated word by word, converting symbols into words such as $y_1, y_2, y_3, \dots, y_n$. For instance, in the sentence “This is a book,”

each word “This,” “is,” “a,” and “book” corresponds to X1, X2, X3, and X4, respectively, while the outputs “Das,” “ist,” “an,” and “Buch” correspond to Y1, Y2, Y3, and Y4, respectively. The Seq2Seq model, often used in NLP tasks, is convenient for generating question papers automatically. Here’s how it works: First, you gather a large dataset containing questions paired with their answers or solutions. These questions can be sourced from textbooks, past papers, or other educational materials. Each question-answer pair is then turned into sequences of words or tokens, with the question serving as the input and the model of converting input questions into output solutions. Once the model is trained, you can build entire question papers. Give it a topic or prompt, and it will create a sequence of questions based on that input. You can adjust the length and complexity of the questions by tweaking parameters like temperature or beam size. Based on the evaluation, you should refine the model or adjust training data to improve the questions. This process is iterative, meaning you might undergo multiple training and refinement rounds for satisfactory results. Feedback from educators and students can also help improve the quality of generated question papers over time. Overall, using Seq2Seq models for automatic question paper generation streamlines the process and reduces the workload for educators while ensuring the creation of diverse and relevant assessments. After training a Seq2Seq model for automatic question paper generation, educators gain a valuable tool for crafting evaluations efficiently. They start by providing a prompt or topic, which guides the model in formulating questions. The model then applies its learned knowledge to generate a sequence of tokens representing questions. Seq2Seq models can help summaries lengthy academic articles and textbooks, assisting students to grasp essential information quickly. Universities with robust computational infrastructure can effectively train and deploy Seq2Seq models for diverse applications, from automated grading to research support. These platforms can leverage Seq2Seq models to provide real-time tutoring and adaptive learning experiences.

4.3 Word Embeddings

Word embedding is a crucial representation of words in natural language processing (NLP), represented as real-number vectors capturing semantic essence. They facilitate similarity comparisons by placing similar words closer within a vector space. Word2Vec and GloVe are prominent techniques for generating word embeddings, with Word2Vec using neural networks to learn word meanings from large text sets via Continuous Bag of Words (CBOW) and Skip-gram approaches. At the same time, GloVe combines global matrix factorization with local context windows to understand word meanings based on word co-occurrence frequencies. Table 3 deliberates the word embedding comparison.

The Word2Vec algorithm creates word embeddings by trying to guess a word based on the words around it in a sentence. These nearby words are called the context. These vectors help us understand how words are connected in sentences and are helpful for many tasks in understanding language better. GloVe works by looking at how often words appear together across the text. It creates a matrix that shows how likely one word will appear with another. Then, it learns vector representations by trying to make the dot product of word vectors match the logarithm of how often the words appear together. This way, GloVe captures the relationships between words based on how often they appear together in text. In Word2Vec, the word “apple” is embedded by considering its context

Table 3. Word Embedding’s Comparison between Word2Vec and GloVe

Word	apple	banana	orange	grape
Word2Vec	0.714655	0.889109	0.233248	0.6450
GloVe	0.12933	0.974596	0.485703	0.9459
Word	melon	dog	cat	bird
Word2Vec	0.466061	0.786054	0.647974	0.8369
GloVe	0.089749	0.148998	0.837583	0.4151
Word	elephant	tiger	car	bus
Word2Vec	0.269752	0.022436	0.591754	0.8004
GloVe	0.817887	0.538909	0.627539	0.5621
Word	bus	train	bicycle	motorcycle
Word2Vec	0.800482	0.687062	0.727144	0.2106
GloVe	0.562102	0.296381	0.632839	0.1730

words “love” and “eating,” with respective vectors [0.4, 0.3] and [0.5, 0.2]. By averaging these context vectors, the Word2Vec embedding for “apple” is derived as [0.45, 0.25]. Conversely, GloVe calculates the co-occurrence statistics for “apple” with “love” (10 times) and “eating” (5 times), resulting in an embedding of [0.6, 0.4]. Comparing the Word2Vec and GloVe embeddings reveals insights into their semantic similarities, with higher or closer values indicating more remarkable likeness in meaning.

5 Result and Discussion

To support the comparative analysis of Word2Vec, GloVe, LSA, and Seq2Seq models in Automatic Question Paper Generation (AQPG), we conducted a series of experiments using a dataset of randomly generated academic content across various subjects. The dataset included simulated academic papers from fields such as History, Geography, and Science. The comparison graph shows how Word2Vec and GloVe embeddings are different or similar for a group of words. Each word is shown on the left and right sides of the graph. If two lines are close together, the embeddings are similar. If they are far apart, it means they are different. By looking at the graph, we can understand how well these algorithms capture the meanings of words. The graph shows that close points have similar embeddings, while distant points differ. Parallel lines imply similar behavior while diverging ones indicate differences. This aids in evaluating Word2Vec and GloVe’s effectiveness with diverse words. Word embeddings, such as Word2Vec and GloVe, transform natural language processing by representing words as compact vectors, capturing their semantic connections. Three models are evaluated by comparing their Semantic Representation Quality, Context Understanding, and Computational Complexity. Figure 3 presented below offers a comprehensive comparison among four prominent models (Latent Semantic Analysis (LSA), Word2Vec, GloVe, and Seq2Seq) used in automatic question paper generation (AQPG). Semantic Representation quality

assesses a model's capability to grasp the intrinsic semantic relationships and similarities within textual data. Word2Vec and GloVe emerge as frontrunners in this aspect, scoring 9.2 and 8.8, respectively. These neural network-based models have exceptional proficiency in generating distributed representations of words within a continuous vector space. For instance, when applied to academic literature, Word2Vec and GloVe can accurately discern semantic connections between terms like "Land" and "soil," which often co-occur due to their relation with environment and geology. Context understanding is essential in assessing a model's ability to grasp contextual dependencies within continuous sequences of tokens. Seq2Seq Models are the forerunners in this aspect, achieving a score of 9.3. Utilizing recurrent neural networks (RNNs), Seq2Seq Models exhibit maximum comprehension of textual contexts. Logical reasoning questions can be generated in the given context. For instance, when formulating questions about historical events, a Seq2Seq Model can create queries that delve deeper than surface-level information, requiring analysis of cause-and-effect relationships or prediction of future outcomes. Computational Complexity reflects the resources needed for training and deploying a model, encompassing computational power and training data size. Figure 4 deliberates the word embedding comparison.

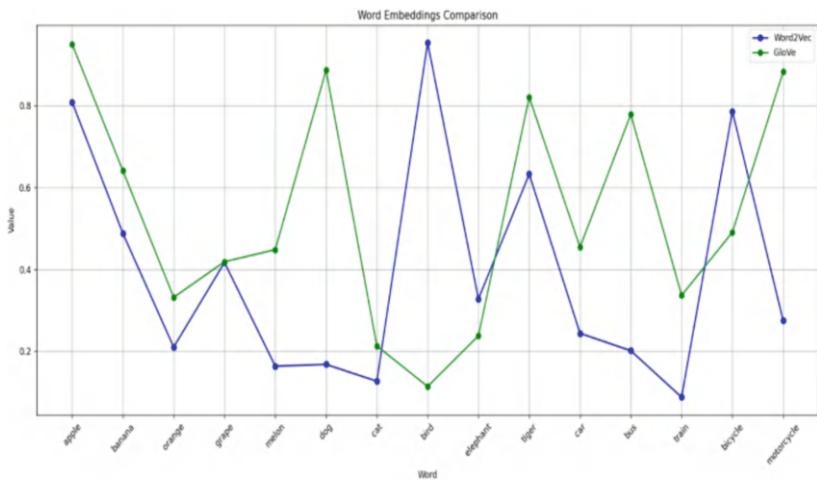


Fig. 4. Word Embedding Comparison

Seq2Seq Models exhibit the highest computational complexity, scoring 8.1. Despite offering unparalleled flexibility and context awareness, they demand substantial training data and computational resources. Conversely, LSA, a classical technique, boasts relatively lower computational complexity, making it more accessible for applications with limited resources. However, it may not capture higher-order linguistic patterns as effectively as neural network-based models like Word2Vec and Seq2Seq.

Figure 5, the grouped bar chart, provides valuable insights into comparing performance among LSA, Word2Vec, GloVe, and Seq2Seq Models in AQPG. While Word2Vec and GloVe excel in capturing semantic relationships, Seq2Seq Models demonstrate

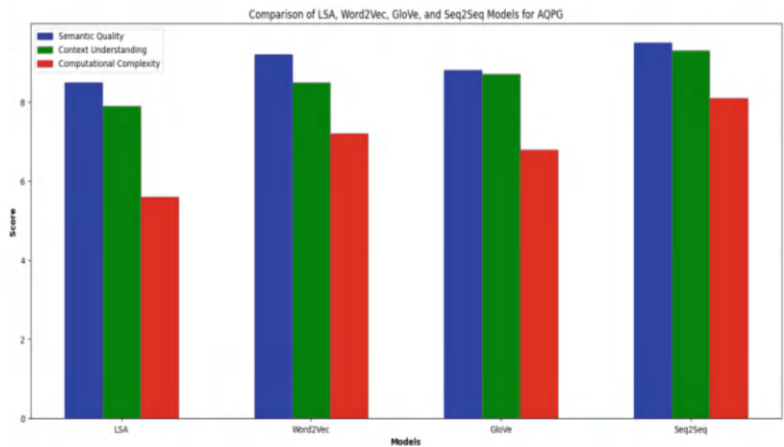


Fig. 5. Comparison of LSA, Word2Vec, Glove and Seq2Seq models

superior contextual comprehension. However, this enhanced performance comes with increased computational complexity, making them less suitable for resource-constrained settings. On the contrary, LSA offers a simpler alternative with lower computational requirements. These three models are categorized in light of task specifications, available resources, and desired performance benchmarks.

6 Conclusion

The exploration of Latent Semantic Analysis (LSA), Word Embeddings (Word2Vec and GloVe), and Seq2Seq Models for Automatic Question Paper Generation has provided valuable insights into the potential of these algorithms in revolutionizing the education sector. Through our comparative analysis, we have identified the strengths and limitations of each approach, shedding light on their feasibility and applicability in generating question papers. LSA has demonstrated its effectiveness in extracting essential concepts and developing diverse and meaningful questions by analyzing text documents. Seq2Seq models have proven adept at understanding and processing complex language patterns, enabling them to generate contextually relevant questions. By training these models on diverse topics and input-output sequences, educators can leverage their capabilities to produce high-quality question papers tailored to different educational levels and learning objectives. Word embeddings, particularly Word2Vec and GloVe, offer a novel approach to understanding word meanings and relationships within text. These algorithms present a promising opportunity to automate and optimize the question paper generation process, saving educators time and resources while ensuring the creation of relevant and diverse assessments.

Further research and practical implementation will be essential to harness these technologies' full potential in enhancing educational systems worldwide. By leveraging the capabilities of LSA, Seq2Seq models, and word embedding, educators can revolutionize how question papers are generated, ultimately leading to more effective and impactful

student learning outcomes. Comparing these techniques shows that LSA is simple yet lacks precision, while word embedding balances complexity and semantic understanding. Seq2Seq models, though complex, offer contextually rich mappings. Customization levels vary, with Seq2Seq models allowing the highest degree of fine-tuning.

Acknowledgments. Thanks to Center for Research, Christ University for provide all the computational facility for doing this research.

References

1. Xu, B., Meng, R., Chen, Y., Xu, X., Dong, C., Sun, H.: Latent semantic diffusion-based channel adaptive de-noising sem com for future 6g systems. In: GLOBECOM 2023–2023 IEEE Global Communications Conference, pp. 1229–1234. IEEE (2023)
2. Zhong, Q., Ding, L., Liu, J., Du, B., Tao, D.: E2s2: Encoding-enhanced sequence-to-sequence pretraining for language understanding and generation. *IEEE Trans. Knowl. Data Eng.* **36**(12), 8037–8050 (2024)
3. Joshua Johnson, S., Ramakrishna Murty, M., Navakanth, I.: A detailed review on word embedding techniques with emphasis on word2vec. *Multimed. Tools Appl.* **83**(13), 37979–38007 (2023). <https://doi.org/10.1007/s11042-023-17007-z>
4. Qian, K., Min, X.Y., Cheng, Y., Song, G., Min, F.: Self-dependence multi-label learning with double k for missing labels. *Artif. Intell. Rev.* **56**(6), 5057–5094 (2023)
5. Zhang, L., VanLehn, K.: Adaptively selecting biology questions generated from a semantic network. *Interact. Learn. Environ.* **25**(7), 828–846 (2017)
6. Valdez, D., Pickett, A.C., Young, B.R., Golden, S.: On mining words: the utility of topic models in health education research and practice. *Health Promot. Pract.* **22**(3), 309–312 (2021)
7. Huang, S., Zhou, X., Chin, S.: Application of seq2seq models on code correction. *Front. Artif. Intell.* **4**, 590215 (2021)
8. Haque, R., Hasanuzzaman, M., Way, A.: Analysing terminology translation errors in statistical and neural machine translation. *Mach. Transl.* **34**, 149–195 (2020)
9. Jiang, Y., Huang, L., Jia, Z., Yang, B.: Automatic question answering method based on IMGRU-Seq2seq. In: 2021 IEEE 6th International Conference on Computer and Communication Systems (ICCCS), pp. 499–504. IEEE (2021)
10. Lee, U., et al.: Few-shot is enough: exploring ChatGPT prompt engineering method for automatic question generation in English education. *Educ. Inf. Technol.* **29**(9), 11483–11515 (2023). <https://doi.org/10.1007/s10639-023-12249-8>
11. Camacho-Collados, J., Pilehvar, M.T.: Embeddings in natural language processing. In: Proceedings of the 28th International Conference on Computational Linguistics: Tutorial Abstracts, pp. 10–15 (2020)
12. Asudani, D.S., Nagwani, N.K., Singh, P.: Impact of word embedding models on text analytics in a deep learning environment: a review. *Artif. Intell. Rev.* **56**(9), 10345–10425 (2023)
13. Kim, S., Park, H., Lee, J.: Word2vec-based latent semantic analysis (W2V-LSA) for topic modelling: A study on blockchain technology trend analysis. *Expert Syst. Appl.* **152**, 113401 (2020)
14. Abraham, J., Cherian, G.J.: Systematic review on humanizing machine intelligence and artificial intelligence. In: 2023 Second International Conference on Electronics and Renewable Systems (ICEARS), pp. 1092–1097. IEEE (2023)
15. Pandey, A.K., Roy, S.S.: Natural language generation using sequential models: a survey. *Neural Process. Lett.* **55**(6), 7709–7742 (2023)



Exploring Information-Theoretic Measures for Feature Extraction in Image Processing: A Comprehensive Analysis and Future Directions

Romisha and Shruti Vashist^(✉)

Department of Electronics and Communication Engineering, School of Engineering, Manav Rachna University, Faridabad, India
khatriromisha27@gmail.com

Abstract. Information measures are often essential in image processing to assess and perform features suitable for representing quality images. Multiple criteria, including Entropy, mutual information, and divergence, are effective means of gaining insight into the nature of image data systems and their intricacy, organization, and connection. This paper discusses several methods that involve information theory to determine. Per Springer style, both city and country names must be present in the affiliations. Accordingly, we have inserted the city and country name. Please check and confirm if the inserted city and country name is correct. If not, please provide us with the correct city and country name. the effectiveness of the methods when used for feature extraction, considering the various image datasets and applications. The measures of choice, conversion of images into suitable representations, computation of the chosen measures for each image or region of interest in a playing time instance, feature extraction, methods of dimensionality reduction, and evaluation and comparison of various Information-theoretic metrics are all essential steps in this analysis. By analyzing the behavior of different information-theoretic metrics, one can determine the best approaches to feature extraction strategies in image processing. The investigation outcomes improved the advancement of image analysis, Classification, and retrieval techniques, which are resilient and effective in medical imaging, remote sensing, computer vision, and multimedia processing.

Keywords: Image Processing · Machine Learning · Entropy · Information Retrieval

1 Introduction

Enhancing image quality, retrieving valuable data, and making interpretation easier are the essence of image processing issues that involve modifying and analyzing digital images [1]. There are also goals for the methods and algorithms: noise reduction, image enhancement, feature extraction, segmentation, and pattern identification, to name several potential objectives [2]. Image processing applications include medical image analysis, satellite image analysis, surveillance systems, remote sensing systems, robotics,

multimedia, and more. Image processing can be defined as a set of methods that involve computer procedures and mathematical formulas to transform raw image data into useful information. This unlocks a lot of opportunities in the areas of innovation, automation, and decision-making in various fields [3]. This is a common area in image processing and refers to the overall process of enhancing a picture or isolating the required features or motifs in an image while rejecting all the unnecessary data or ‘noise’. So, the filter operation can blur or sharpen a picture to detect the edges of objects or remove noise by applying a function to each or some of the pixels in the picture. There is also a possibility of removing or at least decreasing the noise in the background by utilizing filters. Linear filters are familiar picture filters, classified into Gaussian and Median filters. To decrease the noise level, these filters smear or smoothen an image. Sobel and Laplacian are non-linear filters emphasizing features common filters ignore, such as edges and other high frequencies. Filtering image processing is vital, including computer vision jobs, feature extraction, and image restoration [4].

This is because it helps improve the quality of images and the accuracy of analyses. Information-theoretic Entropy, which derives from seminal work by Claude Shannon, can be used to measure the uncertainty or unpredictability of a random variable [5]. The amount of inside information inside the random variable can be measured with the help of it. From the perspective of Entropy, low Entropy is considered good because it represents less confusion or randomness. In comparison, high Entropy is considered bad as it represents more confusion or randomness. The Entropy of a random process gives the average level of randomness of the results of the process. The maximum Entropy is attainable by constant movement of the items from the beginning to the end following a uniform distribution successfully. Because it is the essential starting point from which virtually all other random or variable system analyses emanate, information-theoretic Entropy is used in communication theory, cryptography, data compression, and other fields [6]. Generalized information theoretic entropy tries to extrapolate basic entropy definitions to capture the other types of uncertainty and information content complexity. It is possible to award generalized entropy metrics that are all within the characteristics of data, as well as dependency and non-uniform distributions.

Contrary to classical Entropy, which presupposes the uniform distributions of probability and is based on the Shannon entropy formula [7], From physics and biology, economics, and information theory, the metrics mentioned were a versatile approach to measuring diversity, complexity, and uncertainty. Improved understanding of complex system behavior is achieved by generalized information theoretic entropy utilizing increasingly complex mathematical formulations like Rényi entropy, Tsallis entropy, or Kullback-Leibler divergence. Application implementation in data compression, machine learning, signal processing, and others becomes more straightforward.

2 Application of Generalized Information Theoretic Entropy

Generalized Information Theoretic Entropy, frequently employed in the context of generalized entropy measurements, has applications in various domains, including information theory, physics, statistics, and computer science, among others. A representation of generalized information theory entropy can be found in Fig. 1.

2.1 Data Compression

Data compression techniques quantify non-densities of information content by generalized Entropy based on Rényi entropy or Tsallis entropy. This way, encoding, and compression schemes are more efficient—the efficiency of compression aids in managing multiple media formats. Computer data compression using generalized entropy measures is of tremendous value in understanding data's information content and constructing efficient compression algorithms. Some examples of metrics of the generalized Entropy used in the data compression process are Rényi and Tsallis entropies. It is high time to take a closer look at their mathematical expressions and potential uses in data compression to gather more information about each of them [8]:

Rényi Entropy:

Rényi entropy is a generalized form of Shannon entropy and is defined as:

$$H_{\alpha}(X) = \frac{1}{1-\alpha} \log \sum_i p_i^{\alpha} = \frac{1}{1-\alpha} \ln \sum_i p_i^{\alpha}$$

where:

- $H_{\alpha}(X)$ is the Rényi entropy of random variable X with probability mass function p_i .
- α is the order parameter, which controls the degree of generalization. As α approaches 1, Rényi entropy converges to Shannon entropy.

Data compression uses Rényi entropy, which quantifies a data source's information content or uncertainty. To find redundant or repetitive patterns in the data, compression algorithms use Rényi entropy, which allows for more efficient encoding strategies.

Tsallis Entropy:

Tsallis entropy is another generalization of Shannon entropy and is defined as:

$$H_q(X) = \frac{1}{q-1} \left(1 - \sum_i p_i^q \right) = \frac{1}{q-1} \ln \sum_i p_i^q$$

where:

- $H_q(X)$ is the Tsallis entropy of random variable X with probability mass function p_i .
- q is the entropic index, controlling the degree of deviation from the traditional Boltzmann-Gibbs entropy.

The Entropy of Tsallis highly enhances data compression because it addresses non-extensive phenomena and long-range dependencies in the data feeds. When dealing with non-Gaussian/non-extensive datasets, one can enhance the compression ratios by integrating Tsallis entropy in the schemes. This is the way to enhance compression ratios at that. Thus, Rényi and Tsallis entropies are generic means of dissecting uncertainty and information content to build compression algorithms [9] efficiently. Using these entropy measurements in compression algorithms allows data sources' statistical properties to be used more effectively, meaning superior compression performance and reduced storage requirements.

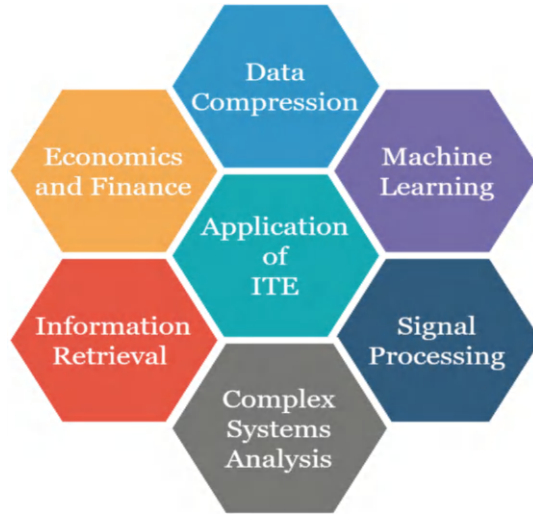


Fig. 1. Application of Generalized Information Theoretic Entropy

2.2 Machine Learning

Data clustering and Classification are significant parts of the data analysis and machine learning processes because both try to categorize information into groups. This means that the methods deal with the arrangement of data in a particular manner. These tasks are essential with generalized entropy functions used to measure diversity or uncertainty in a data cluster. They regulate the procedure's landmark step of pattern and structure identification.

The rationale for clustering is to partition a data set into divisions of like data points known as clusters, where the similarities amongst the points in a cluster are higher than those within other clusters. Specific techniques can be used to quantify the dissimilarity of clusters to each other or speak of their similarity or coherence, and one such, as mentioned above, is the generalized entropy functions, which can be regarded as objective functions.

Applying entropy-based clustering is a standard procedure; such algorithms are designed to achieve the maximum potential of clusters' separation and the lowest Entropy within the clusters. An appropriate choice of entropy function can be made based on certain data features and the clusters' characteristics to be formed [10].

Here is a mathematical formulation of the objective function for entropy-based clustering:

$$\text{Minimize } \sum_k = 1KH(C_k)$$

where:

- K is the number of clusters.
- C_k represents the k -th cluster.
- $H(C_k)$ is the Entropy of cluster C_k , computed using a generalized entropy function.

The entropy $H(C_k)$ represents the dispersion or spread of data points inside the cluster, capturing the uncertainty or variety within cluster C_k . The clustering algorithm aims to produce more consistent clusters with well-defined boundaries by reducing the Entropy.

The raw intention of Classification is to assign the data points to the classes that have been defined based on measured features. The generalized entropy functions are deemed usable as similarity or divergence measures of how far a similarity is, how far the data points or data values are from the class values, or how different they are.

In k-nearest neighbors (k-NN) Classification, for example, the similarity between data points is determined by the choice of distance measure. Compared to more conventional measurements like Euclidean distance, entropy-based metrics like Renyi and Tsallis entropy are better at capturing the distribution's structural features.

The generalized entropy-based distance metric for Classification can be expressed mathematically as:

$$D(x_i, x_j) = H(x_i, x_j)$$

where:

x_i and x_j are two data points.

$H(x_i, x_j)$ is the entropy-based distance between x_i and x_j , computed using a generalized entropy function.

The classifier can label data points according to their feature distributions' similarity because the entropy-based distance metric measures the diversity or uncertainty between the data points. Regarding clustering and classification tasks, generalized entropy functions are beneficial. They offer metrics for uncertainty, variety, and similarity, which help find significant patterns and structures in data. These entropy-based methods improve the performance and understandability of classification and clustering algorithms in different fields and with different uses.

2.3 Signal Processing

Signal analysis employs the generalized entropy values for measuring the signals' complexity and irregularity levels. Mobile communications, among other fields, apply these techniques in feature extraction, noise reduction, and anomaly identification based on biological signal processing. Summarizing, in consideration of the complexity and irregularity of the signals, the generalized entropy metrics are one of the most essential investigatory tools in the field of signal processing. The information offered by these metrics is quite fundamental to several activities that include feature extraction, noise elimination, and anomaly detection. In biomedical signal processing, they understand that the most essential activity is to realize complicated physiological phenomena. In telecommunications, where reliability is essential, these entropy metrics cannot be spared. In turn, evaluating signals' uncertainty and information content enables using generalized entropy measures to extract notable features, which is helpful in the search for significant patterns and structures. Besides, they assist denoising algorithms in rejecting interference while preserving functional signal elements, thus improving the signal's cleanliness. Here, generalized entropy measures are also applicable in the case of anomaly detection.

These metrics assist in identifying abnormalities or even potential problems through the evaluation of when the signal's actions differ from what was anticipated. Generalized entropy measures in signal processing enhance the efficiency of analysis and the use of accurate information for decision-making, enabling the development of prevalent sectors [11].

2.4 Complex Systems Analysis

In the study of complex systems such as ecological networks, social dynamics, or financial markets, entropy measurements play a crucial role. These metrics assess the diversity, stability, and predictability of system behavior, which enables researchers to gain insights into emergent features and critical transitions. In complex systems analysis, entropy measures are essential to comprehend the behavior of complex systems such as ecological networks, social dynamics, and financial markets. These metrics are significant when measuring essential aspects of complex systems, such as diversity, stability, and predictability. They provide valuable insights into emergent properties and critical transitions with their ability to quantify these traits. Complex systems are characterized by several interrelated components that display nonlinear interactions and emergent events. As a result, it is difficult to predict the behavior of complex systems using classic reductionist approaches [12]. The measures of Entropy introduce the manner of defining these systems' complexity and uncertainty, which opens the look at the dynamics and structuring in the backdrop. This study uses entropy measurements to evaluate the amount of biodiversity, the level of resilience, and ecosystem stability in ecological networks. The odds of the system's robustness and capability to bear shocks and adjust to the surroundings form the core of entropy-based assessments. This is done by evaluating the various rates of a variety of species within the ecosystem and their dispersion. It is also possible to find the tipping points and regime shifts with entropy measurements and indicators of critical transition in the ecological processes.

Likewise, entropy measurements offer essential information on the extent of contact communication and the overall change and consistency of social systems regarding social dynamics. To explain the formation of collective behaviors, opinions, and trends, it is crucial to employ entropy-based methodologies that analyze the communication and cooperation patterns within social networks. Besides, the entropy measures provide insights into the stability of the social systems and their vulnerability to shocks/interference from the outside world that assists in strategy for crisis disposition and policy intervention.

In general, entropy measures are applied to analyze the nature of financial market efficiency, risk level, and asset prices. Suppose the concept of Entropy refers to the level of disorder in the markets. In that case, entropic analysis can reveal tendencies in the behavior of markets, moods of the investors, and system risks. These analyses quantify the uncertainty in the data and information content in financial statements. Further, entropy measures can be used to detect the period of market fluctuations, bubbles, and financial cross-influences, which is helpful in decision-making in the field of managing risks and investments. In general, entropy measurements are critically important in analyzing complex systems since the variety in the system's behavior can be more easily incorporated, and an improved understanding of observed emergent phenomena and tipping points can be attained. Entropy-based approaches, by their application in

various fields, advance our knowledge of complex systems and inform decision-making processes that aim to achieve sustainability, resilience, and stability [13].

2.5 Information Retrieval

This holds especially when it comes to improving the effectiveness and efficiency of information search systems; this mandates query optimization. This is particularly true when the two are extreme and refer to a vast situation regarding total size and geographical distribution. Due to the identification of basic inherent structures and patterns of document collections, generalized entropy measures assist in improving query processing. Hence, to enhance the overall retrieval effectiveness and assist the users, there are certain methods that can be applied to incorporate query optimizers to enhance search strategies and eliminate any possible spurious documents that cannot be considered sufficiently important. This is conducted by employing entropy-based measures for assessing documents' informational nature and topics' diversity. Incorporation of the generalized entropy measures into an information retrieval system enables academics and practitioners to develop more complex and robust algorithms in document ranking and scoring, as well as query optimization processes. Those tied with entropy aid in the growth of information search technologies since they are founded on the principles of information theory. They facilitate efficient and effective access to the required information expounding on various fields and applications [14].

3 Behaviors Aspect of Information-Theoretic Measures

In examining the behavior of different information-theoretical parameters in image processing, the questions about the capabilities of these measures defining the essential features of images were considered. Thus, Entropy, mutual information, and divergence are examples of information-theoretic measurements that give valuable insights into image data's complexity, structurality, and relational nature. To do so, academics can compare these measurements with different datasets and applications of feature extraction and image analysis [15]. The following are essential steps in this analysis:

- **Selection of Information-Theoretic Measures:** Determine a collection of information-theoretic measures that are appropriate for image analysis, taking into account elements such as Entropy, mutual information, conditional Entropy, and divergence metrics (for example, Kullback-Leibler divergence).
- **Image Representation:** To make the computation of information-theoretic measures easier, images should be converted into appropriate representations such as grayscale, color channels, and texture descriptors. Images should be preprocessed as required (for example, by normalizing or scaling them) to guarantee consistency and comparability between datasets.
- **Computation of Information-Theoretic Measures:** To determine the information-theoretic metrics chosen for each image or image region of interest, perform the calculation. For instance, one might compute Entropy to measure the degree of uncertainty or unpredictability within pixel intensity distributions. Similarly, one can calculate mutual information to represent the dependencies between various image regions.

- **Feature Extraction:** To represent essential aspects of the images, extract features from the information-theoretic measurements that have been computed. These features can capture specific image qualities, such as texture and shape, which may include entropy values, mutual information scores, divergence measurements, or derived descriptors.
- **Dimensionality Reduction:** To lower the dimensionality of the extracted features while maintaining their ability to discriminate, it is necessary to use dimensionality reduction techniques such as principal component analysis and t-distributed stochastic neighbor embedding. This phase serves to alleviate the effects of the curse of dimensionality and provides an increase in the efficiency of computation.
- **Evaluation and Comparison:** To evaluate the effectiveness of various information-theoretic techniques in collecting crucial image properties, appropriate evaluation metrics should be utilized. Some examples of such metrics include classification accuracy and clustering validity indices. This comparison aims to determine which techniques are the most appropriate for particular applications by comparing the effectiveness of various measurements across various picture datasets and analytic tasks.
- **Visualization and Interpretation:** By visualizing the extracted characteristics in a space with reduced dimensions, you can acquire insights about the distribution of these features and their capacity to be separated. The meaning of the extracted features should be interpreted based on the attributes and characteristics of the underlying image, such as the texture, color composition, or spatial structure.

Researchers can improve their understanding of image content and structure by systematically analyzing the behavior of information-theoretic measures in image feature extraction. This, in turn, leads to the development of improved algorithms for image analysis, Classification, and retrieval across a wide range of domains, such as medical imaging, remote sensing, computer vision, and multimedia processing.

4 Conclusion and Future Scope

Mutual information coefficients are applicable in image processing because they show how complex, structured, and interconnected image data is. This paper contributes to the authors' work of using metrics such as Entropy, mutual information, and divergence to identify elements that effectively represent the intrinsic characteristics of images. Scholars can integrate feature extraction strategies for image processing tasks by surveying the behavior of the entropy-based measures. Potential future research in this area could, therefore, be done through the following: First of all, further improvements in the methodology used for feature extraction can be achieved by investigating new information-theoretic measures to be used in vision processing. Current approaches to picture analysis, Classification, and further flexible and efficient picture indexation for their retrieval may be further enriched by future developments that investigate the possibility of augmenting the metrics coming from information theory with deep learning solutions.

In addition, to have a more adequate understanding of the usefulness of different information theoretical measures, it is plausible to extend the analysis to much more

extensive and diversified picture sets originating from different domains like medical imagery, remote sensing, computer vision, multimedia processing, etc. In addition, industries heavily dependent on image processing technology could result in real advancements as regards the consequence of studying the impact of information-theoretic measures on applicability in medical diagnosis, object recognition, and content-based image retrieval. It is necessary to explore information-theoretic metrics for image processing to develop further opportunities for image analysis systems and overcome the problems in different fields. Possible work to be done in the future could be the inclusion of mutual information coefficients with the new deep learning algorithms in the extraction of features and Image processing. Perhaps new information-theoretic measures and ways to be used in different image processing-related problems, including classification and retrieval problems, may be beneficial. Expanding the study for diverse and more extensive scale data from domains such as medical image analysis and remote sensing could further reveal more insights into these measures' efficacy. Also, improvements may be witnessed by understanding the influence of the information-theoretic measures on practical application, like diagnosing a disease or retrieving specific pictures based on their content.

References

1. Archana, R., Jeevaraj, P.S.E.: Deep learning models for digital image processing: a review. *Artif. Intell. Rev.* **57**(1), 11 (2024)
2. Kaur, H., Koundal, D., Kadyan, V.: Image fusion techniques: a survey. *Arch. Comput. methods Eng.* **28**(7), 4425–4447 (2021)
3. Dastres, R., Soori, M.: Advanced image processing systems. *Int. J. Imagining Robot.* **21**(1), 27–44 (2021)
4. Li, P., Wang, H., Yu, M., Li, Y.: Overview of image smoothing algorithms. *J. Phys: Conf. Ser.* **1883**(1), 12024 (2021)
5. Moskowitz, I.S., Russell, S., Suri, N.: The value of information and the Internet of Things. In: *Artific. Intell. Internet of Everything*, pp. 145–169. Elsevier (2019). <https://doi.org/10.1016/B978-0-12-817636-8.00009-0>
6. Hoang, T.M., Vahid, A., Tuan, H.D., Hanzo, L.: Physical layer authentication and security design in the machine learning era. *IEEE Commun. Surv. Tutorials* **26**(3), 1830–1860 (2024). <https://doi.org/10.1109/COMST.2024.3363639>
7. Liu, J.: Shaping Strategies to Embrace Nonlinear Effects in Optical Fiber Communications. Institut Polytechnique de Paris (2024)
8. Jayasankar, U., Thirumal, V., Ponnurangam, D.: A survey on data compression techniques: from the perspective of data quality, coding schemes, data type and applications. *J. King Saud Univ. Inf. Sci.* **33**(2), 119–140 (2021)
9. Wang, S., Fan, J.: Image thresholding method based on Tsallis entropy correlation. *Multimed Tools Appl* **84**(11), 9749–9785 (2024). <https://doi.org/10.1007/s11042-024-19332-3>
10. Feldman, D., Schmidt, M., Sohler, C.: Turning big data into tiny data: constant-size coresets for k-means, PCA, and projective clustering. *SIAM J. Comput.* **49**(3), 601–657 (2020)
11. Mayor, D., Panday, D., Kandel, H.K., Steffert, T., Banks, D.: Ceps: an open access matlab graphical user interface (gui) for the analysis of complexity and entropy in physiological signals. *Entropy* **23**(3), 321 (2021)
12. Jakimowicz, A.: The role of entropy in the development of economics. *Entropy* **22**(4), 452 (2020)

13. Guo, J., et al.: A deep look into neural ranking models for information retrieval. *Inf. Process. & Manag.* **57**(6), 102067 (2020)
14. Zehlike, M., Yang, K., Stoyanovich, J.: Fairness in ranking, part ii: Learning-to-rank and recommender systems. *ACM Comput. Surv.* **55**(6), 1–41 (2022)
15. Nguyen, A., McMullin, O., Lizier, J.T., Fulcher, B.D.: A feature-based information-theoretic approach for detecting interpretable, long-timescale pairwise interactions from time series (2024). *arXiv Prepr. arXiv2404.05929*



Enhancing Text Classification in Natural Language Processing: A Comparative Study of Transformer Models and the Potential of Few-Shot Learning

Syed Hur Abbas^(✉), Gitanjali Wadhwa, and Sandra Hummel

ScaDS.AI Dresden/Leipzig (ZIH), Technische Universität Dresden (TUD), 01069 Dresden, Germany

{syed_hur.abbas, gitanjali.wadhwa, Sandra.hummel}@tu-dresden.de

Abstract. This research focuses on enhancing machine comprehension in multilingual Natural Language Processing (NLP) environments. Despite advancements in pre-trained models, creating custom models still requires more resources. To overcome this, the study explores FewShot Learning (FSL), a Meta-Learning approach inspired by human learning efficiency, asserting that machines can adeptly learn from minimal examples and task descriptions. The methodology unfolds in two dimensions: practical application and theoretical exploration. In practical terms, Python constructs models using pre-trained frameworks—BERT, DistilBERT, ELECTRA, and MiniLM. These Few-Shot models undergo meticulous evaluation for efficiency and accuracy, especially in processing unseen data with minimal support for contextual understanding. Simultaneously, the theoretical facet involves a comprehensive literature review, shedding light on FSL's effectiveness in diverse NLP contexts. Preliminary findings suggest FSL's promising role in addressing multilingual challenges in NLP, acknowledging limitations in complex linguistic scenarios. This study offers valuable insights into FSL's practical applications and limitations, laying the foundation for future investigations. The nuanced exploration contributes to a balanced understanding of FSL's applicability, potentially guiding the development of more advanced and resource-efficient NLP models.

Keywords: Large Language Model (LLM) · Artificial Intelligence · Natural Language Model (NLP) · Few-shot learning (FSL)

1 Introduction

In the dynamic and rapidly evolving landscape of Natural Language Processing (NLP), text classification poses a significant and multifaceted challenge. As a critical subfield of Data Science, NLP focuses on understanding the intricate complexities of human language syntax, semantics, and pragmatics, with the ultimate goal of categorizing text based on various characteristics, such as topic, emotion, or intent. This process is of

paramount importance for structuring and deriving meaningful insights from the vast and ever-growing volume of digital text data, including emails, social media posts, and archival records.

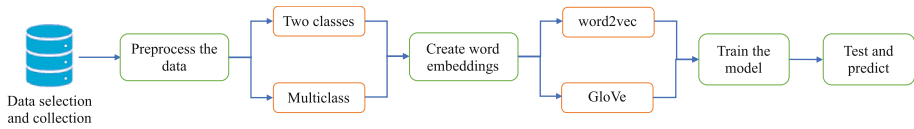


Fig. 1. The overall process of text classification.

Traditional text classification approaches, which rely on machine learning models, necessitate large volumes of labeled training data. However, the acquisition of such data is often costly, time-consuming, and fraught with legal and privacy issues, thereby hindering the development and deployment of effective text classification systems. To address this challenge, Pre-trained Language Models (PLMs) such as BERT, GPT-3, and RoBERTa have emerged as promising solutions, leveraging zero-shot and few-shot learning techniques (Devlin et al., 2019; Yin & Roth, 2019; Pourpanah et al., 2022). These models can perform classification tasks on previously unseen data, making them highly versatile and adaptable to a wide range of tasks without the need for extensive task-specific training.

Few-shot learning, which involves optimizing models on a small number of task-specific instances, further enhances the capabilities of PLMs in the realm of text classification (Wang et al., 2020b; Li et al., 2019). By enabling accurate predictions with limited labeled instances, few-shot learning has the potential to revolutionize the way we approach text classification, particularly in scenarios where labeled data is scarce or difficult to obtain. Moreover, advanced representations from PLMs like BERT, RoBERTa, GPT-3, DistilBERT, ELECTRA, and MiniLM, obtained after training on substantial and diverse datasets, provide dense and highly informative feature representations that can be effectively utilized to predict unseen cases (Liu et al., 2021; Wang et al., 2020a; Yang et al., 2022; Sanh et al., 2019; Clark et al., 2020; Wang et al., 2020c). Techniques such as transfer learning and meta-learning further empower classifiers to identify and adapt to unknown classes with limited labeled samples, thereby enhancing the overall robustness and adaptability of text classification systems (Han et al., 2021; Ruder et al., 2019; Li et al., 2020).

Recognizing the pressing need for accessible, efficient, and less resource-intensive text classification models, this study aims to contribute valuable insights and innovations to the NLP community. We conduct an in-depth investigation of the accuracy and reliability of Few-Shot Learning (FSL) in the absence of labeled data, exploring the potential of various pre-trained models in conjunction with cutting-edge techniques such as transfer learning and meta-learning. Furthermore, we introduce and evaluate the efficacy of interpretable machine learning (IML) techniques for debugging and reliable assessment of FSL-based models, with the goal of enhancing their practicality, transparency, and trustworthiness. By addressing these critical aspects, we aim to bridge the gap between the data-intensive nature of traditional text classification approaches and the need for

efficient and adaptable solutions in the face of ever-evolving and increasingly complex language understanding tasks.

2 Background

(Khurana et al. 2023) The field of Natural Language Processing (NLP) has witnessed remarkable advancements, driven by the increasing availability of data and the development of sophisticated machine learning algorithms. However, a significant challenge remains in the resource-intensive nature of training custom NLP models, which often require large amounts of labeled data to achieve satisfactory performance. Few-shot learning (FSL) has emerged as a promising solution to this challenge, taking inspiration from human learning capabilities to handle limited labeled data.

This research aims to enhance text classification in NLP by leveraging FSL and exploring the potential of transformer models. We build upon a rich body of literature that has contributed to the development of FSL and its application to NLP tasks. Wang et al. (2020b) provide a comprehensive survey of FSL, categorizing it from data, model, and algorithm perspectives, which serves as a foundation for our work. Smith et al. (2019) demonstrate the adaptability of FSL in sentiment analysis and other NLP tasks, highlighting its efficacy with limited labeled data, aligning with our focus on text classification.

Chen and Zhang (2021) introduce strategies for fine-tuning pre-trained models with FSL, which is relevant to our use of models like BERT and GPT-J. Sung et al. (2018) present the Relation Network for few-shot learning, which is pertinent to our goal of efficient learning from limited examples. Ravi and Larochelle (2016) provide insights into optimizing classifiers with an LSTM-based meta-learner, enhancing generalization capabilities with limited data. Ye et al. (2021) address cross-task generalization in FSL, introducing the CROSSFIT Challenge that standardizes task setup, training, and evaluation in NLP, which aligns with our objectives.

Izacard et al. (2022) discuss “Atlas,” a model for knowledge-intensive tasks in FSL, offering valuable insights for our investigation into retrieval-augmented language models. Min et al. (2021) contributes to our understanding with the “Noisy Channel Language Model Prompting,” an approach for few-shot text classification. Brown et al. (2020) highlights the impact of scaling language models like GPT-3, showing robust performance in zero-shot, one-shot, and few-shot settings without explicit fine-tuning, which supports our exploration of model scalability.

Tunstall et al. (2022) introduce SETFIT, a prompt-free framework for few-shot fine-tuning of Sentence Transformers, which aligns closely with our research on prompt-free methods. Pan and Li (2019) focus on few-shot transfer learning with lightweight word embedding-based models, emphasizing parameter-free representations and effective knowledge transfer. Zhao et al. (2022) explore fully zero-shot learning with PLMs using NPPrompt, a nonparametric approach that requires no labeled data, resonating with our aim to enhance model performance across NLP tasks.

Wei et al. (2021) introduce instruction tuning to improve zero-shot learning abilities, showcasing the effectiveness of finetuning on diverse datasets with natural language instructions, which is relevant to our optimization efforts. Gao et al. (2020) present LM-BFF for fine-tuning language models on limited examples, aligning with our objectives to

improve performance in low-resource settings. Bansal et al. (2019) address meta-learning for diverse NLP tasks with LEOPARD, an optimization-based meta-learning method that generalizes well to new tasks with minimal labeled data, providing significant accuracy gains across various NLP domains.

Our research aims to extend the methodologies of these foundational works to enhance text classification in NLP. By comparing transformer models and exploring the potential of FSL, we aim to provide a comprehensive understanding of the strengths and limitations of few-shot models in addressing multilingual challenges in NLP. We believe that our research has the potential to make a significant contribution to the field and pave the way for the development of more efficient and effective NLP systems.

3 Methodology

In the realm of software engineering, extensive research has addressed text classification challenges. However, a critical gap exists in exploring the application of few-shot learning techniques to enhance these tasks. This section aims to fill this void by investigating the potential of few-shot learning in text classification.

3.1 Natural Language Processing (NLP)

Natural Language Processing (NLP) is a cornerstone of computer science, imbuing machines with the ability to comprehend human language, both written and spoken. Through the integration of AI, deep learning models, and computational linguistics, NLP enables computers to grasp not just the literal meaning but also the subtleties, intentions, and emotions conveyed in human communication (Chowdhary, 2020). Its impact extends across various domains, from everyday technologies like voice-activated GPS systems and virtual assistants to strategic initiatives in large-scale enterprises, optimizing operations and enhancing productivity (Indurkha & Damerau, 2010).

3.2 Zero-Shot/Few-Shot Learning in NLP

Zero-shot and few-shot learning techniques represent innovative approaches to NLP tasks. By leveraging pre-trained models, these methods enable machines to generalize their understanding beyond specific training data, even with minimal adaptation (Wang & Yao, 2019). For instance, in characterizing data, models trained on vast text corpora can be fine-tuned with logical class names, showcasing remarkable adaptability (Bansal, 2022).

3.3 Few-Shot Classification

One of the primary challenges in text categorization within NLP is the absence of standardized benchmarks for few-shot learning. To tackle this, our methodology draws on recent research advocating the utilization of pre-trained language models (PLMs) for cross-task transfer (Yin & Roth, 2019; Schick & Schütze, 2020; Chen & Zhang,

2021). Figure 1 illustrates the architecture of our proposed few-shot learning algorithm, emphasizing its adaptability and robustness in scenarios with limited data.

Few-shot learning (FSL) is a machine learning paradigm that aims to enable models to learn from a small number of examples. FSL has gained significant attention in the NLP community due to its potential to address the challenge of data scarcity. However, FSL is not without its limitations. One major challenge is the issue of overfitting, where the model becomes too specialized to the few examples provided and fails to generalize to new, unseen data. Additionally, the quality and diversity of the few examples provided can greatly impact the model’s performance, making it difficult to ensure consistent and reliable results.

Our proposed few-shot learning algorithm for text categorization seeks to address these challenges by leveraging the power of pre-trained language models. Figure 2 depicts the comprehensive architecture of our pioneering few-shot learning algorithm. The framework showcases a meticulously designed structure comprising meta-learning components, including a base model network and a meta-learner. The base model adeptly captures and generalizes from limited training data, while the meta-learner swiftly adapts to new tasks through meta-knowledge assimilation. The algorithm’s architecture embodies an intricate interplay between these elements, enabling rapid adaptation and robust performance when confronted with limited data scenarios, a hallmark of effective few-shot learning paradigms.

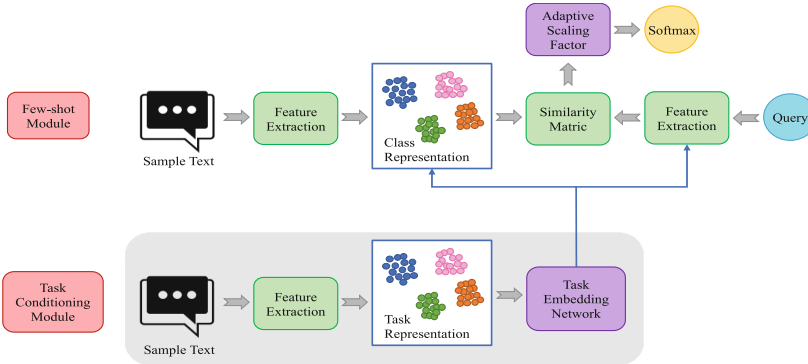


Fig. 2. The architecture of the proposed few-shot learning algorithm.

The mathematical representation provided encapsulates the essence of few-shot text categorization in few-shot text categorization. By adapting pre-trained language models to new tasks with limited labeled data, we minimize task-specific loss, showcasing the model’s ability to make accurate predictions even with constrained data. This mathematical formulation is not merely an equation; it is a precise articulation of our approach, aligning with rigorous standards of clarity and transparency:

- D_T : Target dataset (*few-shot or limited data*)
- M : Pre-trained language model
- T : Fine-tuning process

The objective is to minimize the task-specific loss on the target dataset D_t while finetuning the language model:

$$\min L(M_{\theta_M}(D_t) \text{task-specific labels } \theta_M) \text{ Where:}$$

- θ_M represents the parameters of the pre-trained language model M .
- $M_{\theta_M}(D_t)$ denotes the output of the language model M on the target dataset D_t after fine-tuning.
- L represents a task-specific loss function that measures the dissimilarity between the model's predictions and the true task-specific labels.

The process involves adapting the pre-trained language model (M) to the specific task defined by the target dataset (D_t) by adjusting its parameters (θ_M). This adaptation allows the model to make accurate predictions on the new task, even when the amount of labeled data available for fine-tuning is limited. The effectiveness of this approach lies in the ability of pre-trained models to capture general linguistic patterns from extensive input samples, which can then be fine-tuned to perform well on specific text categorization tasks. This mathematical representation encapsulates the core idea of leveraging crosstask transfer using pre-trained language models for few-shot text categorization in natural language processing.

3.4 Experimental Setup

Unveiling the Masked Language Model: We reframe the text categorization challenge as a Masked Language Model, seamlessly integrating it with NLP problems (Schick & Schütze, 2020; Yin & Roth, 2019). Our experimental setup meticulously investigates exploration-related issues in text categorization, elucidating the roles of the cloze method and the entailment approach.

Software, Tools, and Resource: Given resource constraints, we adopt the Jupyter Notebook environment, leveraging hardware configurations optimized for computational efficiency. This setup, comprising an i7 quad-core processor, 16 GB of RAM, and a GTX960M GPU, strikes a balance between computational capabilities and available resources.

3.5 Dataset

The AG's News dataset is a widely used benchmark dataset for text classification tasks (Zhang X. et al., 2015). It was created by collecting news articles from more than 2,000 news sources and annotating them with one of four categories: World, Sports, Business, and Sci/Tech. The dataset consists of 120,000 training samples and 7,600 testing samples, with each sample containing a title and a description. The dataset is well-balanced, with each category containing an equal number of samples.

The dataset is provided in two files: train.csv and test.csv. The train.csv file contains the training samples, while the test.csv file contains the testing samples. Each file has three columns: class, title, and description. The class column indicates the category of the news article, while the title and description columns contain the title and description

of the article, respectively. The title and description are enclosed in double quotes to avoid ambiguity, and internal double quotes are represented by two consecutive double quotes. Newlines are indicated with an indirect punctuation line followed by “n”.

The AG’s News dataset has been used in several studies to evaluate the performance of text classification models, including few-shot learning models (Schick & Schütze, 2020; Tunstall et al., 2022). It is a challenging dataset due to the diversity of topics covered and the variation in writing styles.

The dataset files are structured as follows:

- **classes.txt:** This file lists the classes used for news article categorization. The four classes in this dataset are World, Sports, Business, and Sci/Tech.
- **train.csv and test.csv:** These files contain the training and testing data respectively. Each record in these files includes three key components: class, title, and description. To avoid ambiguity, these components are enclosed in double quotes. Internal double quotes within the text are represented by two consecutive double quotes. Newlines within records are indicated with a newline character “\n”.

The robust structure of this dataset ensures comprehensive training and testing, facilitating effective model performance evaluation (Table 1).

Table 1. Overview of Dataset Utilized in Experiment

Name	Aspect	No. of Classes	No. of Training Samples	No. of Testing Samples
AG’s News	Topic	4	120,000	7,600

4 Proposed Work

4.1 Model

This section outlines our strategic approach to model selection, emphasizing the careful evaluation of cutting-edge transformer architectures. Leveraging the Transformers Library (Wolf et al. 2020; Brown et al., 2020) by Hugging Face, our model selection process focuses on both direct utilization and integration of pre-trained models to enhance the accuracy of our research.

Strategic Model Selection

Our methodology involves a meticulous evaluation of pre-trained models, resulting in a curated lineup of five models, each chosen for its unique characteristics. These models, detailed in Table 2, represent a convergence of computational prowess and linguistic finesse.

Table 2. Overview of Transformer Models

Model	No. of Parameters
distilbert-base-uncased (Sanh et al., 2019)	66M
google/electra-base-discriminator (Clark et al., 2020))	110M
bert-base-uncased (Devlin et al., 2019)	110M
microsoft/MiniLM-L12-H384-uncased (Wang et al., 2020c)	22M

Strategic Exclusion for Resource Optimization

We make a deliberate choice to exclude resource-intensive models like GPT-3 to streamline our research focus within computational constraints. This decision ensures that our exploration maintains depth and integrity while maximizing available resources.

Distilbert-base-uncased: Efficiency and Training Process

Distilbert-base-uncased plays a pivotal role in our model ensemble, distinguished by its modesty and swiftness in training compared to its predecessor, BERT. With 66 million parameters, this model is a compact and efficient version of BERT-base, achieved through a knowledge distillation process during training. The DistilBERT model retains 97% of BERT’s performance on various NLP tasks while being 60% faster and requiring half the memory. It consists of 6 transformer layers, each with a hidden size of 768 and 12 self-attention heads.

ELECTRA-base-discriminator: Innovative Features and Reliability

The inclusion of ELECTRA marks a paradigm shift, leveraging a Generative Adversarial Network’s (GAN) discriminator for generating “counterfeit” input tokens. Despite its smaller scale with 110 million parameters, ELECTRA achieves reliable results, particularly on datasets like Crew 2.0, showcasing its cutting-edge outcomes. The ELECTRA-base-discriminator model focuses on distinguishing between real and fake input tokens, which contributes to its effectiveness in various NLP tasks. It consists of 12 transformer layers, each with a hidden size of 768 and 12 self-attention heads.

Bert-base-uncased: Versatility and Transformer Design

Bert-base-uncased, nested within the BERT model, demonstrates versatility through its bidirectional transformer encoders. With 110 million parameters, this model is well-suited for a wide range of language processing tasks. The BERT-base-uncased variant does not consider case sensitivity, which allows for more flexibility and robustness when dealing with diverse text data. It consists of 12 transformer layers, each with a hidden size of 768 and 12 self-attention heads.

MiniLM -l12-h384-uncased Compact Design and Efficiency

MiniLM-L12-H384-uncased is a symbol of computational efficiency, with 22 million parameters. It has 12 Transformer layers, a hidden size of 384, and 12 attention heads, providing significant speedup without compromising performance.

4.2 Pre-processing

Before feeding data into the model, we needed to preprocess the input sequence and had to make the proper structure so that a pre-trained model could be fine-tuned. First, we have seen that in data exploration, we have four different classes in AG’s news dataset, which are “World News” (blue bar), “Sport News” (Yellow Bar), “Business News”(Orange bar), and “Science and Technology” (Green Bar) shown in Fig. 3.

Text normalization, a vital aspect of data preparation for natural language processing (NLP), encompasses multiple steps focused on establishing uniform, coherent, and refined text data. Below are key procedures:

- **Tokenization:** The segmentation of text into smaller units like words, phrases, or sentences. This division aids subsequent analysis by treating each unit independently.
- **Lowercasing:** Conversion of all text to lowercase to ensure consistency throughout the data. This helps avoid redundancy caused by variations in capitalization.
- **Punctuation Removal:** Eliminating punctuation marks such as periods, commas, or exclamation points. These marks typically hold limited significance in many NLP tasks, simplifying the text while preserving core content.

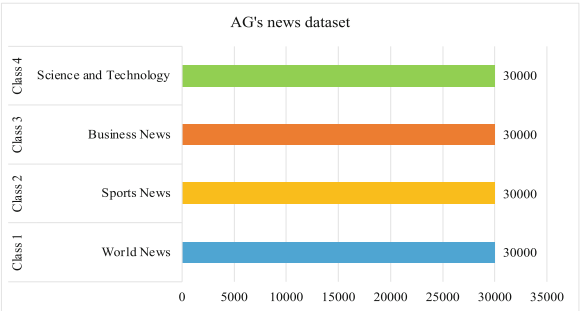


Fig. 3. An overview of the classes of the AG’s news dataset.

Stopword Elimination: Removal of common words like “and,” “the,” or “is,” which often lack substantial contextual meaning. This process streamlines focus onto more meaningful words, reducing analysis noise.

Lemmatization or Stemming: Techniques to condense words to their base or root forms. Lemmatization contextualizes and converts words to their dictionary forms (e.g., “running” to “run”). Stemming simplifies words to their roots by trimming prefixes or suffixes (e.g., “fishing” to “fish”).

Number Handling: Adapting numeric values, which may not significantly contribute to text analysis, into a standardized format for consistency and readability.

Special Character Management: Addressing symbols, emojis, or special characters that could introduce noise or processing challenges. This involves their removal or substitution with appropriate representations to ensure a cleaner dataset.

These normalization techniques enhance the structure, consistency, and applicability of text data across various NLP tasks like sentiment analysis, text classification, language modeling, and more.

Training: Given resource constraints and the absence of a validation dataset for hyperparameter tuning, we adhere to default training parameters provided by the Hugging Face library. Key parameters include learning rate schedule, number of epochs, optimizer, batch size, and gradient accumulation steps.

Automated Data Labeling: We employ few-shot classification methods for automated data labeling, aligning with the semantic richness of labels. This approach reframes the labeling challenge as a classification task, showcasing adaptability and scalability in our framework.

Clustering: Our clustering approach leverages BERT’s contextual encoding to generate distinct representations for various short forms, facilitating diverse clustering’s. Evaluation using the Silhouette Coefficient adds rigor to our methodology, assessing cluster cohesion and separation statistically.

Reliability: We rigorously test various short forms to gauge the impact of different tactics, such as punctuation removal. Normal entropy serves as a crucial evaluation metric, offering insights into variations while maintaining consistent data emphasis, ensuring reliability and robustness in our techniques.

5 Results

We evaluated the performance of several pre-trained transformer models—DistilBERT, ELECTRA-Discriminator, BERT, and MiniLM—for text classification tasks using the AG’s News dataset. The dataset was split into 80% training and 20% testing sets. Each model was fine-tuned over five epochs using the AdamW optimizer with a batch size of 16 and a learning rate of $2e-5$. The tests were repeated four times with different initializations to ensure the robustness of the results. The evaluation metrics used were Precision, Recall, and F1 scores.

5.1 Performance Comparison

Table 3 provides a unified view of the performance metrics (Precision, Recall, and F1 Score) for all models across different random initializations. The third initialization consistently showed the highest performance for each model, indicating that the choice of initialization can impact the model’s performance.

The table clearly shows that the third run for each model achieves the highest Precision, Recall, and F1 scores, demonstrating the robustness of these models in text classification tasks. This consistency across different models suggests that the third initialization might be more suitable for these types of tasks. However, it is essential to investigate the reasons behind this trend to draw more definitive conclusions.

5.2 Visual Performance Comparison

Figures. 4, 5, 6 and 7 illustrate the performance comparison of the model outputs for each initialization. These visualizations reinforce the tabulated results and highlight the consistency of the third initialization’s superior performance.

Table 3. Performance Comparison of Transformer Models Across Different Random Initializations

Model	Initialization	Precision	Recall	F1 Score
DistilBERT-Base-Uncased	1	0.92	0.91	0.92
	2	0.94	0.96	0.95
	3	0.98	0.98	0.98
	4	0.92	0.91	0.91
ELECTRA-Base-Discriminator	1	0.92	0.93	0.92
	2	0.94	0.96	0.92
	3	0.98	0.99	0.97
	4	0.92	0.91	0.90
BERT-Base-Uncased	1	0.91	0.92	0.91
	2	0.95	0.95	0.95
	3	0.98	0.98	0.98
	4	0.92	0.91	0.91
MiniLM-L12-H384-Uncased	1	0.90	0.93	0.92
	2	0.95	0.95	0.95
	3	0.98	0.98	0.98
	4	0.92	0.89	0.91

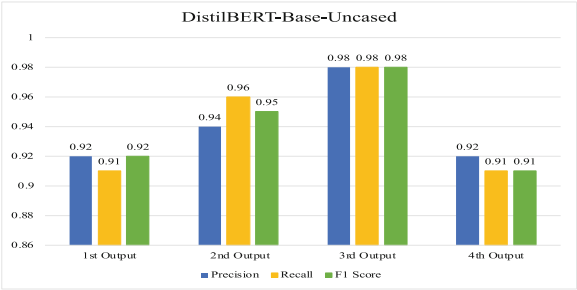


Fig. 4. Performance Comparison of DistilBERT-Base-Uncased Model Output

The visualizations provide a clear picture of how each model performs across different initializations. The consistent superior performance of the third initialization suggests that there might be specific characteristics of this initialization that make it more suitable for text classification tasks. Further investigation into these characteristics could provide valuable insights for improving the performance of transformer models in similar tasks.

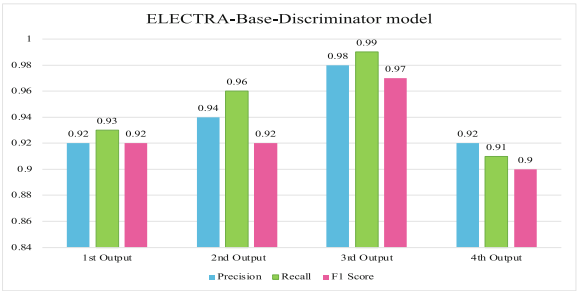


Fig. 5. Performance Comparison of ELECTRA-Base-Discriminator Model Output

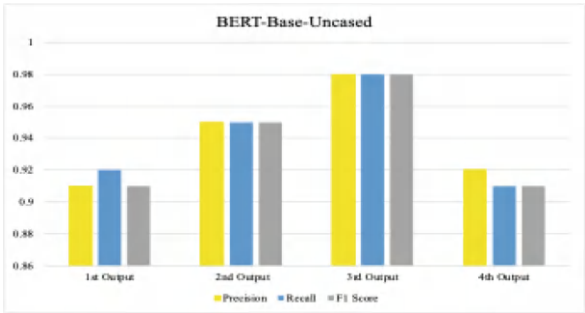


Fig. 6. Performance Comparison of BERT-Base-Uncased Model Outputs

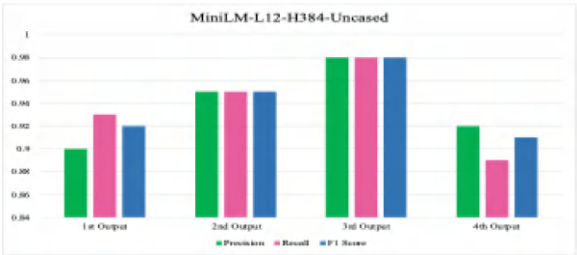


Fig. 7. Performance Comparison of MiniLM-L12-H384-Uncased Model Outputs

6 Discussion

This comprehensive discussion delves into critical aspects related to model performance, automated data labeling, practical considerations, ecological impacts, and ethical concerns surrounding language models and their applications.

Model Performance and Exploration

The outstanding performance of DistilBERT and BERT in news topic categorization is noteworthy. We delve deeper into the factors contributing to their exceptional results, such as their ability to identify intricate patterns within datasets. Furthermore, we

underscore the significance of exploring a broader range of models, including XLNet, RoBERTa, LSTM, and GRU, to obtain a more comprehensive understanding of the diverse capabilities and inherent challenges in current NLP approaches.

Automated Data Labeling and Robust Evaluation Frameworks

(Oniani et al. 2023) We critically examine the limitations and advantages of automated data labeling techniques in this discussion. By highlighting the complex considerations researchers must account for when employing these methodologies, we enhance the depth of the analysis. Moreover, we suggest specific strategies to improve framework robustness, stressing the importance of meticulous labeling for accuracy, particularly in nuanced tasks like emotion detection. The discussion emphasizes the need for extending training data and incorporating advanced models, such as GPT-3 and GPT-J, to boost performance.

Practical Challenges and Solutions

Addressing the real-world challenges of computational limitations and time constraints, we introduce practical and innovative solutions in this discussion. By citing research conducted by Smith et al. (2019) and Liang et al. (2021), we provide evidence-based methods to tackle and overcome challenges inherent in computational and time constraints, including cloud-based solutions, model pruning, and quantization.

Ecological and Ethical Considerations

Our enhanced discussion delves into the energy-intensive processes involved in training language models and explores the implications of data biases. We emphasize the importance of ethically developing and applying language models by balancing technical progress with sustainability and ethical standards.

7 Conclusion

In this study, we have explored the potential of Few-Shot Learning (FSL) in enhancing text classification within Natural Language Processing (NLP). Our findings demonstrate that FSL, when combined with pre-trained language models, can effectively address the challenge of insufficient labeled data, thus paving the way for more accessible and resource-efficient NLP solutions. The in-depth analysis of the accuracy and reliability of FSL across various models and datasets has provided valuable insights into its practical applications and limitations. By shedding light on the nuances of FSL's applicability, this research lays the foundation for future investigations and potentially guides the development of advanced, efficient NLP models.

Future Work

While this study has contributed to the understanding of FSL in the context of text classification, several avenues for further research remain open. Firstly, the limitations of FSL in complex linguistic scenarios need to be thoroughly examined, with a focus on identifying potential strategies to overcome these challenges. Secondly, the exploration of FSL should be extended to other NLP tasks, such as named entity recognition and sentiment analysis, to assess its versatility and adaptability.

Moreover, the development of interpretable machine learning (IML) techniques specifically tailored for FSL-based models is an important research direction, as it can enhance the transparency and trustworthiness of these systems. Additionally, investigating the role of FSL in multilingual and cross-lingual settings is crucial for broadening its impact and fostering language diversity in NLP.

In conclusion, this research has opened up multiple opportunities for the advancement of FSL in NLP, and we encourage the community to build upon these findings to drive the development of efficient, adaptable, and inclusive NLP solutions.

References

- Bansal, T., Jha, R., McCallum, A.: Learning to Few-Shot Learn Across Diverse Natural-Language Classification Tasks. [arXiv:1911.03863](#) (2019)
- Bansal, M.: Efficient Few-Shot Learning without Prompts (SETFIT). [arXiv:2202.05638](#) (2022)
- Brown, T. et al.: Language Models Are Few-Shot Learners. *Adv. Neural Inf. Proc. Syst.* **33**, 1877–1901 (2020)
- Chen, W.-Y., Liu, Y.-C., Kira, Z., Wang, Y.-C.F., Huang, J.-B.: A Closer Look at Few-Shot Classification. [arXiv:1904.04232](#) (2020)
- Chowdhary, K.: *Natural Language Processing*. Springer (2020)
- Clark, K., Luong, M.-T., Le, Q.V., Manning, C.D.: Electra: Pre-training text encoders as discriminators rather than generators (2020)
- de Hoop, M.V., Huang, D.Z., Qian, E., Stuart, A.M.: The Cost–Accuracy Trade-Off in Operator Learning with Neural Networks. [arXiv:2203.13181](#) (2022)
- Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding (2019)
- Gao, T., et al.: Making pre-trained language models better few-shot learners (2020)
- Gao T., Fisch A., Chen D.: Making Pre-Trained LMs Better Few-Shot Learners (LM-BFF). In: *ACL 2021*, 3816–3830 (2021)
- Han, S., Ji, R., Liao, Z., Chen, S., Liu, S.: Making pre-trained language models better few-shot learners (2021)
- Izacard, G., Grave, E., Baevski, A., Joulin, A., Grave, E.: Few-shot learning with retrieval augmented language models (2022)
- Khurana, S., Singh, G., Kaur, K.: *Natural language processing: State of the art, current trends and challenges* (2023)
- Li, S., Ma, Y., Jin, J., He, R.: Meta learning for few-shot text classification (2020)
- Li, X., Wang, T., Xiong, Y., Qian, Y., Sun, H.: Text classification with few labeled examples using self-supervised pre-training and co-training (2019)
- Liao, W., Liu, Z., Dai, H., Wu, Z., Zhang, Y., Huang, X., ... Cai, H.: Mask-Guided BERT for Few-Shot Text Classification. [arXiv:2302.10447](#) (2023)
- Min, S., Lewis, M., Hajishirzi, H., Zettlemoyer, L.: Noisy-Channel LM Prompting for Few-Shot Text Classification. [arXiv:2108.04106](#) (2021)
- Oniani, S., Sake, N., Pitsiladis, Y., Gil, D.C.: Few-shot learning for clinical natural language processing using siamese neural networks: Algorithm development and validation study (2023)
- Pan, S., Li, J.: Few-shot transfer learning for text classification with lightweight word embedding based models (2019)
- Pourpanah, F., Ayatollahi, A., Moosavi, V., Zareapoor, M., Barati, M.: A review of generalized zero-shot learning methods (2022)
- Ruder, S., Bingel, J., Augenstein, I., Sogaard, A.: What is bert really saying? Exploring bert's strengths and weaknesses through visualization (2019)

- Sanh, V., Debut, L., Chaumond, J., Wolf, T.: DistilBERT: A Distilled Version of BERT. NeurIPS EMC2 Workshop (2019)
- Schick, T., Schütze, H.: Exploiting cloze questions for few shot text classification and natural language inference (2020)
- Sung, F. et al.: Learning to Compare: Relation-Network for Few-Shot Learning. In: CVPR 2018, 1199-1208 (2018)
- Tunstall, M., Pappas, N., Xu, H., Wang, Y., Bansal, M.: Efficient few-shot learning without prompts (2022)
- Wang, A., Lu, Y., Michael, J., Hill, F., Bowman, S.R.: K-adapter: Infusing knowledge into pre-trained models with adapters (2020)
- Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., Bowman, S.R.: Generalizing from a few examples: A survey on few-shot learning (2020)
- Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., Zhou, M.: Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers (2020)
- Wang, Y., Yao, Q.: Few-Shot Learning: A Survey. ACM Comput. Surv. **53**(3), 1–37 (2019)
- Wei, J. et al.: Finetuned Language Models Are Zero-Shot Learners (FLAN). [arXiv:2109.01652](https://arxiv.org/abs/2109.01652) (2021)
- Wolf, T. et al.: Transformers: State-of-the-Art NLP. In: EMNLP-Sys 2020, 38-45 (2020)
- Yang, N. et al.: MiniLMv2: Multi-Lingual Self-Attention Distillation. [arXiv:2210.00000](https://arxiv.org/abs/2210.00000) (2022)
- Ye, Q., Lin, B.Y., Ren, X.: CROSSFIT: A Few-Shot Learning Challenge for Cross-Task Generalisation in NLP. In: EMNLP 2021, 7163-7180 (2021)
- Yin, W., Roth, D.: Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach (2019)
- Zhao, Z., Yan, H., Ma, W., Che, W., Liu, T.: Pre-trained language models can be fully zero-shot learners (2022)
- Zhang, X., Zhao, J., LeCun, Y.: Character-Level CNNs for Text Classification. In: NIPS 2015, 649-657 (2015)



A Modified NSGA-II Algorithm Using New Crowding Distance and Chaotic OBL Techniques Applied to Multi-Object Engineering Design Problems

Nastaran Ahmadi Ramezanloo¹, Yousef Sharafi²(✉), and Yashar Jebraeil²

¹ Department of Computer, Islamic Azad University, Pardis Branch, Tehran, Iran

² Department of Artificial Intelligence, Khajeh Nasir Toosi University of Technology, Tehran, Iran

y.sharafi@kntu.ac.ir

Abstract. This research introduces an advanced version of a genetic algorithm designed to tackle multi-objective optimization challenges. NSGA-II selects next-generation individuals using crowding distance and non-dominated sorting. This study makes several key contributions: firstly, it ensures the distribution of selected members on the PF by proposing a novel method to address the shortcomings of traditional crowding distance. Secondly, it introduces a hybrid mechanism combining chaotic maps and opposition-based learning (OBL) to diversify and generate the initial population. Thirdly, it presents a new approach to evaluating the cost function for selecting individuals for the next generation. Proposed algorithm benchmarked against MOPSO, MOICA, MOGWO, MOCSSO, and NSGA-II. The outcomes illustrate that the suggested approach excels in obtaining a set of optimal solutions on the PF surface while maintaining appropriate dispersion to achieve the Pareto optimal front (POF) and solve engineering multi-objective problems.

Keywords: Crowding distance · Initial population · Pareto optimal front · OBL

1 Introduction

Numerous practical applications entail multiple conflicting goals. As a result, there exists a category of problems known as MOPs, which comprise a set of optimal individuals called Pareto optimal solutions [1]. Each of these individuals represents a particular efficiency trade-off among the objectives and is considered optimal within the POF of MOEAs. Unlike single-objective optimization problems, MOPs simultaneously consider several distinct objectives. Some fundamental concepts of MOP are discussed briefly.

1.1 Multi-objective Problem

The MOP is generally expressed as detailed below [1]:

$$\text{Minimize } F(x) = (f_1(x), f_2(x), \dots, f_m(x)) \quad (1)$$

$$f_i : R^n \rightarrow R, \forall i \in (1, 2, \dots, m) \quad (2)$$

$$x = (x_1, x_2, \dots, x_n) \quad (3)$$

$$x_l \leq x_i \leq x_u$$

where n denotes problem dimensions, and m represents problem objectives, also x_l and x_u are represent the lower and upper limits of the problem variables, respectively. A MOP may have p equal constraints and k unequal constraints, as detailed below:

$$g_j(x) \geq 0 \forall j \in \{1, 2, 3, \dots, k\} \quad (4)$$

$$h_t(x) = 0 \forall t \in \{1, 2, 3, \dots, p\} \quad (5)$$

2 Literature Review

The MOEAs operate using one of the following approaches: aggregate objective function, population-based, and Pareto-based methods [1]. In the aggregate function approach as the easiest way for MOEAs, all objectives are converted to a single objective using mathematical operator and finally is solved the same as a single objective version. In the second approach, which utilizes multi-population strategies, the primary population of the algorithm is divided into sub-populations based on the number of problem objectives, with each sub-population focusing on a single objective function. The pivotal third approach focuses on identifying non-dominated solutions within the population, continuously maintaining, and updating them throughout optimization [1]. Typically, Pareto-based methods develop a smaller subset of non-dominated individuals in parallel with the main population throughout the evolutionary algorithm's search process. In recent decades, numerous variants of MOEAs have been introduced, including NSGA-II [2], MOPSO [3], MOCSO [4], MOICA [5], and MOGWO [6].

The NSGA-II algorithm relies on two fundamental concepts to rank population members: non-dominated sorting, determined by objective function values, and crowding distance (CD). These principles guide the selection of individuals for the next-generation population. Several techniques are utilized in MOEAs to rank solutions, including the Goldberg method [13], the Fonseca and Fleming method [7], the approach used in the SPEA algorithm, the ARDS ranking method that incorporates penalties [7], and the method introduced in the NSGA-II. Among these, the NSGA-II method is particularly esteemed. Various methods are presented in MOGA to obtain a diverse range of objective functions space or phenotypic, including fitness sharing methods, which reduce the

efficiency at dense points, and encourage the genetic algorithm to search in other points with less density in the objective function space. The NSGA uses the fitness sharing to diversify the solutions. The cell-based density method segments the objective function space into numerous n -dimensional cells, with the density of each cell determined by the number of solutions it holds. This technique is used in the PESA approach, where cells with less density are more probably to be selected [8–11]. The third method, used in the NSGA-II, is the CD method, which seeks to maintain a diverse array of individuals along the PF.

References [12–14] identify several critical weaknesses in the crowding distance (CD) concept, demonstrating that the CD algorithm can sometimes produce incorrect and illogical results. A significant issue arises when members of the Pareto Front (PF) exhibit similar cost function values, which severely reduces the effectiveness of CD [12]. When the density in a portion of the PF is very high, the CD values of individuals in that region drop sharply, decreasing their chances of being selected using the roulette wheel method. As a result, CD may fail to function properly in certain scenarios.

Another significant concept that has garnered substantial attention in recent years is the creation of the algorithm's initial population using chaotic maps and opposition-based learning (OBL). The effectiveness of these approaches has been demonstrated in algorithms such as MOCS, MODE, etc. The MOEAs, choose their next-generation population with two approaches. In the first approach, the evolutionary algorithm has no middle-generation population, and the current population changes it and it will be used as the next-generation population, this issue can be seen in the MOICA algorithm [5]. In the second approach, evolutionary algorithms such as NSGA-II [2] feature a middle-generation from which the upcoming generation is chosen. The population size of the middle-generation is always greater than that of the initial generation. Selection of the next-generation population is generally based on convergence criteria (non-dominated members) and diversity (high-quality members). Since MOPs involve multiple objective functions, member selection is not usually based on the direct fitness value. This study introduces a new fitness concept, derived from non-dominated sorting of population members and the new crowding distance (NCD). This concept is utilized for selecting the next-generation population in the proposed method.

The rest of this research is organized as follows: Sect. 3.1 introduces a new hybrid strategy for generating the initial population. Section 3.2 presents a novel definition of the crowding distance concept, referred to as NCD. In Sect. 3.3, a new mechanism for selecting members of the next generation population is proposed. Section 3.4 describes a method for controlling the diversity of sparsely populated areas in the Pareto Front (PF) using the k -means. The final section reports the simulation outcomes of the suggested approach on a collection of engineering design MOPs, comparing these results with other common methods.

3 The Proposed Algorithm

The proposed method is structured into four sections. In the first part, a novel method for create the initial population is introduced. The second part presents a new concept of CD to ensure the diversity of the PF members. The third sub-section offers a solution for selecting the next population based on the new fitness. The fourth sub-section proposes a method using the clustering method to address the sparsely populated areas of the PF. The final section includes the flowchart and steps of the proposed method. Figure 1 illustrates the overall procedure of the stages involved in the method algorithm.

3.1 The Proposed Initial Population

Diversification of initial population members has an important role in accelerating convergence towards global optimal solutions and preventing of early convergence for an evolutionary optimization algorithm. In the traditional method, Eq. 6 is used to generate an initial population.

$$x_{ij} = Lower_j + rand * (Upper_j - Lower_j) \tag{6}$$

Accordingly, x_{ij} is j-dimensional position of the member i in the population. Also, $Lower_j$ and $Upper_j$ are the floor and the ceiling of the search space for the j-dimension, respectively. Various methods, such as chaotic maps and opposition-based learning (OBL), have been utilized in many evolutionary algorithms (EAs) to enhance the diversity of population members.

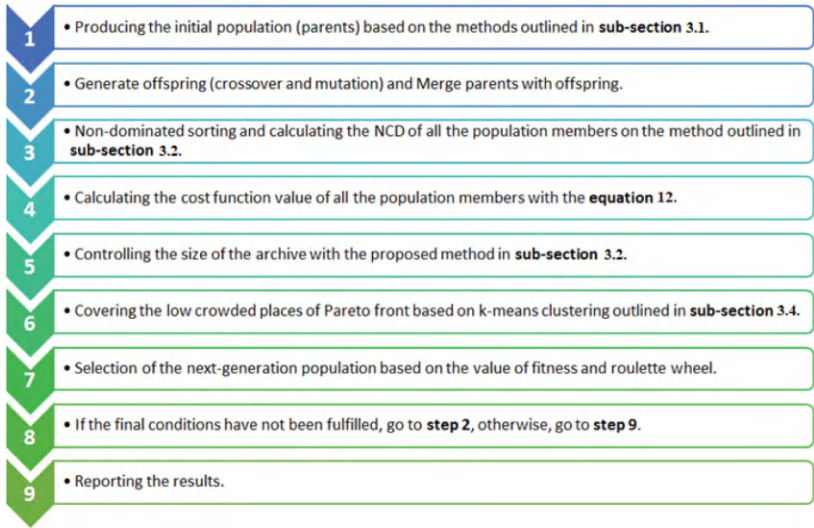


Fig. 1. The general procedure of proposed algorithm.

In the proposed method, two chaotic maps of tent and logistic are applied; Eqs. 7 and 8 show tent and logistic maps, respectively.

$$r_{Tent}(t+1) = \begin{cases} 2 \cdot \alpha \cdot r_{Tent}(t) & r_{Tent}(t) < 0.5 \\ 2 \cdot \alpha \cdot (1 - r_{Tent}(t)) & r_{Tent}(t) \geq 0.5 \end{cases} \quad (7)$$

$$r_{Logistic}(t+1) = \mu \cdot r_{Logistic}(t) \cdot (1 - r_{Logistic}(t)) \quad (8)$$

Another important technique is opposition-based learning, as shown in Eq. 9.

$$\bar{x}_{ij} = Lower_j + Upper_j - x_{ij} \quad (9)$$

Here, $Lower_j$ and $Upper_j$ are the floor and the ceiling of the search space for the j -dimension. Also, $\bar{x}_{ij}$ is considered as the mirror of x_{ij} relative to the dimension search space. In Algorithm1, proposed methods of combining chaotic and opposition-based learning concepts are provided to create the initial population of the proposed algorithm.

3.2 The Proposed Crowding Distance

The concept of CD is integral to the NSGA-II for selecting the next generation. However, the weaknesses of the CD algorithm are discussed in [12]. The CD algorithm performs effectively when the members of a Pareto Front (PF) differ significantly in their objective function values. Conversely, when the objective function values are similar, the CD value becomes an unreliable criterion. According to [12], the main cause of drawback is the sorting used in the concept of the CD. Hence, the problem is partly resolved by providing a new method for sustainable sorting. Considering Fig. 2 and assuming that the existing solutions in a particular Pareto Front (PF) resemble those illustrated in Fig. 2, a number of individuals are expected to be selected from this PF based on the CD concept. The results of the algorithm, selecting nine and five solutions for the Traditional CD, are depicted in Fig. 3 (a) and (b), respectively. In accordance with Fig. 3, the CD algorithm does not perform effectively in areas where the PF members are densely crowded, leading to the members in those areas not being selected. This is because the CD values of the members in these densely populated areas are negligible, which is a significant weakness in the crowding distance concept. In this research, this weakness is addressed by introducing a new method. The green points represent all the solutions, while the red points represent the solutions selected by an algorithm.

Algorithm 1 The Hybrid Approach for Initialization Population

```

1. //Chaotic and Uniform Search
2. t=1
3.  $r_{Tent}(1)=rand, r_{Logistic}(t)=rand$ 
4. for  $i:=1$  to  $Pop\_size$  do
5.   for  $j:=1$  to  $Dimension\_size$  do
6.      $x_{ij} = Lower_j + rand \cdot (Upper_j - Lower_j)$  // Uniform Search
7.      $y_{ij} = Lower_j + r_{Tent}(t) \cdot (Upper_j - Lower_j)$  // Chaotic Tent Search
8.      $z_{ij} = Lower_j + r_{Logistic}(t) \cdot (Upper_j - Lower_j)$  // Chaotic Logistic
9.   Search
10.  Increase  $t$ 
11.  end for;
12. end for;
13. // Opposition based learning
14. for  $i:=1$  to  $Pop\_size$  do
15.   for  $j:=1$  to  $Dimension\_size$  do
16.      $\bar{x}_{ij} = Lower_j + Upper_j - x_{ij}$  // Uniform & Opposition
17.      $\bar{y}_{ij} = Lower_j + Upper_j - y_{ij}$  // Chaotic Tent & Opposition
18.      $\bar{z}_{ij} = Lower_j + Upper_j - z_{ij}$  // Chaotic Logistic & Opposition
19.   end for;
20. end for;
21. //Integration of populations
22.  $Temp = \{x, y, z, \bar{x}, \bar{y}, \bar{z}\}$  //Greedy Selection
23. // Selecting better members for the initial population

```

To resolve the fault of the crowding distance algorithm, its equation is corrected as follows:

$$NCD_i = \frac{(f_i^1 - f_{min}^1) \cdot (f_{max}^2 - f_i^2) + (f_i^2 - f_{min}^2) \cdot (f_{max}^1 - f_i^1)}{(f_{max}^1 - f_{min}^1) \cdot (f_{max}^2 - f_{min}^2)} \quad (10)$$

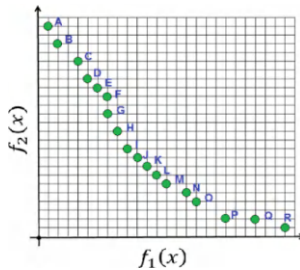


Fig. 2. Pareto front with two objective functions.

Accordingly, as illustrated in Fig. 4, f_{min}^k and f_{max}^k represent the lower and upper limits of the k -th objective. The result of the suggested approach for the example in Fig. 2, to choose nine and five solutions is illustrated in Fig. 5-(a) and Fig. 5-(b), respectively. As

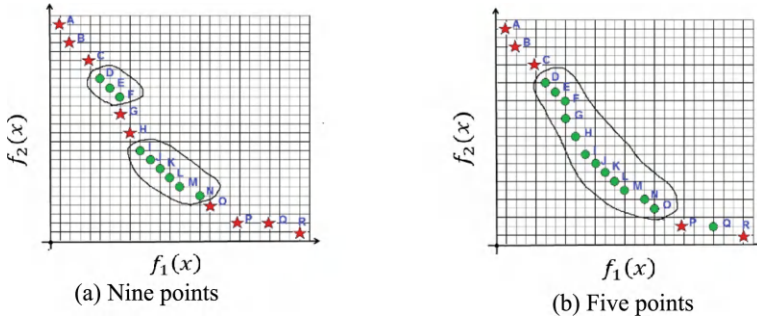


Fig. 3. Selection of some solutions from the PF based on the CD.

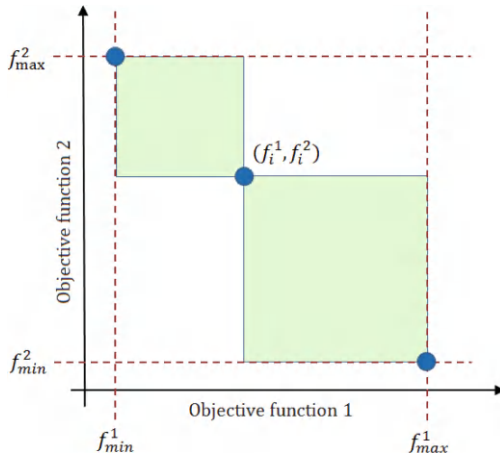


Fig. 4. Proposed method to modify the concept of crowding distance.

the findings show, the diversity of the selected answers is much more regular than the traditional crowding distance method.

3.3 The Proposed Fitness Value

In the NSGA-II, the selection of next-generation members is based on the concepts of crowding distance (CD) and non-dominated sorting, as illustrated in Fig. 6. The next-generation members are not chosen directly based-on the new fitness value.

In this paper, the choosing of next-generation members is done based on the cost function value (fitness), which is defined by the following equation:

$$fitness_i = (Rank(i) * m) - NCD(i) \quad (11)$$

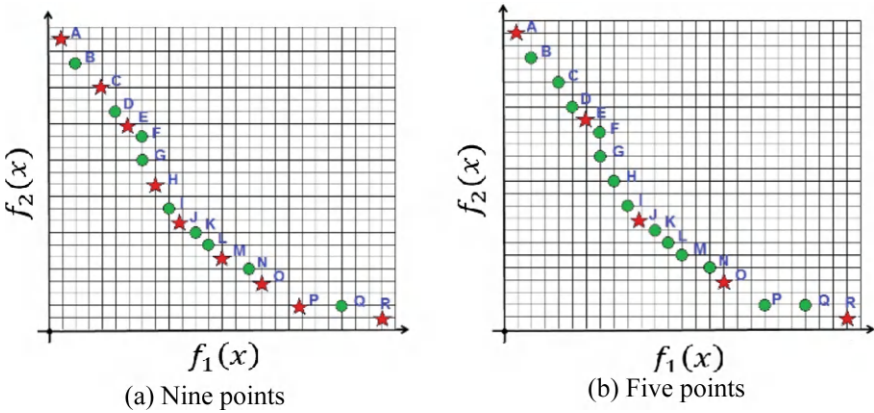


Fig. 5. Selection of several solutions from the PF based on the proposed method.

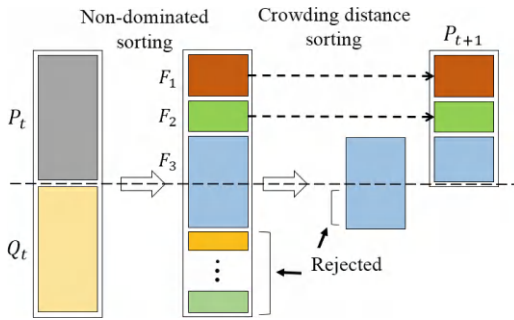


Fig. 6. The loop of NSGA-II procedure.

The normalization of the cost function value by the following equation is carried out:

$$NewFitness_i = \frac{fitness_i}{\sum_{i=1}^N fitness_i} \tag{12}$$

3.4 The Coverage of Low Crowded Places in PF Based on k-means Algorithm

The diversity control of the PF members is very important in MOPs, and on the other hand, in most of the EAs, there is no mechanism for precise observation of low crowded places in PF. In this study, based on the clustering algorithm, a mechanism is presented to control the diversity of low crowded places in PF; so that the PF space is divided into k clusters as shown in Fig. 7. The clusters with more members usually have less diversity. The cluster with the least number of population members will be selected, and then, a mutation is performed in the members of that cluster, twice as large as the number of

population members for that cluster, and their non-dominated population is incorporated into the external archive population.

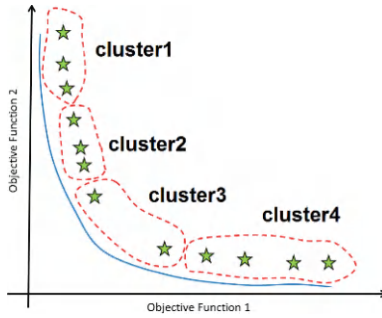


Fig. 7. The clustering of members in the Pareto front based on k-means algorithm.

Based on the concepts mentioned in Sects. 3.1 to 3.4, a new procedure for MOGA is presented. According to the Fig. 6, after ranking the members of the population, and calculating the new crowding distance and evaluating the cost function value on the basis of Eq. (12), members of the next-generation population will be selected like the pure GA algorithm.

4 The Experimental Results

In this part, a set of MOPs are utilized to evaluate the performance of the suggested NMOGA. NMOGA's performance is compared to other MOEAs, including MOPSO [3], MOICA [5], NSGA-II [2], MOGWO [6], and MOCSSO [4]. All computations are executed using MATLAB R2021a on a computer with a Core i-5 CPU and 4-GB of RAM. To evaluate NMOGA's performance, various MOPs are tested. In our trials, we utilized 500 evenly distributed Pareto optimal solutions to approximate the POF. The performance metrics are summarized as follows:

- IGD metric [8].
- HV metric [8].
- Δ metric [4].

The default parameter configurations for all algorithms are detailed in Table 1.

This subsection includes three well-known MOPs instances utilized for performance evaluation of the proposed NMOGA algorithm.

4.1 Four-Bar Truss Design

The goal is to minimize both the truss volume and joint displacement by treating link areas as variables in the design problem (Fig. 8).

$$\text{Minimize } f_1(x) = L(2x_1 + \sqrt{2}x_2 + \sqrt{x_3} + x_4) \quad (13)$$

Table 1. Default parameters settings.

Algorithm	Parameter	Value
NMOGA & NSGA-II	crossover probability	0.9
	mutation probability	1/dimension
MOPSO	c_1	2.05
	c_2	2.05
	w	0.9 to 0.1
MOCSSO	SMP	10-15
	SPC	True
	CDC	0.4
	SRD	0.25
	c_1	2.05
	w	0.9 to 0.1
MOGWO	MR	0.05
	No Parameter	

Assimilation Coefficient = 2.0
Revolution Probability = 0.1
Probability of Revolution on a Specific Variable = 0.15
Colonies Mean Cost Coefficient = 0.1
Initialization Selection Pressure = 1.0
Inter-Empire Competition Selection Pressure = 1.0

$$Minimize f_2(x) = \frac{F \cdot L}{E} \left(\frac{2}{x_1} + \frac{2\sqrt{2}}{x_2} - \frac{2\sqrt{2}}{x_3} + \frac{2}{x_4} \right) \tag{14}$$

where

$$\begin{aligned} (F/\sigma) \leq x_1 \leq 3(F/\sigma), \sqrt{2}(F/\sigma) \leq x_2 \leq 3(F/\sigma) \\ \sqrt{2}(F/\sigma) \leq x_3 \leq 3(F/\sigma), (F/\sigma) \leq x_4 \leq 3(F/\sigma) \end{aligned} \tag{15}$$

$F = 10 \text{ kN}, E = 2 \times 10^5 \text{ kN/cm}^2, L = 200 \text{ cm}, \sigma = 10 \text{ kN/cm}^2$

4.2 Multi-plate Disk Brake Design

This problem aims to optimize braking efficiency for airplanes during landing. Depicted in Fig. 9.

$$Minimize f_1(x) = 4.9 \times 10^{-5} (x_2^2 - x_1^2) (x_4 - 1) \tag{16}$$

$$Minimize f_2(x) = \frac{9.82 \times 10^6 (x_2^2 - x_1^2)}{x_3 x_4 (x_2^3 - x_1^3)} \tag{17}$$

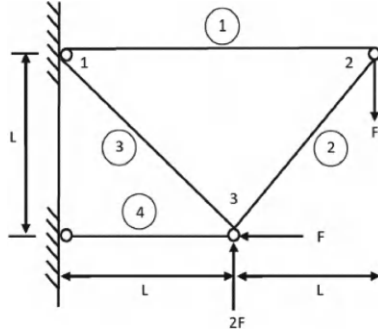


Fig. 8. Design of four bar truss problem.

subject to

$$g_1(x) = 20 + x_1 - x_2 \leq 0$$

$$g_2(x) = 2.5(x_4 + 1) - 30 \leq 0$$

$$g_3(x) = \frac{x_3}{3.14(x_2^2 - x_1^2)^2} - 0.4 \leq 0 \quad (18)$$

$$g_4(x) = \frac{2.22 \times 10^{-3} x_3 (x_2^3 - x_1^3)}{(x_2^2 - x_1^2)^2} - 1 \leq 0$$

$$g_5(x) = 900 - \frac{2.66 \times 10^{-2} x_3 x_4 (x_2^3 - x_1^3)}{(x_2^2 - x_1^2)} \leq 0$$

where

$$55 \leq x_1 \leq 80, 75 \leq x_2 \leq 110, 10^3 \leq x_3 \leq 3 \times 10^3, 2 \leq x_4 \leq 20 \quad (19)$$

Table 2 summarizes mean and standard deviation for truss and brake optimization. These results were obtained using six different algorithms, each run independently 20 times. Statistical contrasts between NMOGA and other algorithms are emphasized. Table 2 indicates NMOGA's superior performance in mean and standard deviation. Figure 10 graphically compares NMOGA with MOCSO, MOICA, MOGWO, NSGA-II, and MOPSO (11).

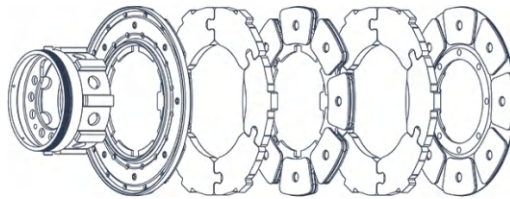


Fig. 9. Multi-plate disk brake.

Table 2 presents the mean and variability of best solution metrics for the problems, evaluated using six algorithms across 20 runs. NMOGA demonstrates superior performance in this analysis.

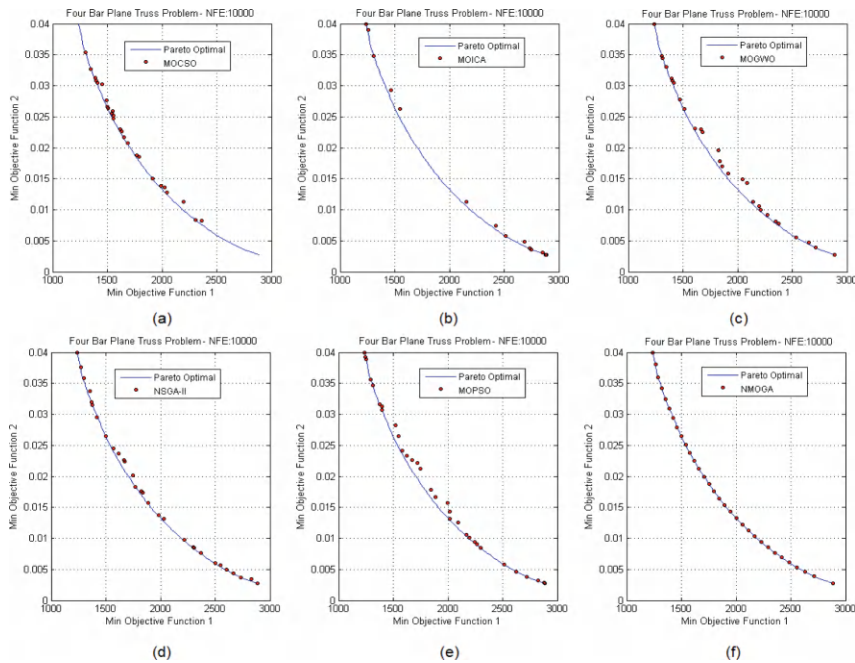


Fig. 10. Distribution of solutions for the four-bar problem across different algorithms.

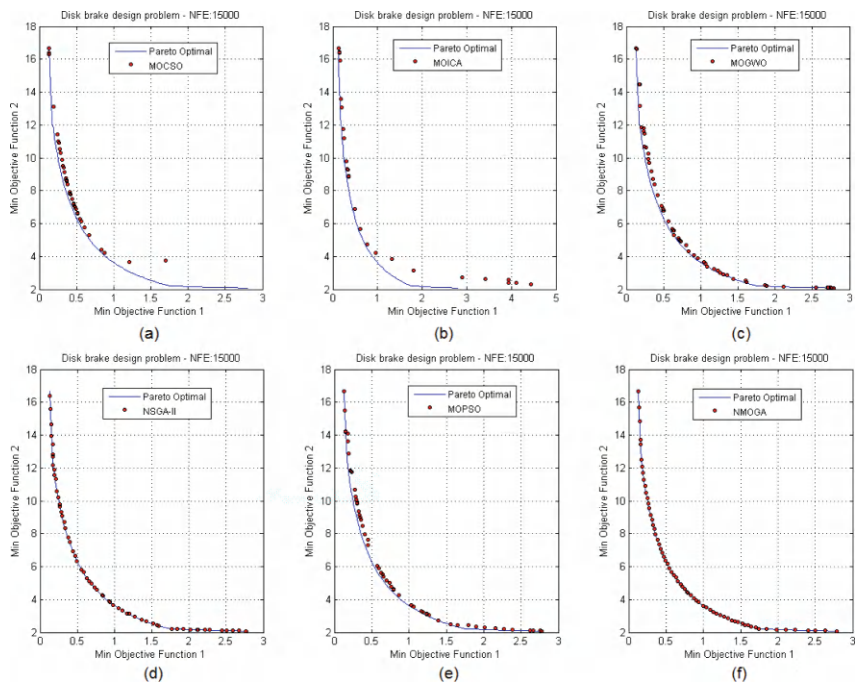


Fig. 11. Distribution solutions for the multi-plate disk problem across different algorithms.

Table 2. NMOGA versus other algorithms: Mean and SD comparison.

#Problem	#Metric	NSGA-II	NMOGA	MOPSO
<i>Four-bar truss</i>	<i>GD</i>	1.4007 ± 0.3834	0.5193 ± 0.0812	1.7162 ± 0.1148
	<i>S</i>	38.0938 ± 0.8417	26.1133 ± 0.2786	44.0788 ± 0.9115
	Δ	0.5483 ± 0.0127	0.3023 ± 0.0013	0.6589 ± 0.0585
<i>Multi-plate disk brake</i>	<i>GD</i>	0.0255 ± 0.0026	0.0157 ± 0.0002	0.0304 ± 0.0017
	<i>S</i>	0.2201 ± 0.0123	0.2010 ± 0.0095	0.3042 ± 0.0078
	Δ	0.6904 ± 0.0215	0.577 ± 0.0128	0.8671 ± 0.0058
#Problem	#Metric	MOCSSO	MOICA	MOGWO
<i>Four-bar truss</i>	<i>GD</i>	1.2983 ± 0.0853	1.464 ± 0.5493	1.475 ± 0.3945
	<i>S</i>	36.122 ± 4.9431	114.5253 ± 22.2738	49.6612 ± 5.6994
	Δ	0.8141 ± 0.0953	1.2005 ± 0.1497	0.7189 ± 0.062
<i>Multi-plate disk brake</i>	<i>GD</i>	0.0174 ± 0.0012	0.0641 ± 0.0221	0.0269 ± 0.0044
	<i>S</i>	0.5827 ± 0.1706	0.6672 ± 0.1359	0.3651 ± 0.0057
	Δ	1.177 ± 0.0757	1.0635 ± 0.104	0.9394 ± 0.0302

5 Conclusion

In this research, a new version of MOGA is presented based-on new crowding distance and chaotic opposition-based learning to solve MOPs. The main study contributions include: Initially, ensuring the distribution of selected members on the Pareto Front (PF) is achieved through the introduction of a novel method that addresses shortcomings inherent in traditional crowding distance approaches. In the second contribution, a new hybrid method uses chaotic maps and opposition-based learning to diversify the initial population in varied search spaces. In the third contribution a new concept of the cost function is presented to select members of the next generation population. The fourth contribution is that the coverage of low crowded is located in PF based on k-means algorithm. The outcomes of implementing our algorithm are contrasted with other MOEAs such as MOPSO, MOICA, MOGWO, MOCSSO, and NSGA-II. Our approach demonstrates superior capability in attaining an optimal solution set across the Pareto front, ensuring adequate dispersion for solving engineering multi-objective problems (MOPs).

References

1. Zhang, W., Xiao, G., Gen, M., Geng, H., Wang, X., Zhang, G.: Enhancing multi-objective evolutionary algorithms with machine learning for scheduling problems: recent advances and survey. *Front. Indust. Eng.* **2**, 1337174 (2024)
2. Ma, H., Zhang, Y., Sun, S., Liu, T., Shan, Y.: A comprehensive survey on NSGA-II for multi-objective optimization and applications. *Artific. Intell. Rev.* 1–54 (2023)
3. Han, H., Liu, Y., Hou, Y., Qiao, J.: Multi-modal multi-objective particle swarm optimization with self-adjusting strategy. *Inf. Sci.* **629**, 580–598 (2023)

4. Ahmed, A.M., et al.: GMOCSSO: Multi-objective Cat Swarm Optimization Algorithm based on a Grid System (2023)
5. Enayatifar, R., Yousefi, M., Abdullah, A.H., Darus, A.N.: MOICA: a novel multi-objective approach based on imperialist competitive algorithm. *Appl. Math. Comput.* **219**(17), 8829–8841 (2013)
6. Makhadmeh, S.N., Alomari, O.A., Mirjalili, S., Al-Betar, M.A., Elnagar, A.: Recent advances in multi-objective grey wolf optimizer, its versions and applications. *Neural Comput. Appl.* **34**(22), 19723–19749 (2022)
7. Gaikwad, R., Lakshmanan, R.: Study of evolutionary algorithms for multi-objective optimization. *SN Comput. Sci.* **3**(5), 409 (2022)
8. Emmerich, M.T., Deutz, A.H.: A tutorial on multiobjective optimization: fundamentals and evolutionary methods. *Nat. Comput.* **17**, 585–609 (2018)
9. Kumar, L., Kumar, P.: Multiobjective Based LQR Design for Fractional Order Systems with Perturbations using PESA-II Algorithm. *arXiv preprint [arXiv:2004.10982](https://arxiv.org/abs/2004.10982)* (2020)
10. Yen, G.G., Lu, H.: Dynamic multiobjective evolutionary algorithm: adaptive cell-based rank and density estimation. *IEEE Trans. Evol. Comput.* **7**(3), 253–274 (2003)
11. Wang, Z., Liu, W., Yang, M., Han, D.: A multi-objective evolutionary algorithm model for product form design based on improved SPEA2. *Appl. Sci.* **9**(14), 2944 (2019)
12. Fortin, F.A., Parizéau, M.: Revisiting the NSGA-II crowding-distance computation. In: *Proceedings of the 15th Annual conference on Genetic and Evolutionary Computation*, pp. 623–630 (2013)
13. Lambora, A., Gupta, K., Chopra, K.: Genetic algorithm-A literature review. In: *2019 international conference on machine learning, big data, cloud and parallel computing*, pp. 380–384. IEEE (2019)
14. Shi, J., Gong, M., Ma, W., Jiao, L.: A multipopulation coevolutionary strategy for multi-objective immune algorithm. *Sci. World J.* **2014**, 1–23 (2014). <https://doi.org/10.1155/2014/539128>

Author Index

A

Abbas, Syed Hur 366
Agnihotri, Alka 177, 300
Annamalai, Suresh 45, 73
Antony, Joyal 343
Ayyalasomayajula, Madan Mohan Tito 62

B

Bade-Gite, Shilpa 177
Bajpai, Ayush 221
Balusamy, Balamurugan 45, 73, 177, 300
Bapat, Mayuri 114
Bhagat, Janhavi 15
Bobby, George 343
Borhade, Piyush 87

C

Chandy, Mishbah Pinto 21
Chaugule, Sushma V. 3
Chintale, Pradeep 207, 234, 249, 261

D

Darak, M. S. 195
Dasari, Vineet 33
Deshmukh, Hrushikesh 207, 249
Deshmukh, N. K. 195
Divya Rani, R. 164

F

Farjana, U. 154

G

Garg, Ritu 221
Garud, Urjita 87
Gupta, Prateek 140

H

Hambarde, K. A. 195
Hegadi, Ravindra S. 331
Houde, Kavita V. 331

Hummel, Sandra 366

J

Jadhav, Sharad B. 195
Jamal, Arshi 101
Jayapandian, N. 21, 343
Jebracily, Yashar 381
Joshi, Jowin 343
Joshi, Sanskruti S. 15

K

Kambayal, Vrashabh 275
Kamble, Rutuja 87
Karade, Shreya 114
Khanna, Anirudh 207, 234, 249
Khari, M. 127
Korada, Laxminarayana 261
Kulkarni, Ambarish 15
Kumar, C. Madhu 33
Kumar, K. 127

L

Lankadas, Jaswanth Singh Kumar 315
Lankadasu, Naga Venkata Yaswanth 33

M

Madgi, Manohar 275
Mahida, Ankur 249, 261
Malemath, V. S. 3
Mali, S. M. 15, 87
Manogaran, Nalini 45, 73
Mishra, Ashutosh 331
Mmihiraansh, 33

N

Nagane, Aniket S. 114
Nagavibha, R. 21
Nagubandi, Veera Venkata Varshith 315
Nair, Resmi R. 45
Najana, Madhavi 207, 234, 249, 261

Natesan, Deepa 177, 300
 Norbu, Tsewang 275

P

Padmanabhuni, Sri Silpa 286
 Pandey, P. 127
 Patil, Shivanand 3
 Perumal, Arun Pandiyan 234, 261
 Pesarlanka, Devendra Babu 33
 Prabhakar, C. J. 164
 Prasad, D. Durga 286
 Prasad, Vejandla Pavithra 286

R

Rai, Prabhat K. 275
 Ramesh, K. 101
 Ramezanloo, Nastaran Ahmadi 381
 Reddy, Dappili Bharat Kumar 33
 Reddy, Satti Satya 315
 Romisha, 356

S

Sahoo, Deepali Rani 300
 Sai, Gowrineni Harshitha 154
 Saisanthiya, D. 300
 Sanne, Sri Harsha Vardhan 261
 Satheesh, Mohith Manoharan 21
 Seerangan, Koteeswaran 73

Sharafi, Yousef 381
 Sharma, Ajay 33, 315
 Sharma, Shambhawi 15
 Sharma, Shamneesh 33, 315
 Singh, P. P. 127
 Siri, Perumalla Prasanna 286
 Sushama 177, 300

T

Tabbassum, Ayisha 234
 Tanwar, Sushama 45, 73
 Thakur, Gopal Kumar 62
 Tiwari, Meena 140
 Trisha, Singh 315

V

Vashist, Shruti 356
 Veluvali, Parimala 45, 73
 Verma, Vivek 275
 Victoria, Helen 177
 Vineetha, Repalli 286

W

Wadhwa, Gitanjali 366

Y

Yadav, A. 127