

Shailesh Tiwari  
Munesh C. Trivedi  
Mohan L. Kolhe  
Brajesh Kumar Singh *Editors*

# Advances in Data and Information Sciences

Proceedings of ICDIS 2024, Volume 2

# **Lecture Notes in Networks and Systems**

**Volume 1193**

## **Series Editor**

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

## **Advisory Editors**

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

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

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

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

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

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

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

Imre J. Rudas, Óbuda University, Budapest, Hungary

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

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

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

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

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

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

All books published in the series are submitted for consideration in Web of Science.  
For proposals from Asia please contact Aninda Bose ([aninda.bose@springer.com](mailto:aninda.bose@springer.com)).

Shailesh Tiwari · Munesh C. Trivedi ·  
Mohan L. Kolhe · Brajesh Kumar Singh  
Editors

# Advances in Data and Information Sciences

Proceedings of ICDIS 2024, Volume 2



Springer

*Editors*

Shailesh Tiwari  
SRM University  
Sonepat, Haryana, India

Mohan L. Kolhe  
Faculty of Engineering and Science  
University of Agder  
Kristiansand, Norway

Munesh C. Trivedi  
Department of Engineering and Technology  
PSS Central Institute of Vocational  
Education (PSSCIVE)

Bhopal, Madhya Pradesh, India

Brajesh Kumar Singh  
Department of Computer Science  
and Engineering  
R. B. S. Engineering Technical Campus  
Agra, Uttar Pradesh, India

ISSN 2367-3370

ISSN 2367-3389 (electronic)

Lecture Notes in Networks and Systems

ISBN 978-981-97-9618-2

ISBN 978-981-97-9619-9 (eBook)

<https://doi.org/10.1007/978-981-97-9619-9>

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature  
Singapore Pte Ltd. 2025

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

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

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

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

If disposing of this product, please recycle the paper.

# Preface

The ICDIS is a major multidisciplinary conference organized with the objective of bringing together researchers, developers and practitioners from academia and industry working in all areas of computer and computational sciences. It is organized specifically to help computer industry to derive the advances of next generation computer and communication technology. Researchers invited to speak, will present the latest developments and technical solutions.

Technological developments all over the world are dependent upon globalization of various research activities. Exchange of information, innovative ideas is necessary to accelerate the development of technology. Keeping this ideology in preference, the 6th International Conference on Data and Information Sciences (ICDIS-2024) has been organized at Raja Balwant Singh Engineering Technical Campus, Bichpuri, Agra, India during May 24–25, 2024.

The 6th International Conference on Data and Information Sciences has been organized with a foreseen objective of enhancing the research activities at a large scale. Technical Program Committee and Advisory Board of ICDIS-2024 include eminent academicians, researchers and practitioners from abroad as well as from all over the nation.

A sincere effort has been made to make it an immense source of knowledge by including 72 manuscripts in the proceedings volumes of ICDIS-2024. The selected manuscripts have gone through a rigorous review process and are revised by authors after incorporating the suggestions of the reviewers.

ICDIS-2024 received around 350 submissions from around 605 authors of different countries such as China, USA, Norway, Russia, Vietnam, and Bulgaria. Each submission has been gone through the similarity check. On the basis of similarity report, each submission has been rigorously reviewed by atleast two reviewers. Even some submissions have more than two reviews. On the basis of these reviews, 72 high quality papers were selected for publication in two proceedings volumes, with an acceptance rate of 20.5%.

We are thankful to the keynote speakers- Prof. Valentina E. Balas, University of Arad, Romania and Prof. B. K. Panigrahi, IIT Delhi, India to enlighten the participants with their knowledge and insights. We are also thankful to delegates and the authors

for their participation and their interest in ICDIS-2024 as a platform to share their ideas and innovation. We are also thankful to the Prof. Janusz Kacprzyk, Series Editor, LNNS, Springer Nature and Mr. Aninda Bose, Executive Editor, Springer Nature, India for providing guidance and support. Also, we extend our heartfelt gratitude to the reviewers and Technical Program Committee Members for showing their concern and efforts in the review process. We are indeed thankful to everyone directly or indirectly associated with the conference organizing team leading it towards the success.

Although utmost care has been taken in compilation and editing, however, a few errors may still occur. We request the participants to bear with such errors and lapses (if any). We wish you all the best.

Agra, India

Shailesh Tiwari  
Munesh C. Trivedi  
Mohan L. Kolhe  
Brajesh Kumar Singh

# Contents

<b>An Automated Early Lung Cancer Detection Using Convolution Neural Network .....</b>	1
Santosh Kumar Satapathy, Manan Gandhi, Vanshita Patel, Jui Mehta, Devam Patel, and Rishi Joshi	
<b>Empowering Robust Speech Emotion Recognition Using Deep Neural Network .....</b>	13
Muthukuru Jayanth, Saravanan Palani, and M. Marimuthu	
<b>DeepLeaf: A Custom CNN Approach for Mulberry Leaf Classification .....</b>	25
Tripti Mishra, Vanshaj Singhal, Yashaswat Verma, Monika, and Manish Raj	
<b>Detecting Polycystic Ovary Syndrome Through Blending Ensemble Method .....</b>	35
Kashish Gandhi, Mansi Prajapati, Dev Bhut, and Ruhina Karani	
<b>Verbatim: Empowering Seamless Communication with Authentic Voice Translation .....</b>	55
C. I. Chandas Patel, Swimpy Pahuja, Rutika Babasab Patil, R. Yeswas, Arati Chabukswar, and M. S. Pratap	
<b>Facial Emotion Detection Using Artificial Intelligence .....</b>	67
Ananya Debnath, Vineet Singh, Bramah Hazela, and Shikha Singh	
<b>News-Scope: Intelligent Categorization of News Content Using Machine Learning .....</b>	79
Rahul Karmakar, Mrinal Manna, Sidhartha Bakuli, Rajayshree Bhattacharyaa, and Avijit Das	
<b>Brain Tumor Detection Using MRI and Deep Learning Techniques .....</b>	91
Kajal Singh, Vineet Singh, Bramah Hazela, and Shikha Singh	

<b>Cyberbullying Trends in Indian Sports—A Sentiment Analysis of Twitter Feeds .....</b>	103
Durga Sharma and Rahul Johari	
<b>AI-Based Adaptive Legal Analysis Engine for Enhanced Policing and Predictive Law Enforcement .....</b>	119
Nagendra Singh, Abhishek Tiwari, Ruchi Tiwari, Priyanka Tiwari, Chaitanya Pushkarna, and Jitesh Choudhary	
<b>AI-Based Data Analytics &amp; Business Intelligence Chatbot Using Azure Functions and OpenAI .....</b>	129
N. Praveen Sundra Kumar, S. Ramakrishnan, and M. Vignesh	
<b>Assistive Live Audio Transcription Glasses for Individuals Suffering Auditory Impairment .....</b>	139
Siddharth Menon, Aparna Padma Balaji, Jayant Sasikumar, Thazhai Mugunthan, V. Ravikumar Pandi, Soumya Sathyan, Vipina Valsan, and Kavya Suresh	
<b>Graph-Based Predictive Modeling in Drug Response .....</b>	153
T. P. Athulya Valsan and Anuraj Mohan	
<b>Leveraging MaxEnt and TF-IDF Trigrams Against Fake News .....</b>	165
S. Siji Rani, Gade Sai Panshul, Tathipamula Harini Sai, Lingutla Prem Kumar, and Hareendra Sri Nag Nerusu	
<b>VR Phantom Haven: Phantom Limb Pain Management Using Virtual Reality .....</b>	179
Aditya Shah, Siddhi Muni, Gautam Mehendale, and Chetashri Bhadane	
<b>Beyond the Surface: Exploring Segmentation Techniques in DL for Early Brain Tumor Detection .....</b>	189
Soni Singh, Pratyush Mishra, Md. Kaish, Jordan-Kény Gnansounou Dansi, Sunaina Singh, Johnstone Joel Ngorma, and Sahla Ambrein	
<b>Salary Prediction Using Machine Learning Techniques .....</b>	201
Pijush Ghorai and Rupashri Barik	
<b>Tax Technology as a Catalyst for Globalization of Companies and Digital Transformation .....</b>	215
Zornitsa Yordanova	
<b>Sightless Fashion: Deep Learning Shopping Solutions .....</b>	227
Clara Joseph and Sruthy Manmadhan	
<b>Handwritten Signature Verification and Forgery Detection using Deep Learning .....</b>	239
Harsh Vardhan, Gaurav Kumar Gautam, Harshit Gupta, and Rahul Katarya	

<b>Exploring Methodologies for Computing Sentence Similarity in Natural Language Processing .....</b>	251
Sagar Mondal, Abirami Gurushanker, Mirudhula Loganath, Rishima Chowdhury, Sankari Karthik, Lekshmi Kalinathan, Janaki Meena Murugan, Marimuthu Marimuthu, and Saravanan Palani	
<b>Enhancing Real-Time Gesture Recognition Systems for Virtual Reality Applications Using Deep Learning Techniques .....</b>	263
Rahul Kumar, Lekshmi Kalinathan, and Janaki Meena Murugan	
<b>Machine Learning for Power Analysis: A New Paradigm in CMOS VLSI Design .....</b>	273
Naiyya Mittal, Srishty Sharma, Tithi Pandey, and Shobha Sharma	
<b>An Ensemble-Based Lexicon Dictionary Coupled with Annotated Fine-Grained Emotions and Sentiments .....</b>	285
Shelley Gupta and Archana Singh	
<b>Netflix Analysis Using Tableau and ML .....</b>	297
E. Elakiya, Leki Chom Thungon, Benoy Joseph, and Manas Kamal Das	
<b>Acoustic Monitoring of Biodiversity .....</b>	319
Aniket Kumar, Swati Kale, Amey Jojare, and Siddesh Sabade	
<b>Design and Implementation of Real-Time Environment Tracking System Using Internet of Things (IoT) .....</b>	333
Ajeet Singh, Sanjay Singh, and Rupali Mahajan	
<b>Automatic Weed Detection Using CNN .....</b>	345
Shubham Kumar Gupta, Sarthak Agarwal, Yash Garg, and Dilkeshwar Pandey	
<b>Height Measurement of Pose Bent Knees by Using Pose Estimation of MediaPipe .....</b>	357
Nguyen Phan Kien, Hoang Van Thao, Duc-Tan Tran, and Vijender Kumar Solanki	
<b>Non-contact Height Measurement in 2D with the Pose of 45° Side Standing .....</b>	367
Nguyen Phan Kien, Dong Quoc Dat, Duc-Tan Tran, and Vijender Kumar Solanki	
<b>A Navigation Tracking Line Algorithm for the Mobile Robot Based on Traditional Vision .....</b>	377
Khoa Nguyen Dang, Pham Tuan Minh, Duc-Tan Tran, and Vijender Kumar Solanki	
<b>A Systematic Literature Review on Lung Cancer with Ensemble Learning .....</b>	389
Fahum Nufikha Jahan, Shakik Mahmud, and Md Kamrul Siam	

<b>A Study on Privacy-Preserving Multiparty Computation Protocols . . . . .</b>	<b>399</b>
Chinmaya Bikram Pattanaik, Munesh Chandra Trivedi, Ruchi Jain, and Mohan Lal Kolhe	
<b>Intelligent Object Detection for Visually Impaired People Using YOLO Algorithm . . . . .</b>	<b>411</b>
Anil Kumar Dubey, Sejal Maheshwari, Swapnika Agrawal, and Mohan Lal Kolhe	
<b>Performance Analysis of Intelligent Surveillance System in a Fog Computing Environment . . . . .</b>	<b>425</b>
Pradeep Singh Rawat, Prateek Kumar Soni, and Punit Gupta	
<b>Rash Driving Detection Using IoT and ML . . . . .</b>	<b>437</b>
Arnaav Anand, Ishita Mehta, and Punit Gupta	
<b>Author Index . . . . .</b>	<b>453</b>

## About the Editors

**Prof. Shailesh Tiwari Ph.D.** currently works as a Professor and Pro-Vice Chancellor at SRM University, Sonepat, Haryana, India. He is an alumnus of Motilal Nehru National Institute of Technology Allahabad, India. His primary areas of research are software testing, implementation of optimization algorithms and machine learning techniques in various engineering problems. He has published more than 100 publications in International Journals and in Proceedings of International Conferences of repute. He has edited special issues of several Scopus, SCI and E-SCI-indexed journals. He has also edited several books published by Springer. He has published 6 Indian patents as IPRs. He has organized several international conferences under the banner of IEEE, ACM and Springer. He is a Senior Member of IEEE, member of IEEE Computer Society, Fellow of Institution of Engineers (FIE).

**Dr. Munesh C. Trivedi** has more than 20 years of experience in the field of Computer Science & Engineering and has worked in Prestigious Institutions. Currently working as Professor (Computer Science), PSSCIVE Bhopal (A Constituent unit of NCERT, Under Ministry of Education, Government of India). He had successfully filed 65 patents (52 National and 13 International Patents (Germany, South Africa, and Australia)), out of which 35 patents were granted. He has published 12 textbooks and 153 research papers in different International Journals and Proceedings of repute. He has also edited 38 books for Springer Nature. He successfully supervised 14 Ph.D. students and received numerous awards, including the Young Scientist Visiting Fellowship, Albert Einstein Research Scientist Award, Best Senior Faculty Award, Outstanding Scientist, Dronacharya Award, Author of the Year, and Vigyan Ratan Award from different national, as well as international forums. He has organized more than 32 international conferences technically sponsored by IEEE, ACM, and Springer.

**Prof. Mohan L. Kolhe** is with the University of Agder (Norway) as full professor in electrical power engineering with focus in smart grid and renewable energy in the Faculty of Engineering and Science. He has also received the offer of full professorship in smart grid from the Norwegian University of Science and Technology

(NTNU). He has more than twenty-five years' academic experience at international level on electrical and renewable energy systems. He is a leading renewable energy technologist and has previously held academic positions at the world's prestigious universities, e.g., University College London (UK/Australia), University of Dundee (UK); University of Jyvaskyla (Finland); Hydrogen Research Institute, QC (Canada); etc.

**Prof. Brajesh Kumar Singh** is presently working as Professor and Head in Department of Computer Science and Engineering at Raja Balwant Singh Engineering Technical Campus, Agrawith more than 22 years of Teaching experience. Presently, HisArea of research work is Software Engineering, Software Project Management, Data Mining, Soft Computing, Computer Vision, IoT, and Cloud Computing. He has completed his doctorate degree in Computer Science and Engineering from Motilal Nehru National Institute of Technology, Allahabad, Prayagraj (U.P.). He has supervised 01 Ph.D. candidate from AKTU Lucknow and presently supervising 3 Ph.D. candidates. He has more than 80 publications to his credit in national and international journals and proceedings of high repute with large number of citations of his research manuscripts.

# An Automated Early Lung Cancer Detection Using Convolution Neural Network



Santosh Kumar Satapathy, Manan Gandhi, Vanshita Patel, Jui Mehta, Devam Patel, and Rishi Joshi

**Abstract** Lung cancer is one of the leading causes of cancer worldwide. Early diagnosis and treatment are critical to improving patient survival. However, traditional diagnostic methods, such as histopathological evaluation of tissue biopsies, can be misleading and time-consuming. In recent years, convolutional neural networks (CNN) have become an effective method for diagnosing lung cancer. CNNs are deep learning models that learn to recognize and classify patterns in images. In this study, we used transfer learning to develop a pre-trained Google InceptionV3 CNN model to classify lung cancer from CT scan images. We reviewed and tested two different versions of the InceptionV3 model: a base model without weights and a model with some weights. The model with updated weights achieved the best performance with 97.81% training accuracy and 96.00% precision. This demonstrates that adaptive learning can be used to improve the accuracy and reliability of CNN-based lung cancer diagnosis with small datasets. The results of this study show that CNN-based lung cancer diagnosis can improve the early detection and treatment of cancer, thereby improving patients' outcomes. However, more research is needed to confirm these findings on larger data sets and to create better and more comprehensive models.

**Keywords** Medical imaging · Lung cancer screening · Feature extraction · CNN

## 1 Introduction

The unusual growth of cells in the human lung is called lung cancer. Lung cancer is one of the most serious conditions in the present world and has been the leading cause of mortality in the once several decades. Lung cancer is a prominent cancer among both men and women, making up about 25 of all cancer deaths [1]. The primary cause of death from lung cancer about 80 is smoking. Lung cancer in non-smokers can be caused by exposure to radon, separate-hand bank, air pollution, or other factors like

---

S. K. Satapathy (✉) · M. Gandhi · V. Patel · J. Mehta · D. Patel · R. Joshi

Department of Information and Communication Technology, Pandit Deendayal Energy University, Gandhinagar, Gujarat, India

e-mail: [Santosh.Satapathy@sot.pdpu.ac.in](mailto:Santosh.Satapathy@sot.pdpu.ac.in)

workroom exposure to asbestos, diesel exhaust, or certain other chemicals. Non-smokers may get lung cancer as a result of exposure to radon, secondhand smoke, air pollution, or other substances such as diesel exhaust, asbestos at work, or other chemicals lung cancers in some people who do not smoke [2].

Cancer analysis is performed in a pathology laboratory. Microscopic investigation, such as biopsy, and electronic modalities, such as CT, ultrasound, and others, are used to analyze the cancer tissue. A CT scan is the most likely utilized pathological test, and it is very popular for diagnosis. For pathologists and other medical professionals, diagnosing lung cancer and its types is a time-consuming process. There is a huge change in how cancer types are misdiagnosed, which leads to incorrect treatment and may cost patients' lives. So, we will use a convolutional neural network (CNN) like Google's Inception-v3 model, which is employed as a pre-trained deep learning architecture with transfer learning techniques applied to fine-tune the model on a dataset of lung cancer images for accurate classification for early detection of lung nodules in medical imaging, which will contribute to better patient outcomes and overall accuracy for precise diagnosis. Accuracy, precision, recall, F1-score, and the confusion matrix plot were used to assess the performance of the created CNN model.

In Sect. 2, some previous related works are reviewed. The methodology used is described briefly in Sect. 3. Similarly, the research's output is explained and shown with plots and tables in Sect. 4. The conclusion of the paper and future scope are explained in the following sections. Also, cited sources are mentioned in the References section.

## 2 Related Work

The authors of this paper advise a methodology using convolutional neural networks (CNNs) and Google Net to locate lung cancer in CT images. The methodology uses a CNN to extract capabilities from CT snapshots. These features are then fed right into a Google Net classifier to classify the images as both everyday and peculiar. The authors trained their model on a dataset of lung images from both healthy and malignant people. They finished with an accuracy of 98% [3].

The authors have proposed a brand-new approach for lung cancer prognosis from CT scans using a deep learning-assisted guide vector machine (SVM). The approach first extracts a set of capabilities from the CT snapshots using a deep learning network. Those functions are then used to teach an SVM classifier to distinguish between cancerous and non-cancerous nodules. Approaches will be evaluated in the publicly available LIDC/IDRI database. The proposed technique achieved an accuracy of 94% [4].

The authors proposed a lung cancer detection device using a 3-D convolutional neural network (CNN) on CT scan images. The gadget was evaluated on the LUNA-16 dataset. The gadget consisted of two most important elements: pre-processing and classification. Within the pre-processing stage, the CT photograph was resized

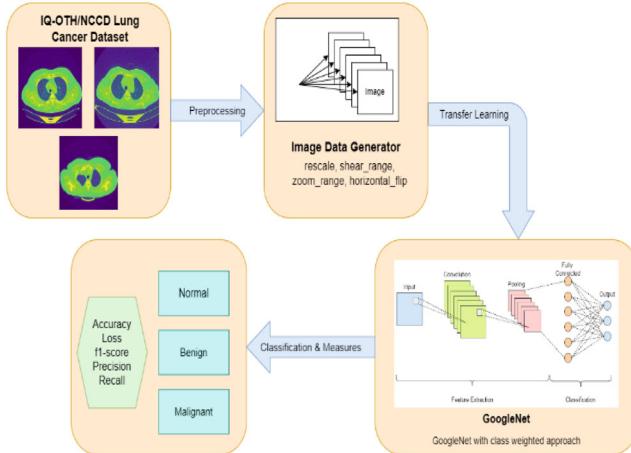
to  $20 \times 50 \times 50$  pixels. In the class stage, a vanilla 3-D CNN is used to classify the photograph as cancerous or non-cancerous. The proposed machine achieved an accuracy of 80% [5].

The author proposed a hybrid algorithm for the automated detection and type of lung cancer in CT scans. The algorithm combines a convolutional neural network (CNN) with the Ebola optimization search set of rules (EOSA). A set of rules first trains a CNN model to extract capabilities from the CT scans. The extracted capabilities are then passed to the EOSA algorithm, which optimizes the weights and biases of the CNN version to attain a viable overall performance. The output of the set of rules is a class label indicating whether the CT scan shows regular, benign, or malignant lung cancer. The set of rules was evaluated on the publicly available Iraq-Oncology Teaching Sanatorium/Countrywide Middle for Cancer Illnesses (IQ-OTH/NCCD) lung cancer dataset. The proposed algorithm achieved an accuracy of 93.21% [6]. The authors propose a deep mastering ensemble second CNN approach for the detection of lung cancer. Their method consists of three one-of-a-kind CNN models, each with a distinctive structure. The outputs of the three fashions are then mixed to produce a final prediction. The authors use the subsequent steps of their method: information pre-processing, model schooling, and model assembling. The authors use the LIDC-IDRI dataset, which is a publicly available dataset of CT test pictures of the lungs. The author has an accuracy rate of 95% [7].

### 3 Methodology

A new practice is displayed here for improved images, here we have used two different types of convolutional neural network architectures: the first is Inception V3 without updating weights in the layers and Inception V3 with updated weights in the layers inside. Massively known for uniformity; Here we have used Inception V3 without updating weights in the first step which we have done using the transfer learning method to extract hierarchical characteristics from the input pictures.

The answer that we got from the transfer learning method is then used as the input for, Inception V3 with updated weights, to further get more precise and give results with more accuracy. The thought of combining two models will give leverage to Inception V3 modules' effective capabilities. We will successively add these two models, it will become easier on our side to compare and visualize the method as its whole, which will improve the system's overall predictability and resilience. For those difficulties that require accurate scenario categorization and sophisticated feature extraction, our two-step technique offers a sustaining practice for improved images (Fig. 1).



**Fig. 1** Complete layout of the proposed Lung cancer detection model

### 3.1 Dataset Generation

The first and most important step is to collect a huge data dataset of CT scan images, which can be easily identified with the help of Kaggle. To give a fair sample of medical disorders, this dataset is filled with diverse images of CT scans showing three categories: benign, malignant, and normal instances [8].

### 3.2 Image Data Generation

After finding the dataset we need to enhance the generalization of Inception V3 models, so here we do pre-process of images using the practice known as the augmentation technique. The process of making image data involves rescaling to a standard size and adding shear and zoom ranges to copy differences in the capture of the images, to add up the diversity in the dataset we will flip the image vertically and horizontally. We will initialize this step, so we can make sure that the model can apply to any circumstances and variances that are seen in medical imaging [9-11].

### 3.3 Model Architecture

The selected deep learning architecture for both the base and second models is Inception V3, a convolutional neural network with a depth of 48 layers. It is possible to load a pre-trained version of this network, which has been trained on over a million images from the ImageNet database. This pre-trained network has the capability to

classify images into 1000 object categories, including items like keyboards, mice, pencils, various animals, and more. The base model is pre-trained on a comprehensive and varied dataset, whereas the second model is initialized, and its weights are updated throughout the training process [12].

### ***3.4 Transfer Learning (Base Model)***

The starting model uses transfer learning, making use of a pre-trained Inception V3 model. Transfer learning in machine learning means using a model trained on one task for a new task. It's like taking what the model already knows and applying it to a different job. By using this pre-trained model, the starting model can learn general features from the original data, making it better at handling your CT scan dataset [13].

### ***3.5 Weight Updating (Second Model)***

The next model starts by setting up a method to update weights. This means the second Inception V3 model begins with specific updated weights and gets refined while being trained on our particular CT scan images dataset. Fine-tuning is about making small tweaks to reach the desired outcome or performance. In deep learning, it means using the weights of a trained neural network to fine-tune another deep learning algorithm for a similar task. This helps the medical images adjust their features to match the characteristics of our dataset [14].

### ***3.6 Training Process***

The training process includes the optimization of models on our dataset. The base model is trained to capture foundational features, and its output serves as input for the second model. The second model then undergoes further training with weight updating, refining its features based on the specific characteristics of the CT scan images [15].

### ***3.7 Classification Task***

Both models are made for a classification job, aiming to sort CT scan images into various categories like Normal, Benign, and Malignant. The ultimate predictions

depend on the features that both the pre-trained base model and the fine-tuned second model have picked up during their learning process [16].

### 3.8 Evaluation

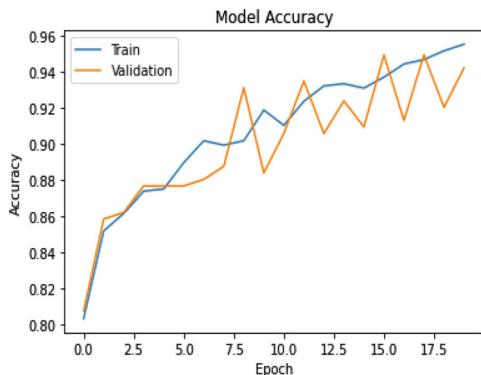
After training, the models are evaluated using a separate test set to assess their performance on unseen data. Metrics such as accuracy, precision, recall, and F1 score can be employed to quantify the models' effectiveness in classifying CT scan images. By combining transfer learning with a pre-trained model and weight updating with a second model, this methodology aims to benefit from both generic feature extraction and fine-tuned adaptation to the specific characteristics of your CT scan dataset [17].

## 4 Result Analysis

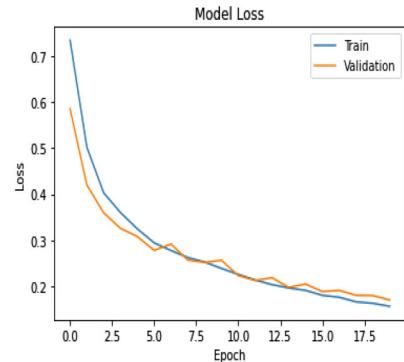
### 4.1 Inception V3 Model

The images underwent training for 20 epochs, utilizing a batch size of 8 with 103 steps in each epoch. In the final epoch, the model attained a training accuracy of 96.72% and a validation accuracy of 95.64%. Figures 2 and 3 depict the model accuracy and loss for both the training and validation data. Figure 4 showcases the confusion matrix for the three-class classification problem using the Inception V3 model.

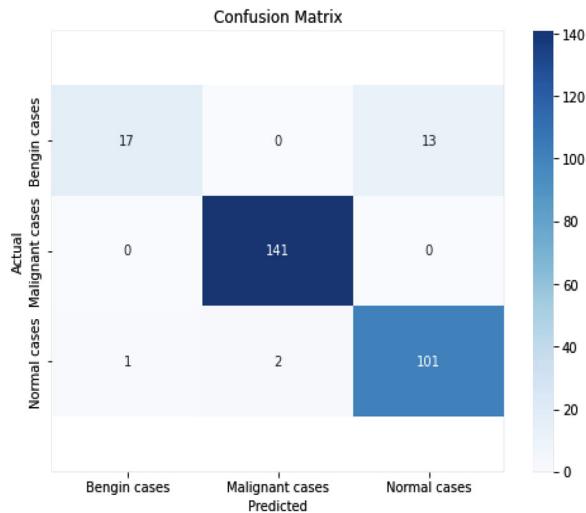
**Fig. 2** Plot of model accuracy versus epoch



**Fig. 3** Plot of model loss versus epoch



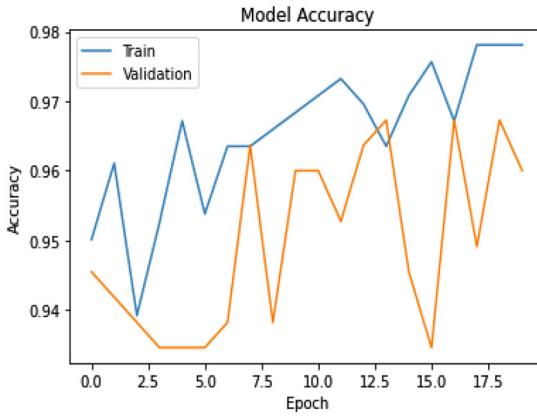
**Fig. 4** Confusion matrix of different image categories for validation images



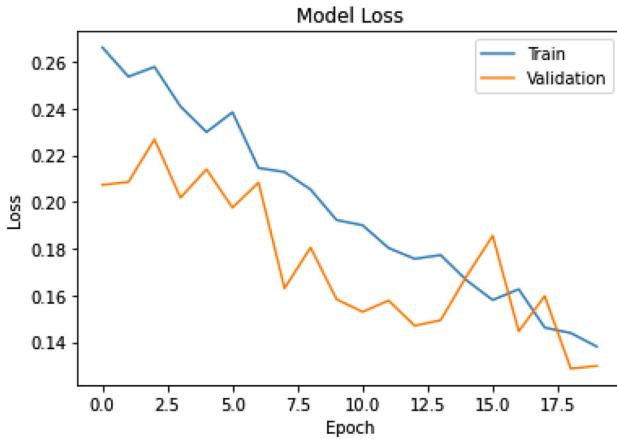
#### 4.2 Inception V3 Model with Class Weighted Approach

The images underwent training for 20 epochs, utilizing a batch size of 8 with 103 steps in each epoch. In the final epoch, the model attained a training accuracy of 98.54% and a validation accuracy of 96.00%.

Figures 5 and 6 showcase the accuracy and loss curves for both training and testing data using the Inception V3 model with a weighted approach. Table 1 provides precision, recall, and F1-score values for various CT scan image categories. Additionally, Fig. 7 displays the confusion matrix, representing the actual label vs. predicted label of images for validation data in specified labeled categories.



**Fig. 5** Plot of model accuracy versus epoch for training and validation images



**Fig. 6** Plot of model loss versus epoch for training and validation images

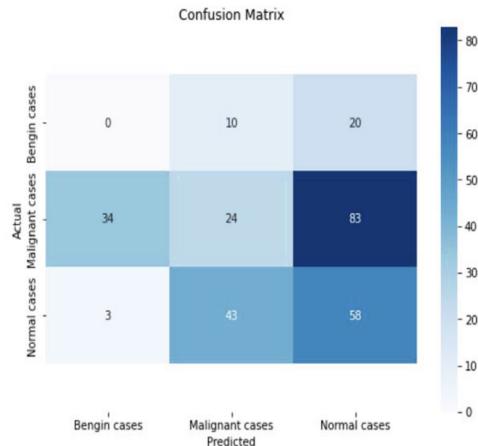
**Table 1** Output score for trained model

Category	Precision	Recall	F1-score	Support
Benign cases	0.94	0.57	0.71	30
Malignant cases	0.99	1.00	0.99	141
Normal cases	0.89	0.97	0.93	104

### 4.3 Comparative Study

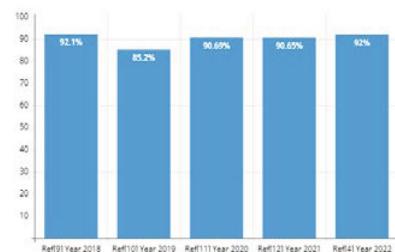
To evaluate the effectiveness of the proposed model for accurate lung cancer detection with the existing state-of-the-art works. Figures 8 and 9 illustrate the comparative

**Fig. 7** Confusion matrix of different image categories for validation images using a weighted approach

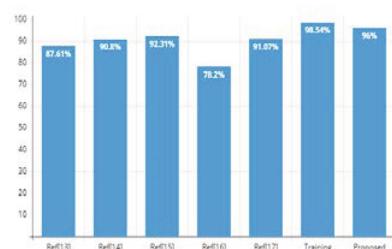


analysis between the proposed model's accuracy with the existing published similar research works.

**Fig. 8** Year-wise comparison



**Fig. 9** Comparative study



## 5 Conclusion

This study focuses on detecting lung cancer using CT scan images, employing a Convolutional Neural Network (CNN) with a transfer learning approach to classify images into benign, malignant, and normal categories. The pre-trained model achieved 95.50% and 94.18% training and validation accuracy, respectively, while the model with updated weights achieved 97.81% and 96.00%. To assess model performance, precision, f1-score, and recall were calculated, and a confusion matrix plot was created. However, it's essential to evaluate CNN models on larger and more diverse datasets. The dataset used in this study was relatively small, and assessing CNN models on more extensive and varied datasets is crucial for ensuring their applicability in real clinical settings. Future developments could include proof-of-concept studies that explore methods enhancing medical images using CNN models, highlighting strengths and weaknesses. It emphasizes the dependence on the area and type of data. Additionally, experimental CNN models could optimize within their own range. Beyond CT scan images, integrating other data modalities like clinical and genetic data could enhance predictive AI model performance. In terms of design, there's a need for the development of explainable AI models. While CNNs achieve high accuracy, their decision-making processes are often opaque. Interpretable AI models could benefit doctors by providing a better understanding of AI-based diagnoses, potentially fostering greater trust. Moreover, addressing biases inherent in CNNs, such as those related to patients' race and class, is crucial to ensure fair and unbiased healthcare outcomes.

## References

1. American Cancer Society, Lung Cancer Statistics (2020). <https://www.cancer.org/cancer/lung-cancer/about/key-statistics.html>
2. American Cancer Society, Lung Cancer Causes (2019). <https://www.cancer.org/cancer/lung-cancer/causes-risks-prevention/what-causes.html>
3. Pandian R, Vedanarayanan V, Ravi Kumar DNSR, Rajakumar R (2022) Detection and classification of lung cancer using CNN and Google net. Meas Sens 24(2):100588
4. Shafi I, Din S, Khan A, De La Torre Díez I, del Palí Casanova, RJ, Tutusaus Pifarre K, Ashraf I (2023) An effective method for lung cancer diagnosis from CT scan using deep learning-based support vector network. Cancers 14(21):5457
5. Ahmed T, Parvin MS, Haque MR, Uddin MS (2020) Lung cancer detection using CT image based on 3D convolutional neural network. J Comput Commun 8(04):90
6. Mohamed TIA, Oyelade ON, Ezugwu AE (2023) Automatic detection and classification of lung cancer CT scans based on deep learning and Ebola optimization search algorithm. PLoS One 18(11):e0285796
7. Shah AA, Malik HAM, Muhammad AH, Alourani A, Butt ZA (2023) Deep learning ensemble 2D CNN approach towards the detection of lung cancer. Nat Sci Rep 13(1):19437
8. Alyasriy, hamdalla; AL-Huseiny M (2021) The IQ-OTHNCCD lung cancer dataset. Mendeley Data, V2, <https://doi.org/10.17632/bhmdr45bh2.2>
9. Liu X, Hou F, Qin H, Hao A (2018) Multi-view multi-scale CNNs for lung nodule type classification from CT images. Pattern Recognit 77:262–275

10. Li Y, Zhang L, Chen H, Yang N (2019) Lung nodule detection with deep learning in 3D thoracic MR images. *IEEE Access* 7:37822–37832
11. Ali I, Muzammil M, Haq IU, Khaliq AA, Abdullah S (2020) Efficient lung nodule classification using transferable texture convolutional neural network. *IEEE Access* 8:175859–175870
12. Tong C, Liang B, Su Q, Yu M, Hu J, Bashir AK, Zheng Z (2020) Pulmonary nodule classification based on heterogeneous features learning. *IEEE J Sel Areas Commun* 39:574–581
13. Li W, Cao P, Zhao D, Wang J (2016) Pulmonary nodule classification with deep convolutional neural networks on computed tomography images. *Comput Math Methods Med* 2016:1–7. <https://doi.org/10.1155/2016/6215085>
14. Cheng JZ, Ni D, Chou YH, Qin J, Tiu CM, Chang YC et al (2016) Computer-aided diagnosis with deep learning architecture: applications to breast lesions in US images and pulmonary nodules in CT scans. *Sci Rep* 6:24454. <https://doi.org/10.1038/srep24454>
15. Hussein S, Gillies R, Cao K, Song Q, Bagci U (2017) TumorNet: lung nodule characterization using multi-view convolutional neural network with Gaussian process. [arXiv:1703.00645v1](https://arxiv.org/abs/1703.00645v1)
16. Gruetzmacher R, Gupta A (2016) Using deep learning for pulmonary nodule detection& diagnosis. In: Twenty-second Americas conference on information systems, San Diego, 2016
17. Nibali A, He Z, Wollersheim D (2017) Pulmonary nodule classification with deep residual networks. *Int J Comput Assist Radiol Surg* 12:1799–1808. <https://doi.org/10.1007/s11548-017-1605-6>

# Empowering Robust Speech Emotion Recognition Using Deep Neural Network



Muthukuru Jayanth, Saravanan Palani, and M. Marimuthu

**Abstract** Speech Emotion Recognition, also known as SER, refers to the act of recognizing emotion from a speech input of a human. This is based on the general fact that the emotion of a person is reflected in the tone and pitch of his/her voice. In traditional recognition techniques, handcrafted features may not capture all relevant information present in the speech signal, leading to suboptimal performance. The paper, proposes a Deep Neural Network based Speech Recognizing technique called DeNSER. The work is aimed at offering advantages like end-to-end learning, automatic feature learning, capturing complex relationships, scalability, and adaptability. The proposed system has used the RAVDESS dataset for training and testing speech and song samples. The implemented DNN-based SER, DeNSER has performed well in recognizing emotions with 89.6% accuracy.

**Keywords** Speech emotion recognition · Deep neural networks · Feature extraction · Signal processing · MelSpectrogram · Chroma\_stft

## 1 Introduction

Speech Emotion Recognition (SER) is a technological domain centered on detecting and categorizing human emotions from speech cues. This intricate process entails examining diverse acoustic features, including pitch, intensity, and spectral properties, to discern the speaker's emotional state, such as happiness, sadness, anger, fear, and disgust. The importance of SER has grown significantly due to its diverse applications across sectors like call centers, healthcare, and criminal investigations.

In call centers, SER can enhance customer service by identifying satisfaction levels and potential issues, thereby fostering customer loyalty. In the medical domain, it aids in diagnosing and treating patients with emotional disorders by monitoring changes in their emotional states. Meanwhile, law enforcement agencies utilize

---

M. Jayanth · S. Palani (✉) · M. Marimuthu

School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, India 600127

e-mail: [saravanan.p@vit.ac.in](mailto:saravanan.p@vit.ac.in)

SER in criminal investigations to analyze speech patterns and emotional cues from recorded conversations, which could serve as evidence.

However, SER faces challenges, especially with noise and distortion in audio data, which stem from background noise, microphone interference, or speech pattern variations. To combat this, data augmentation techniques are employed, generating additional training data through modifications like adding white noise or changing the speech signal's pitch. This approach improves the robustness of SER models against noise and distortions and enhances their accuracy in real-life problems.

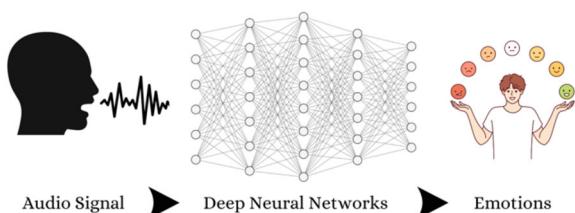
Speech emotion recognition has a numerous application in the everyday life. It is used in criminal investigation, call centers, hospitals, intelligent assistance, surveillance, and detection of potentially hazardous events. So, in order to meet the above requirements or try contributing something in the above-mentioned fields, speech emotion recognition is really necessary and can be useful in many fields.

Deep Neural Networks (DNNs) have become a popular tool in SER due to their ability to learn complex input data representations. DNNs can be used for many purposes in speech recognition. They consist of multiple interconnected layers that extract and classify relevant speech signal features. Besides DNNs, other machine learning algorithms like support vector machines and decision trees are also utilized in SER. The various advantages of the SER with DNN models are as follows:

1. DNNs directly map the input speech signals into transcriptions without intermediate processing.
2. DNNs autonomously learn hierarchical representations of speech features. Hence, manual feature engineering is eliminated.
3. Using several layers and non-linear functions, DNNs can manage complex interactions between input features. This can improve the accuracy of emotion recognition.
4. DNNs can be effectively scaled for large datasets and complex tasks.
5. DNNs are most versatile in nature due to its adaptability to diverse speakers, languages, and accents by generalizing the contexts.

The objective is to create an SER system as shown in Fig. 1 that identifies a wide array of emotions with high accuracy and performance. This involves selecting a suitable DNN architecture that can process acoustic features effectively and exploring data augmentation techniques to enhance model reliability across different scenarios. By meeting these goals, the project aspires to advance SER technology, making it more applicable in various sectors where emotion recognition is essential.

**Fig. 1** Speech emotion recognition



The scope of this project covers the development and evaluation of an SER system using DNNs, from data collection and augmentation to feature extraction and DNN model training. The project seeks to validate the SER system's performance in realistic conditions and explore its applications across multiple domains. Successfully achieving the project's objectives could significantly improve SER systems, making them more versatile and effective for recognizing human emotions from speech.

## 2 Literature Survey

Bhangale et al. [1] present an improved speech emotion recognition method that combines diverse acoustic features and a one-dimensional deep convolutional neural network, achieving high accuracy on standard datasets and promising enhanced human–computer interaction. Alluhaidan et al. [2] enhance speech emotion recognition by integrating MFCCs with time-domain features and employing CNNs, achieving up to 97% accuracy across various datasets, showcasing the method's broad applicability and improved performance. Subramanian et al. [3] discuss a deep learning model for emotion recognition in speech, achieving up to 83% accuracy with specific audio features. This approach promises improvements in speech recognition and audio analysis systems.

Haddad et al. [4] propose a multimodal automatic emotion recognition method combining audio, visible, and infrared images, achieving 86.36% accuracy. This innovative approach utilizes CNN and ANN for feature extraction and emotion prediction, showcasing its potential in improving emotion recognition systems. Suryakanth et al. [5] explore speech emotion recognition using Deep Learning, a shift from traditional algorithms to more effective, data-driven approaches. This study underscores the potential of deep learning in improving computer–human interaction by accurately interpreting emotional speech.

Ishak et al. [6] highlight advancements in NLP and speech recognition, emphasizing a novel neural network-based method that enhances phoneme identification and reading accuracy. This approach, leveraging deep speech technologies and frameworks like TensorFlow, showcases superior performance in speech understanding and literacy support, marking significant progress in computational linguistics. Makarand et al. [7] into music emotion recognition using deep learning, aiming to automate emotional tagging of songs. Their research compares two deep learning models, highlighting the potential of these technologies to improve music streaming services by accurately categorizing songs into emotional genres, thereby enhancing user experience.

Ashraf et al. [8] present a multimodal emotion recognition system combining audio-visual inputs with CNNs and extreme learning machines, achieving high accuracy across different datasets. This approach underscores the efficacy of integrating speech and video data in enhancing emotion recognition capabilities, offering significant improvements in human–computer interaction. Han et al. [9] enhance music emotion recognition using an innovative neural network structure

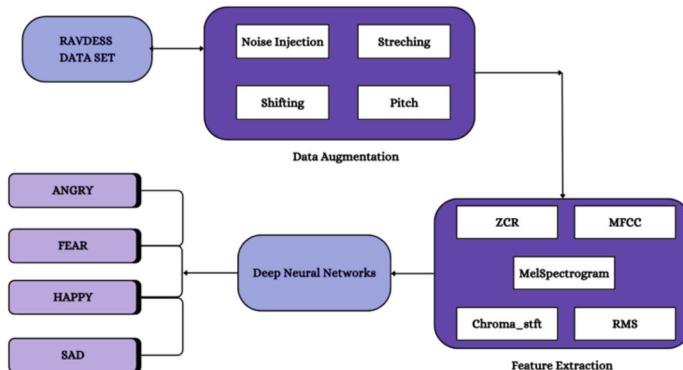
that combines an improved Inception module with a GRU for feature extraction, achieving 84% accuracy on the Soundtrack dataset. This method outperforms traditional models, demonstrating its potential for more precise music emotion analysis. Reggiswarashari et al. [10] into speech emotion recognition using a 2D-convolutional neural network, addressing the challenges of feature selection and model choice in affective computing. Their work emphasizes the significance of accurately identifying emotions in speech, with potential applications in enhancing human–computer interaction across various fields.

### 3 Methodology

The architecture of the proposed approach for speech emotion recognition involves several steps, including dataset understanding, data augmentation, feature extraction, and data modeling as shown in Fig. 2. These steps are critical for developing an accurate and robust speech emotion recognition system.

Compared to the base paper [2] which stated that to solve this issue, in their model of research, they have combined MFCCs and time-domain features (MFCCCT) to enhance the performance of SER systems. The proposed hybrid features were given to a convolutional neural network (CNN) to build the SER model. The hybrid MFCCCT features together with CNN outperformed both MFCCs and time-domain (t-domain) features on the datasets by achieving a better accuracy. The proposed features have the potential to be widely utilized to several types of SER datasets for identifying emotions. Additionally, CNN achieved better performance compared to the machine learning (ML) classifiers that were recently used in SER.

The proposed model is trained with the combination of MFCC and intruding feature extraction techniques such as Zero Crossing Rate (ZCR), Mel Frequency Cepstral Coefficients (MFCC), MelSpectrogram, Chroma\_stft and Root Mean



**Fig. 2** Architecture of DNN-Based SER (DeNSER)

Square (RMS). And we have achieved better accuracy than the base paper that I have mentioned in the above.

### **3.1 Dataset**

The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) by Ryerson University's Centre for Digital Humanities offers over 24,000 recordings with 24 actors portraying eight emotions and neutral expressions. Meticulously labeled by professionals, it ensures accurate emotional annotations, vital for SER model development.

### **3.2 Architecture of Proposed DNN-Based SER (DeNSER)**

Data augmentation techniques are applied to enhance model robustness, introducing disturbances like noise, pitch variation, and time stretching to the audio files. This process enriches the dataset, preparing the model for real-world scenarios with distorted or noisy input audio. Feature extraction is then conducted to derive significant features from the audio data, such as Mel Frequency Cepstral Coefficients (MFCCs), pitch, and energy. These features serve as inputs for the model during both training and testing phases.

The dataset is partitioned into training and testing sets, and a Deep Neural Network (DNN) model is employed. The DNN architecture consists of convolutional and pooling layers followed by flatten and dense layers, enabling the model to learn complex patterns in the data. The model undergoes training on the training set and validation on the testing set across multiple epochs to achieve high accuracy and generalize well to unseen data.

### **3.3 Data Augmentation**

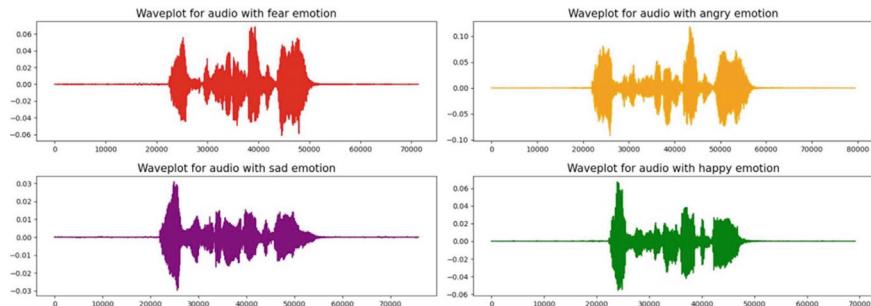
The data augmentation techniques we used in speech emotion recognition model Fig. 5.

- A. **Noise Injection:** Adding random noise to the audio signal can simulate the presence of background noise in the real-world environment. This can help the model to learn to distinguish between the target emotion and the noise.
- B. **Stretching:** Stretching the audio signal in time can simulate variations in the speaker's speed of speech. This can help the model to learn to recognize the target emotion across different speech rates.

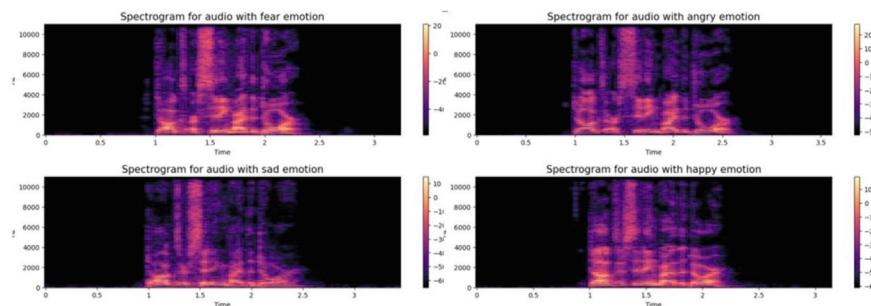
- C. **Shifting:** The idea behind shifting is very simple. It is used in shifting the audio to either left or right with a random second. The following picture represents the wave plot for the resulting audio when shifting is performed on the simple audio input mentioned earlier.
- D. **Pitching:** Changing the pitch of the audio signal can simulate variations in the speaker's pitch or intonation. This can help the model to learn to recognize the target emotion across different speaking styles.

### 3.4 Data Visualization

Figure 3 depicts the wave with different emotion for the selected audio. Figure 4 depicts the spectrogram of audio samples encompassing all emotions present in the dataset.



**Fig. 3** Wave plots for audio with different emotions (Fear, Angry, Sad and Happy)



**Fig. 4** Spectrogram for audio with different emotions (Fear, Angry, Sad and Happy)

### 3.5 Feature Extraction

#### Noise Removal

Due to the even symmetry of the autocorrelation function, the sinusoidal waveform components of the Fourier series will all be 0. Consequently, Eq. (1) can be simplified to include only valid cosine factors.

$$S(f) = \frac{1}{t} \int_0^T r_{auto}(t) \cos\left(\frac{2\pi m t}{F_0}\right) dt \quad (1)$$

where  $r_{auto}(t)$ —auto-correlation function.

$S(f)$ —modified log-spectrum.

$F_0$ —Pitch frequency.

$m = 0, 1, 2 \dots N$ .

Every audio file is made up of two important things: sample data and sample rate. Now based on these two things, i.e., sample data and sample rate, the following features can be extracted by performing several transformations on them:

- Zero Crossing Rate (ZCR)
- Mel Frequency Cepstral Coefficients (MFCC)
- MelSpectrogram
- Chroma\_stft
- Root Mean Square (RMS).

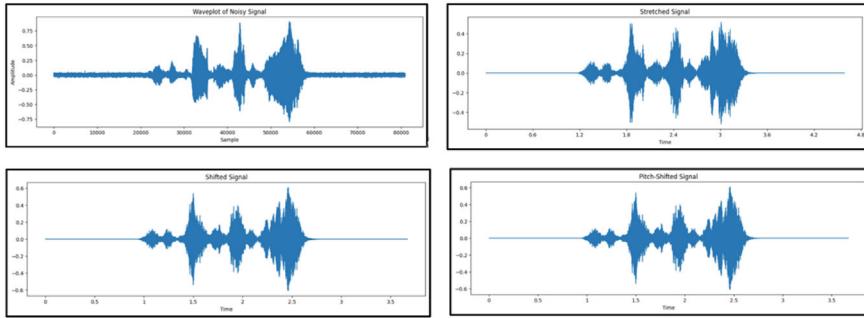
### 3.6 Data Preparation

This involves dividing the entire dataset into two independent subsets as training and testing, and ensures reproducibility by setting a seed for random splitting importing the train and test datasets in the ratio of 70:30 ration of training and testing.

## 4 Outputs

### 4.1 Wave Plots of Data Augmentation

To improve the robustness of my Speech Emotion Recognition (SER) model, I employed various data augmentation techniques. These techniques involve manipulating audio data, such as adding background noise to simulate real-world scenarios,



**Fig. 5** Wave plots of data augmentation by noise injection, stretching, shifting and pitch-shifting

stretching or shifting the audio to account for speaking speed variations and modifying the pitch to capture different voice characteristics. The effectiveness of these techniques is visualized in Fig. 5, and the methodology section provides a deeper explanation of each.

## 5 Results

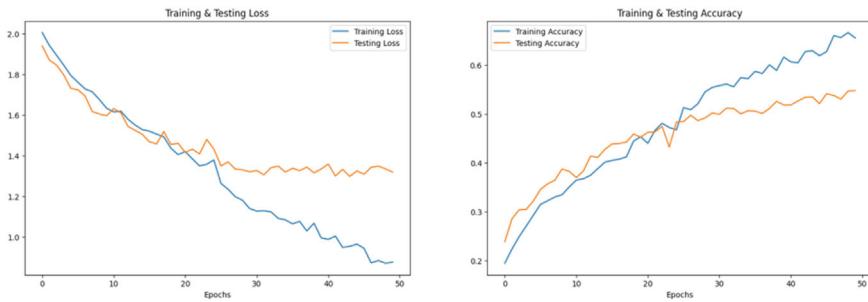
In this research of speech emotion recognition (SER), the framework looks for potential signs of SER by employing DNN approaches, with an emphasis on. The data augmentation of the dataset Ravdess is approached by using different techniques of feature extraction to obtain better accuracy. The concepts underlying these models are explained in the paper, along with specifics about the training procedure, such as hyperparameter selection and assessment measures. A number of performance indicators are displayed in the results section, such as accuracy, precision, recall, F1-score, and confusion matrix. By comparison, current techniques have demonstrated accuracy of up to 89.6%

Overall, Fig. 6 serves as a crucial component in the evaluation and optimization of machine learning models, offering a clear and concise representation of their performance dynamics throughout the training process.

Table 1 shows the predictions of the model on the test data. As seen in the figure, the model has correctly predicted the labels of most of the instances, resulting in an overall accuracy of 89.6%.

The following the details are mentioned in Table 2.

**Precision:** This column represents how many of the predictions for each class were actually correct. For example, for the “angry” class, the model had a precision of 0.92, which means 92% of the time the model predicted “angry” it was correct.



**Fig. 6** Model loss and accuracy of the proposed system

**Table 1** Predicted and actual labels

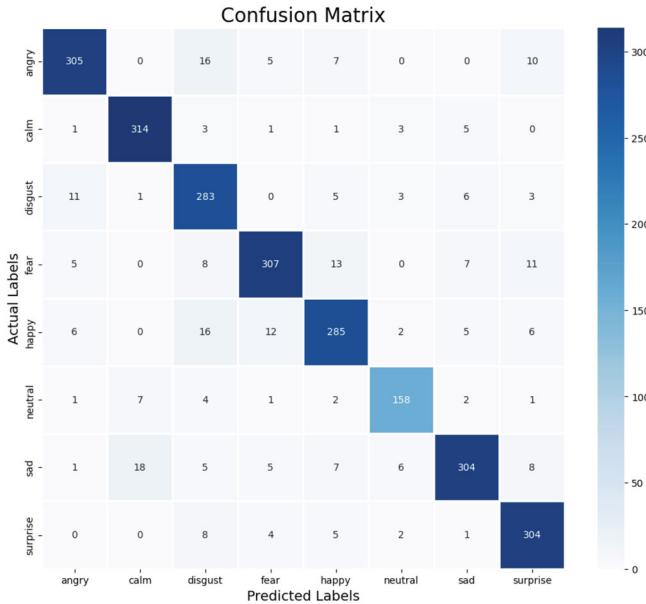
	Predicted labels	Actual labels
0	Fear	Happy
1	Sad	Sad
2	Happy	Happy
3	Disgust	Disgust
4	Neutral	Neutral

**Table 2** Classification report

	Precision	Recall	F1-score	Support
Angry	0.92	0.89	0.91	343
Calm	0.92	0.96	0.94	328
Disgust	0.83	0.91	0.86	312
Fear	0.92	0.87	0.90	351
Happy	0.88	0.86	0.87	332
Neutral	0.91	0.90	0.90	176
Sad	0.92	0.86	0.89	354
Surprise	0.89	0.94	0.91	324
<i>Accuracy</i>			0.90	2520
<i>Macro avg</i>	0.90	0.90	0.90	2520
<i>Weighted avg</i>	0.90	0.90	0.90	2520

**Recall:** This column represents how well the model identified all the actual instances of each class. For instance, for the “angry” class, the model had a recall of 0.89, which means it correctly identified 89% of the actual “angry” cases.

**F1-Score:** This column is a harmonic mean of precision and recall, combining their performance into a single metric. A score of 1 indicates perfect balance between



**Fig. 7** Confusion matrix

precision and recall. In this case, all F1-scores are around 0.9, suggesting the model performs well in all classes.

**Accuracy:** This is the overall proportion of correctly classified data points. The model achieved an accuracy of 90%, indicating it classified 90% of the data points correctly.

Confusion Matrix Fig. 7: This table provides a visual representation of a classification model's performance. The projected classes are represented by columns, and the actual classes are represented by rows. The amount of data points that belong to one class but were initially projected to belong to a different class is indicated by each cell.

Overall, the results indicate that the model performs well in classifying sentiment across different categories. It achieves high precision, recall, and F1-score for all classes, demonstrating its effectiveness in sentiment analysis. The high accuracy further confirms the model's overall ability to correctly identify sentiment in the data.

## References

1. Bhangale K, Kothandaraman M (2023) Speech emotion recognition based on multiple acoustic features and deep convolutional neural network. *Electronics* 12(4):839
2. Alluhaidan AS, Saidani O, Jahangir R, Asif Nauman M, Saidani Neffati O (2023) speech emotion recognition through hybrid features and convolutional neural network. *Appl Sci* 13(8):4750
3. Subramanian, Raja R, Aditya Ram K, Lokesh Sai D, Venkatesh Reddy K, Akarsh Chowdary K, Datta Reddy KD (2022) Deep learning aided emotion recognition from music. In: 2022 international conference on automation, computing and renewable systems (ICACRS). IEEE, pp 712–716
4. Haddad S, Daassi O, Belghith S (2023) Emotion recognition from audio-visual information based on convolutional neural network. In: 2023 international conference on control, automation and diagnosis (ICCAD). IEEE, pp 1–5
5. Suryakanth VG, Dubey AK (2023). Speech emotion recognition using deep learning. *Int J Sci Technol Eng.* <https://doi.org/10.22214/ijraset.2023.49703>
6. Ishak MK, Øivind Madsen D, Al-Zahrani FA (2023) An optimal method for speech recognition based on neural network. *Intell Autom Soft Comput* 36(2):1951–1961. <https://doi.org/10.32604/iasc.2023.033971>
7. Makarand V, Sneha T, Wadkar HS (2022) Evaluating deep learning models for music emotion recognition. *Int J Eng Appl Sci Technol.* <https://doi.org/10.33564/ijeast.2022.v07i06.026>
8. Ashraf A, Gunawan TS, Arifin F, Kartiwi M, Sophian A, Habaebi MH (2022) On the audio-visual emotion recognition using convolutional neural networks and extreme learning machine. *Indones J Electr Eng Inform (JEEI)* 10(3):684–697
9. Han X, Chen F, Ban J (2023) Music emotion recognition based on a neural network with an inception-GRU residual structure. *Electronics* 12(4):978
10. Reggiswarashari F, Sihwi SW (2022) Speech emotion recognition using 2D-convolutional neural network. *Int J Electr Comput Eng* 12(6):6594

# DeepLeaf: A Custom CNN Approach for Mulberry Leaf Classification



Tripti Mishra, Vanshaj Singhal, Yashaswat Verma, Monika, and Manish Raj

**Abstract** Mulberry leaves provide a lot of medicinal and industrial uses along with being an excellent food source for silkworms. But one has to know the correct type of mulberry leaf to make use of it correctly. This research paper aims to achieve a high percentage of accuracy in mulberry leaf image classification using deep learning models. The dataset chosen is a mulberry leaf dataset consisting of 10 different classes. The reason behind using deep learning is the efficiency of such neural network models which can easily simulate the decision-making power of a human brain. A custom model has been developed for the research which uses convolution neural networks to achieve an accuracy of over 92.25%. This study also compares the performance of this custom model with other pre-trained models which are deemed to be highly accurate for the work of image classification and are frequently used in developing high-end models for a bigger use.

**Keywords** Custom CNN · Deep learning · Mulberry leaf · Augmentation

## 1 Introduction

Plants play a vital role in maintaining the balance of nature and are crucial for our survival. They have hundreds of uses in cooking, medicine, and therapy. There are thousands of types of plants on Earth ranging from small herbs to towering trees. Plant forests cover about 31% of the earth's surface, which is about 4.06 billion hectares, and absorb roughly 15.6 billion tons of carbon dioxide ( $\text{CO}_2$ ) from the atmosphere every year. But, mass deforestation has put these benefits at risk as we are losing about 10 million hectares of plant cover globally. Therefore, it is important to monitor plant species and their populations.

---

T. Mishra · V. Singhal · Y. Verma · Monika · M. Raj (✉)  
Bennett University, Greater Noida, India  
e-mail: [manish.raj@bennett.edu.in](mailto:manish.raj@bennett.edu.in)

Monika  
e-mail: [monika@bennett.edu.in](mailto:monika@bennett.edu.in)

Earth houses about 374,000 different known species of plants that are classified into herbs, shrubs, trees, creepers, and climbers. Leaves are an important factor in determining these plant types. The broad qualities like the size, color, and shape provide the first indication of the plant like the coniferous trees are adapted to cooler climes and have needle-like leaves, contradictory strongly with the huge, broad leaves of tropical rainforest trees. On the other hand, the minute qualities of leaves such as veins, chlorophyll content, arrangement, and edges are all examined in detail to differentiate them from each other. For instance, vein patterns can differ between dicots and monocots, from net-like to parallel [1]. The Mulberry is a genus of flowering plants called *Morus* which belong to the Moraceae family of deciduous trees. The 10 distinct species of this plant are all native to the temperate climates of Asia and North America. It is a greatly valuable resource due to its uses in medicine and industry. The leaves of specific mulberries are used as animal feed as well as the main nourishment source for silkworm larvae which utilize proteins (namely, fibroin and sericin) to produce silk fibers for their cocoons [2]. These cocoons are then harvested and spun into high-quality silk fibers which are sold all around the world. Traditional Chinese medicine has utilized this plant for centuries for its ability to lower serum glucose and blood pressure. This is due to the abundance of bioactive compounds like flavonoids, polyphenols, alkaloids, steroids, etc., in the leaves and bark of the tree [3].

It is essential to identify the type of mulberry before it is used for commercial purposes as the different species have varying compositions of medicinal chemicals and usage properties. Currently, botanists use their expert knowledge to identify and classify different plant species largely through naked-eye observations which is a time-consuming and ineffective method [2]. There is always a margin of error when making mere observation-based classifications in this manner. A survey conducted in paper [4], shows that participants accurately identified 79% of plant characters with experienced botanists performing only slightly better than novices or beginners. This highlights the need for a system that can classify plants faster and more accurately, taking away the laborious task from humans and automating it.

Correct identification and classification of plant species especially Mulberry leaves remains an exhausting endeavor as the method highly depends on human skills and knowledge which are prone to errors, time-consuming, and lead to inaccuracies. An automated and accurate technique for classification is required as the demand for mulberry leaves is increasing in several domains. Hence, this paper discusses a deep learning approach for using images of mulberry leaves to identify and differentiate between its 10 different species. Several techniques have been previously used in the classification of leaves from beans [5], olive [6] grape vines, tomato plants [7], etc., and we will use our learning from them to create a highly accurate CNN architecture.

The main contributions of the paper are as follows:

1. A novel, very effective Custom CNN model has been developed containing convolutional layers, max-pooling layers, flattened layers, and dense layers to classify 10 classes of Mulberry Leaves.

2. A detailed comparison has been made among the Sequential Model, InceptionV3, VGG16, and the custom model.
3. Validation accuracy and training time are utilized as evaluation metrics for the comparison. The suggested CNN model surpasses the rest of the pre-trained models that have been taken into consideration.
4. The methodology applied in this paper provides a fresh perspective by using the most recent developments in deep learning. Notably, to the best of our knowledge, the model attains an unmatched level of accuracy of 92.25% compared to other previous studies.

Section 2, the first section of the paper, delves into earlier research that is closely related to our investigation. After that, Sect. 3 delves further into the development of our original concept. Section 4 then goes over the outcomes of the model that we were able to collect. These findings are examined and contrasted with those of other models in Sect. 5. Section 6 provides a summary of the main results, conclusions from the experiments, and possible directions for further research.

## 2 Related Work

Many approaches have been researched for many different types of leaves and their diseases, as each plant can have numerous diseases, along with different conditions that can take part in their structure and patterns as well as temperature, climate, weather, elevation, etc. So, there can be approaches based on regression, transfer learning, neural networks, ensemble learning, SVMs, etc., like a deep hybrid model proposed in [6] for image-based olive leaf diseases classification where EfficientNetB0 has been combined with a logistic regression classifier which gave an accuracy of 96.14%. Similarly, a CNN model utilizing transfer learning model VGG-16 is used for tomato leaf classification in [8] which gives a 95% accuracy.

The size of the dataset can also produce an impact on the accuracy of the model a custom CNN model in [9] gives 99.81% accuracy because it had enough data to train on with a total of 54,000 images in the dataset. It can be similarly seen in [5] how a small dataset with around 1300 images produces an accuracy of 91.74% even after applying models like MobileNetV2, EfficientNetB6, and NasNet. The research in [10] produced a brilliant result of 99.98% accuracy with ant colony optimization with CNN for leaf disease identification and it was done through optimized deep learning.

Paper [11] provides images that are augmented to produce an even bigger dataset with 87,000 images. Each of the approaches is unique with quite complex models used for prediction like fuzzy c-means clustering algorithm along with LSTM, squeeze, and excitation CNN, a custom CenterNet model, etc., but the common characteristic is high accuracy with most of them having an accuracy of more than 99.5% while there is only one which has 96% accuracy. The [12] study's squeeze-and-excitation CNN model gets an impressive 99.28% accuracy, proving its exceptional

ability to detect and classify 38 classes consisting of 54,303 images of chilli plant diseases.

This achievement gives way to a promising future in automated diagnosis and disease management for chilli crops. The research [7] unveils a new method (DLMC-Net) for identifying diseases in plant leaves. The exciting results show that DLMC-Net achieves impressive classification accuracies of 93.56%, 92.34%, 99.50%, and 96.56% for classifying different plant diseases. This innovative model is both powerful and efficient, making it ideal for precisely categorizing various plant diseases. Research [13] consists of two classes tomatoes and grapes having 23,558 images. Various augmentation techniques have been implemented to increase the performance. VGG Model has been employed achieving an accuracy of 95.71% and 98.40%.

The research in [14] aims to create a dataset of the growth cycle and stages of a pomegranate which has 5 categories bud, flower, early fruit, mid-growth, and ripe. In a similar fashion, a wider view of research [15] gives a dataset called PSFD-Musa about the banana plant, stem, fruit, leaf, and disease. A complex approach can be seen in [16] according to which the existing methods still lack the ability for detailed feature identification, thus, it uses a combination of multiple models where ResNet50 is used as a backbone for solving gradient disappearance and explosion, ghost module to enhance the features, ResNeSt Module to enhance the extraction effect and finally the hybrid activation function of RRelu and Swish to speed up the model training, which can classify the three major types of diseases found in banana leaves that are Sigatoka, Cordana, and Pestalotiopsis with an accuracy of 96.425, 97.54% and 95.58% respectively.

There are more studies on the same topic and disease categories with easier approaches like in [17], the Banana SqueezeNet model is proposed as the model that gives 96.25% accuracy. This particular is very fast and lightweight so that it can run on a mobile phone as well and it is done by optimizing the model using Bayesian optimization and fire modules. A self-made dataset is also available that has 937 images for the same categories of banana leaf diseases where each image is augmented with techniques like shear, gaussian blurring, linear contrast adjustment, etc., to increase the size of the dataset to 400 images per category termed as BananaLSD in [18]. This kind of dataset can be used to test the model and fine-tuning approach to be used. A deeper dataset is provided for research in [19] with 16,092 images which contain the images of the banana leaves in Tanzania, which can help to gather more information based on different types of conditions. Table 1 shows the literature review of 15 different papers that are related to mulberry leaves.

### 3 Proposed Methodology

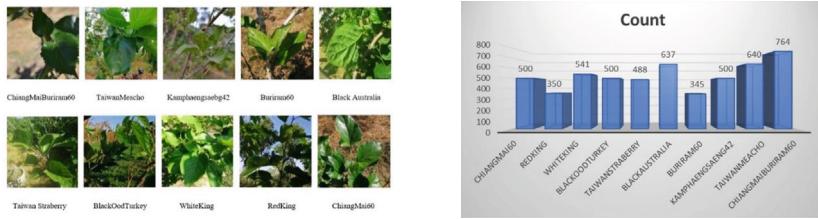
The methodology of the research paper consists of three sections dataset section collection, data pre-processing, and model architecture of the custom CNN model giving a detailed explanation of the framework.

**Table 1** Literature review

Authors	Dataset (classes)	Methods	Accuracy (%)
Hicham and El Akhal [6]	4138(4)	EfficientNetB0 combined with a logistic regression classifier	96.14
Paul et al. [8]	25,851 (11)	Custom convolutional neural network (CNN) model utilizing TL-based models VGG-16 and VGG-19	95
Shewale et al. [9]	54,303 (38)	Custom CNN	99.81
Singh et al. [5]	1295 (3)	MobileNetV2, EfficientNetB6, and NasNet	91.74
Algani et al. [10]	–	Ant Colony Optimization with Convolution Neural Network	99.98
Umamageswari et al. [11]	87,000 (38)	Fuzzy C-means clustering algo, LSTM	96
Naik et al. [12]	54,303 (38)	Squeeze and Excitation CNN	99.28
Sharma et al. [7]	25,597 (4)	DLMC-Net	96.56
Paymode et al. [13]	23,558 (2)	VGG	98.4
Jifei and Zhao [14]	5857 (5)	–	–
Deng et al. [15]	13,021 (4)	Ghost ResNeSt-Attention RReLU-Swish Net, K-scale VisuShrink algorithm	96
Medhi and Deb [16]	8000 (2)	–	–
Bhuiyan et al. [17]	937 (4)	BananaSqueezeNet	96.25
Arman et al. [18]	1600 (2)	–	–
Mduma and Leo [19]	16,092 (3)	–	–

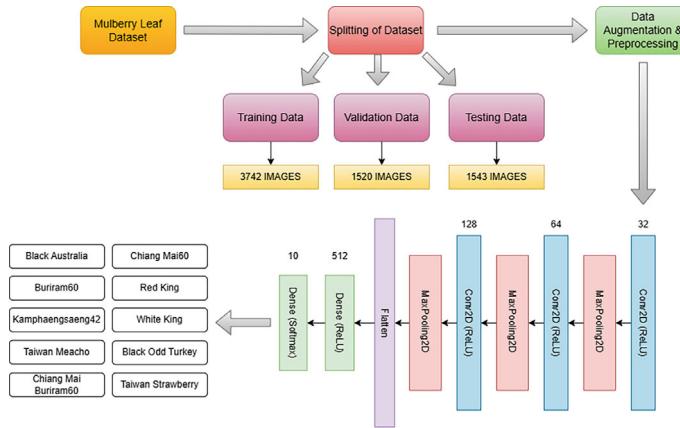
### 3.1 Dataset Collection

The Mulberry Leaf dataset contains a total number of 6808 which are classified into ten different types of mulberry leaves. All the images are  $224 \times 224$  pixels per electron microscope and are in RGB channel. The dataset has been divided into an 80–20 split to form training data (5265 images) and testing data (1543 images). The training data is further divided into an 80–20 split to create training data and validation data. The 10 types of mulberry leaves are Chiang Mai60, Red King, White King, Black Odd Turkey, Taiwan Strawberry, Black Australia, Buriram60, Kamphaengsaeng42, Taiwan Meacho, Chiang Mai Buriram60. Figure 1 shows the images of Mulberry leaves classified into 10 classes. Figure 2 depicts the frequency



(a) Sample images of the Mulberry leaf dataset (b) Frequency of each class of the dataset

**Fig. 1** Sample dataset and frequency of each class



**Fig. 2** Model architecture

of each class of the dataset using a bar graph with ChiangMaiBuriram60 consisting of the highest number of 764 images while Buriam60 has the least number of 345 images.

### 3.2 Data Pre-processing

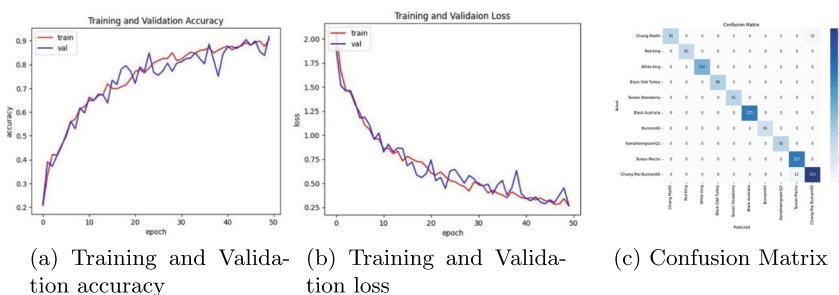
Using the Image Data Generator, these images are resized to  $224 \times 224$  pixels and set to the RGB (Red, Green, Blue) three-color channel. Several augmentations like height and width shift, shearing, zooming, rotation, and flips have also been applied to the images to generate more samples. The more unique data is available for training, the more robust our convolution model will become. After the final normalization, which ensures that all pixel values are between 0 and 1, the prepared images are clubbed together in batches of 64 images and fed into the sequential model to be classified.

### 3.3 Model Architecture

This research paper explores the classification of different species of the Mulberry Plant to aid the agricultural domain. A custom convolutional neural network is being employed to identify the 10 classes of the Mulberry leaf image dataset, namely, Chiang Mai60, Red King, White King, Black Odd Turkey, Taiwan Strawberry, Black Australia, Buriram60, Kamphaengsaeng42, Taiwan Meacho, and Chiang Mai Buriram60. Figure 3 illustrates the architecture of the model. Firstly Mulberry leaf dataset of 10 classes has been collected, then after splitting of the dataset into training, testing, and validation is performed, thirdly different augmentation techniques are employed, and finally, a custom CNN model is made to classify 10 different classes of Mulberry leaf. Figure 3 shows the workflow of the architecture of custom CNN. A mulberry leaf dataset was taken then the dataset is split into training, testing, and validation, then after several augmentation techniques have been employed to enhance the performance of the model, and then finally custom CNN model is applied to classify the data into 10 different classes.

- The proprietary CNN model has been trained to carry out multi-class classification of Mulberry leaf images.
- Predicting the correct species of Mulberry leaf from 10 classes is a difficult task considering the variation in species and their images.
- Several images in the dataset are of poor quality, often having blurry subjects, a smaller number of pixels, and corrupted sections.
- Another limitation is the restricted sample size available for the leaf images. Furthermore, there is an inequality of samples in different classes.
- Training the model with a limited dataset can lead to a problem of underfitting. To curb this issue, the images have been augmented in several ways to generate more variable data.

The custom CNN model at the heart of our implementation consists of a combination of convolutional, dense, max-pooling, and flattened layers arranged to maximize identification performance. It has been trained on 3742 colored images of leaves in



**Fig. 3** Results for training and validation accuracy with confusion matrix

various lighting conditions and the validation has been performed with the help of 1520 separate images. These images which are of shape  $224 \times 224$  pixels and have three color channels (Red, Green, and Blue) feed into our sequential model to be classified.

The model is built up of five sections or blocks that extract information from the images for processing purposes. The first three blocks consist of a convolution layer of  $3 \times 3$  kernel size and a  $2 \times 2$  max-pooling layer each. These convolution layers contain 32, 64, and 128 filters respectively, and make use of the ‘relu’ activation function. They detect patterns and collect data regarding the edges, textures, etc. The initial three layers are followed by a flattened layer and the flattened one-dimensional vector flows into a ‘relu’ activated Dense layer having 512 neurons. The final layer is also dense but it consists of only 10 neurons and uses the SoftMax activation function to compute probabilities of each of the 10 species of Mulberry leaves and make the final prediction. The model is compiled with the Adam optimizer and uses categorical cross-entropy as the loss function to compute the accuracy metric. All these layers give us a highly robust model with 44,401,226 trainable parameters.

## 4 Results

Training system hardware is equipped with 24 GB memory, 64 GB RAM, and Windows 11. Python’s Jupyter Notebook has been utilized in all training model applications. The Adam optimizer is employed with a learning rate of 0.0001 and categorical loss-entropy guided the optimization process. The model is run on 50 epochs to achieve optimal performance and accuracy. The batch size used is 32 for effective processing. 90.42% training and 92.25% validation accuracy are obtained. Similarly, 26.42% training loss and 27.10% validation loss are obtained. The custom CNN model hence developed is robust and effective. Figure 3(a), (b) depict the comparison between the training and validation accuracy and loss of the model when run over 50 epochs. Figure 3(c) shows the confusion matrix of the test dataset for 1543 images. Table 2 represents the classification report giving precision, recall, f1-score, and support of the dataset.

**Table 2** Comparison of training and validation accuracy of proposed model with different models

Model	Training accuracy (%)	Validation accuracy (%)
Sequential model	71.89	72.12
InceptionV3	76.66	70.72
VGG16	63.20	67.57
Proposed model	90.42	92.25

## 5 Discussion

To ascertain the competitiveness of our proposed convolutional neural network model in the practical application, other publicly available pre-trained models like the Sequential Model, VGG-16, Inception V3, etc., were also tested on the Mulberry Leaf dataset. Then the respective performance of these models was compared with the metrics of the custom CNN and the results can be seen below in Table 2.

After analyzing the performance metrics in the comparison table above, it is clear that the proposed custom convolutional neural network surpasses the other popular pre-trained models in terms of both validation and testing accuracy. The proposed CNN model outshines the other with an accuracy score of 92.25% which is more impressive taking into account the fact that it has fewer layers, is simpler, and trains much faster than its sizable pre-trained counterparts.

## 6 Conclusion

The paper proposes a custom-built CNN model to classify 10 different classes of Mulberry Leaf that are Chiang Mai60, Red King, White King, Black Odd Turkey, Taiwan Strawberry, Black Australia, Buriram60, Kamphaengsaeng42, Taiwan Meacho, Chiang Mai Buriram60. The performance of the model has been evaluated on different parameters like accuracy, precision, recall, and f1-score. The model attains a validation accuracy of 92.25%. This accurate and automated technique has the potential to have profound consequences. The limitation is that the current work is static, lacking the adaptability of the dynamic model where weights can be trained dynamically. Dynamic models can help improve the performance of the Mulberry leaf dataset as it will allow the system to continuously learn and adapt to the changes in leaves due to changes in season. Future endeavors may involve categorizing the dataset on the basis of environmental factors or geographical regions. Likewise deploying the model can also help botanists and farmers with their personalized usage and support which can help them to promptly identify mulberry leaves and classify them correctly.

## References

1. Rahmani ME, Amine A, Hamou MR (2015) Plant leaves classification. ALLDATA 2015 82
2. Nahiduzzaman M et al (2023) Explainable deep learning model for automatic mulberry leaf disease classification. Front Plant Sci 14
3. Kadam RA, Dhumal ND, Bhimasha Khyade V (2019) The Mulberry, *Morus alba* (L.): the medicinal herbal source for human health. Int J Curr Microbiol Appl Sci 8(4):2941–2964
4. W'aldchen J et al (2022) Towards more effective identification keys: a study of people identifying plant species characters. People Nature 4(6):1603–1615

5. Singh V, Chug A, Singh AP (2023) Classification of beans leaf diseases using fine tuned CNN model. *Procedia Comput Sci* 218:348–356
6. El Akhal H et al (2023) A novel approach for image-based olive leaf diseases classification using a deep hybrid model. *Ecol Inform* 77: 102276
7. Sharma V, Tripathi AK, Mittal H (2023) DLMC-Net: deeper lightweight multi-class classification model for plant leaf disease detection. *Ecol Inform* 75:102025
8. Paul SG et al (2023) A real-time application-based convolutional neural network approach for tomato leaf disease classification. *Array* 19:100313
9. Shewale MV, Daruwala RD (2023) High performance deep learning architecture for early detection and classification of plant leaf disease. *J Agric Food Res* 14:100675
10. Algani A, Methkal Y et al (2023) Leaf disease identification and classification using optimized deep learning. *Meas: Sens* 25:100643
11. Umamageswari A, Deepa S, Raja K (2022) An enhanced approach for leaf disease identification and classification using deep learning techniques. *Meas: Sens* 24:100568
12. Naik BN, Malmathanraj R, Palanisamy P (2022) Detection and classification of chilli leaf disease using a squeeze-and-excitation-based CNN model. *Ecol Inform* 69:101663
13. Paymode AS, Malode VB (2022) Transfer learning for multi-crop leaf disease image classification using convolutional neural network VGG. *Artif Intell Agric* 6:23–33
14. Zhao J et al (2023) A dataset of pomegranate growth stages for machine learning-based monitoring and analysis. *Data in Brief* 50:109468
15. Deng J-S et al (2023) Identification of banana leaf disease based on KVA and GR-ARNet1. *J Integrat Agric*
16. Medhi E, Deb N (2022) PSFD-Musa: a dataset of banana plant, stem, fruit, leaf, and disease. *Data Brief* 43:108427
17. Bhuiyan MAB et al (2023) BananaSqueezeNet: a very fast, lightweight convolutional neural network for the diagnosis of three prominent banana leaf diseases. *Smart Agric Technol* 4:100214
18. Arman SE et al (2023) BananaLSD: a banana leaf images dataset for classification of banana leaf diseases using machine learning. *Data in Brief* 50:109608
19. Mduma N, Leo J (2023) Dataset of banana leaves and stem images for object detection, classification and segmentation: a case of Tanzania. *Data Brief* 49:109322

# Detecting Polycystic Ovary Syndrome Through Blending Ensemble Method



Kashish Gandhi, Mansi Prajapati, Dev Bhut, and Ruhina Karani

**Abstract** The objective of this research is to improve the early detection of Polycystic Ovary Syndrome (PCOS) focusing on the challenge of early detection of PCOS and its profound implications for reproductive health. PCOS, a prevalent endocrine disorder affecting millions of women worldwide, presents a significant hurdle to achieving pregnancy due to elevated infertility rates. Recognizing the urgent need for accurate diagnostic tools, we employ advanced machine learning, deep learning, and ensemble modeling techniques on the Polycystic Ovary Syndrome (PCOS) dataset by Prasoon Kottarathil (Polycystic ovary syndrome (PCOS), Version 3, 2020. <https://www.kaggle.com/datasets/prasoonkottarathil/polycystic-ovary-syndrome-pcos>) which comprises of clinical data gathered from 10 different hospitals across Kerala, India, including 541 women patients split into 2 different groups—One who had PCOS and fertility issues, while, on the other hand, there were women who had PCOS but didn't have fertility issues. Our approach includes machine learning models like support vector machine, random forest regression, logistic regression, XGBoost regression, and gradient boosting regression, alongside ensemble techniques like blending and stacking, and deep learning model feedforward neural network coupled with ExtraTree classifier for feature selection and data balancing techniques like ADASYN and ENN from comprehensive clinical parameters. The blending ensemble model which consisted of an ensemble of random forest regression, XGBoost regression, and gradient boosting regression was our highest-performing model with 98.03% accuracy along with a precision value of 0.981, recall value of 0.983, and F1-score of 0.979. This research heralds a future where early detection of PCOS fosters informed decision-making and fosters improved reproductive outcomes.

**Keywords** Polycystic ovary syndrome · Machine learning · Deep learning · Ensemble model · Blending ensemble learning · Stacking ensemble learning

---

K. Gandhi (✉) · M. Prajapati · D. Bhut · R. Karani

Department of Computer Engineering, Dwarkadas J. Sanghvi College of Engineering, Mumbai, India

e-mail: [kashishgandhi6112003@gmail.com](mailto:kashishgandhi6112003@gmail.com)

R. Karani

e-mail: [ruhina.karani@djsce.ac.in](mailto:ruhina.karani@djsce.ac.in)

## 1 Introduction

Polycystic ovary syndrome casts a long shadow over the reproductive health of millions. According to Renato Pasquali et al. in [1], PCOS is affecting 5–10% of reproductive-age women worldwide, it stands as one of the most common endocrine disorders. For those diagnosed with PCOS, the path to parenthood can be significantly more arduous. Among women with PCOS, infertility rates are significantly higher compared to the general population, with estimates ranging from 70 to 80%. This highlights the urgent need for effective diagnostic and prognostic tools to mitigate the impact of PCOS-related infertility by early-stage detection of PCOS. We can entitle women to take command over their reproductive health and alleviate the effects of infertility by facilitating early detection of PCOS.

We leverage the power of machine learning, deep learning and ensemble modeling techniques to address this pressing challenge. By analyzing extensive clinical datasets from individuals diagnosed with PCOS, we employ a comprehensive suite of machine learning methodologies to predict PCOS in each individual. Along with stand-alone models like support vector machine, random forest regression, logistic regression, XGBoost regression, and gradient boosting regression, we have also employed an ensemble of multiple models which includes stacking of random forest regression and SVM, stacking of random forest regression and XGBoost regression and blending of random forest regression, XGBoost regression, and gradient boosting regression, working meticulously to improve the performance metrics of each model used. We have also employed deep learning models like feedforward neural network.

Along with model selection, much effort was put into the selection of features from the vast clinical data of individuals. In addition to the generic clinical data containing hormone levels (including anti-Mullerian hormone, follicle-stimulating hormone, beta-human chorionic gonadotropin hormone, luteinizing hormone, blood group type, number of follicles in each ovary; physical aspects like body mass index, height, weight, length of menstrual cycle, and various symptoms like hair growth, weight gain, skin darkening, and occurrence of pimples were utilized. Through meticulous refinement and optimization of our computational models, we transcend mere analytical tools, elevating them to be sophisticated companions in our investigative journey. With each iteration, they become more adept at unraveling the intricate complexities of PCOS, discerning subtle patterns and associations that may have escaped the human eye.

Ultimately, the culmination of this research aspires to empower on multiple fronts. Clinicians use these powerful models as a beneficial tool, allowing them to tailor treatment plans with greater efficacy. Patients too will benefit from this enhanced knowledge, gaining a deeper understanding of their own condition. By embracing the synergy of medical science and computational prowess, we envision a future where early detection of PCOS heralds a paradigm shift, enabling women to navigate the challenges of infertility with confidence and informed decision-making.

## 2 Literature Survey

The research by [2] analyzed data from the Northern Finland Birth Cohort 1966 to assess the impact of PCOS symptoms on the quality of life of women aged 31 and 46. Results revealed decreased quality of life in PCOS and hirsutism cases compared to controls, highlighting PCOS as a significant risk factor which makes it very crucial for us to detect PCOS at an early stage.

The study in [3] highlights PCOS as a multifaceted condition with symptoms ranging from irregular periods to excessive hair growth. Early detection is crucial, prompting research into machine learning algorithms and optimization techniques. This study compares classifiers and feature selection methods, achieving promising results in PCOS identification. Additionally, it explores deep learning methodologies and aims to develop a PCOS detection application for improved diagnosis and monitoring. The papers [4, 5] highlight the Rotterdam criteria for diagnosing PCOS, incorporating hormonal markers and clinical parameters. It offers key thresholds for young women undergoing puberty, enhancing diagnostic accuracy to over 90%. Utilizing this comprehensive framework facilitates precise PCOS detection and management, crucial for improving clinical outcomes. Thus, we have utilized these parameters in our research.

Subrato Bharati et al. [6] utilized machine learning algorithms on a dataset of 541 women, including 177 with PCOS, the same dataset used by us, identifying crucial predictors. This helped us understand the important features in detection of PCOS.

Various classifiers, such as gradient boosting, random forest, logistic regression, and a hybrid of random forest and logistic regression (RFLR), were applied to the dataset. Among these, RFLR exhibited the highest testing accuracy of 91.01% and a recall value of 90% when subjected to 40-fold cross-validation on the top 10 most significant features. Leveraging the same dataset, this study [7] utilizes ensemble learning algorithms such as random forest, bagging classifier, AdaBoosting, and gradient boosting. Results demonstrate gradient boosting superior performance, achieving a 91.7% accuracy.

This study [8] too evaluates various algorithms, including random forest classifier, support vector machine, XGBoost, and ensemble learning, to accurately classify PCOS in women. Results demonstrate that XGBoost classifier performs well initially, while RFC and SVC exhibit superior accuracy post-PCA, achieving an overall accuracy of 91.11%. These studies along with [9–11] encouraged us to use these models in our implementation too. Drawing from the study by [12], we applied early stopping methods in our study to mitigate overfitting concerns by the Gaussian Naive Bayes model which emerged as their top performer, achieving 100% accuracy. By adopting strategies from their research, we ensured the robustness of our model while maintaining high accuracy and reliability.

The proposed system in [13, 14] utilizes the random forest classifier and important features for model training are selected using statistical methods like chi-square and kappa coefficient in respective studies. Dr. Ashok Munjal et al. [15] highlight the significance of feature selection in PCOS diagnosis and introduce the use of the

ExtraTrees classifier for this purpose. This ensemble learning method efficiently identifies the most informative features from high-dimensional clinical data. Inspired by its effectiveness, we employed the ExtraTrees classifier in our research to enhance feature selection, ultimately improving the accuracy and precision of PCOS detection models. The authors of the research [16] advocate for the adoption of ensemble-based gradient boosting algorithms, particularly XGBoost, to enhance PCOS classification accuracy. They employ resampling techniques like SMOTE and ENN to address class imbalance and outliers in the data. Since we also faced issues of data imbalance, we incorporated ADASYN and ENN to address this issue just like these authors.

The study by Sayma Alam Suha et al. [17] introduces an innovative ensemble machine learning classification approach utilizing a stacking technique. This method combines five traditional machine learning models as base learners with one bagging or boosting ensemble model serving as the meta-learner. Inspired by this research and the research by M. Khan et al. [18] which also employed ensemble stacking methods, we decided to go ahead with ensemble models for our research. Various machine learning models, including logistic regression, support vector machine, decision tree, K-nearest neighbor, and naive Bayes, were utilized as base learners for classifying patients into PCOS and non-PCOS categories. Along with these studies another study [19] which used ensemble learning for PCOS detection, implemented voting classifier which inspired us to use the same for our ensemble stacking models.

The literature survey highlights the importance of early PCOS detection for women's quality of life and reproductive health. While existing studies explore various machine learning and ensemble techniques, there's a gap in integrating these approaches comprehensively. Existing research primarily focused on individual algorithms or ensemble methods without fully leveraging the potential synergies among different techniques. Motivated by this gap, our study aims to contribute by proposing a novel blending approach that integrates diverse machine learning models and ensemble techniques, drawing insights from a range of methodologies discussed in the literature. This research strives to improve diagnostic tools and ultimately contribute to better reproductive outcomes for affected individuals.

### 3 Data Visualization

#### 3.1 Dataset Collection

We utilized two distinct datasets to comprehensively explore the aspects of PCOS in different types of women and its implications on infertility. The clinical data, sourced from [20], provided a rich repository of physical and clinical parameters crucial for determining PCOS and associated fertility issues. This dataset, gathered from 10 different hospitals across Kerala, India, encapsulated detailed information such as the number of abortions, hormone levels (including beta-HCG, FSH, LH), physical measurements (such as hip and waist size), and various symptoms (e.g.,

hair growth, skin darkening, pimples). The clinical data was from two different types of women—one who had PCOS and fertility issues, while, on the other hand, there were women who had PCOS but didn't have fertility issues. We merged both datasets based on the common identifier to make a final dataset which was used for further processing.

### 3.2 Data Preprocessing and Analysis

In the preprocessing phase, we focused on refining the merged dataset by addressing data inconsistency issues and missing values. To address this, we imputed missing values by replacing them with the median values of their respective columns. This approach helps maintain the dataset's overall statistical characteristics while minimizing the impact of missing data on subsequent analyses, thus ensures consistency and simplifies data handling and referencing throughout the analysis pipeline. Once the data was preprocessed, we proceeded to explore the interrelationships between different variables using a correlation matrix. The correlation matrix quantifies the pairwise correlations between all feature pairs in the dataset, including the target variable "PCOS (Y/N)" and other relevant attributes. Visualizing this correlation matrix as a heatmap offers a powerful way to emphasize on the important patterns and dependencies within the data (Fig. 1).

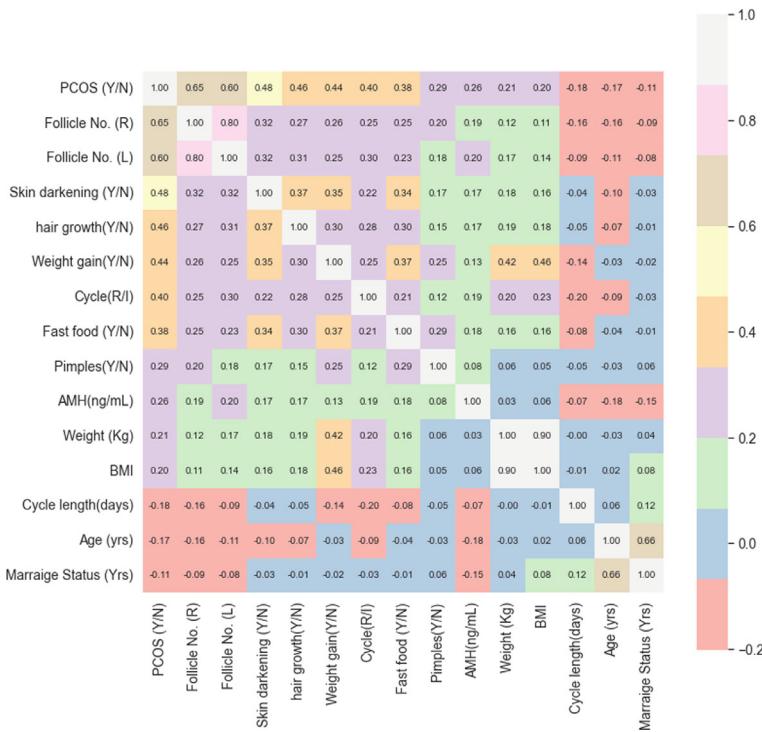
The top 10 most correlated features were AMH, Follicle No. (R), Weight (Kg), Follicle No. (L), hair growth, weight gain, skin darkening, fast food, Cycle (R/I), and pimples. After preprocessing the data, we conducted an analysis to explore how various features differ between individuals with normal reproductive health and those diagnosed with polycystic ovary syndrome (PCOS) (Fig. 2).

*Length of Menstrual Phase:* We looked at the duration of menses in terms of age between normal individuals and those with PCOS. What we found was that, among people without any health problems, period length does not change much as they get older—while women diagnosed with polycystic ovary syndrome experience increasingly longer periods over time.

*Patterns of BMI (Body Mass Index):* Then we examined body mass index or BMI over the years which indicated weight gain in PCOS patients compared to normal individuals. While BMI was steady with healthy growing up, people with PCOS BMI increased significantly as they aged.

*Patterns of Irregularity in Menstruation:* We used the "Cycle(R/I)" feature where 4 indicates irregular periods, while 2 means regularity to determine menstrual cycle irregularities. Our findings showed that normal people became more regular as they grew older but this trend was reversed for those having PCOS, i.e., irregularities increased with age among them.

*Number of Follicles:* Moreover, we investigated follicle distribution between both ovaries among women suffering from PCOS versus those having healthy reproductive

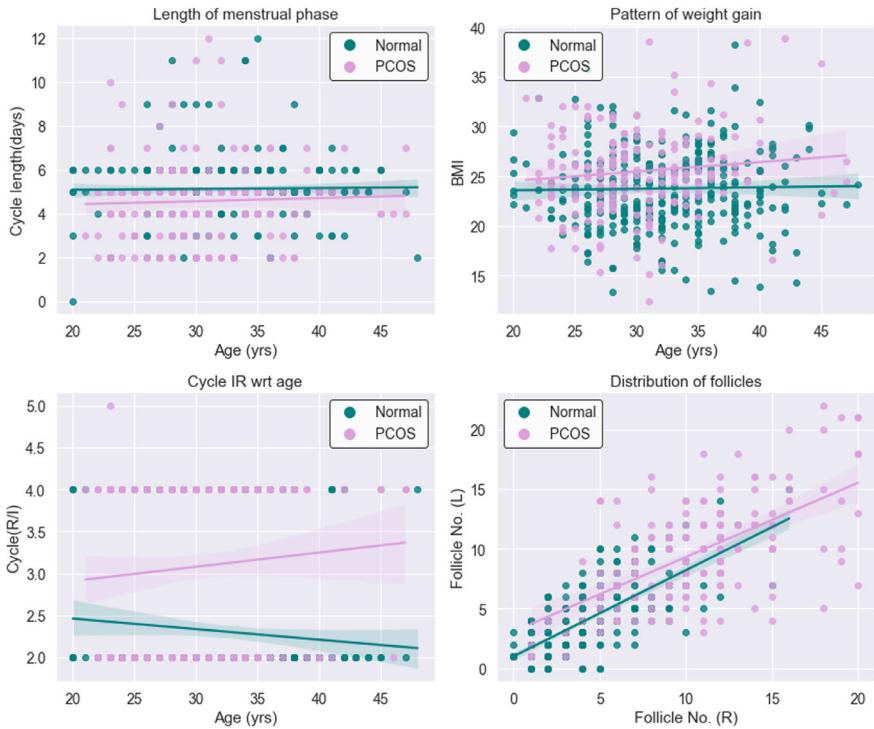


**Fig. 1** Heatmap illustrating the correlation between various features and the presence of PCOS

health status. Interestingly enough our study found out that there is a left-right imbalance of ovarian follicles in case of PCOS unlike the even distribution between left and right sides among the healthy women. We have also employed boxen plots along swarm plots for further analysis around this subject matter. The latter show individual counts of follicles on either side (left/right) under different conditions such as normal reproductive health or polycystic ovary syndrome (PCOS). These graphs reveal how many points fall into each group category along categorical axes which can be used to detect potential overlap patterns between two groups or distinct patterns between the two groups (Fig. 3).

## 4 Proposed Methodology

See Fig. 4.

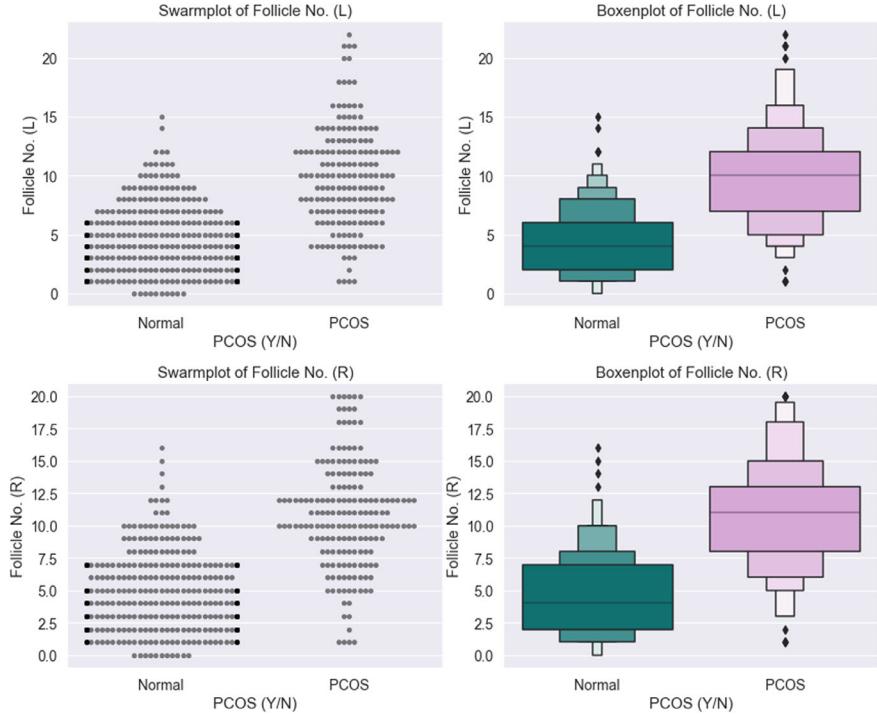


**Fig. 2** Comparative analysis of menstrual phase length, BMI patterns, menstrual cycle irregularity, and follicle distribution between normal and PCOS patients

#### 4.1 Feature Selection

Our solution begins by quantifying the importance of each feature in predicting the presence of PCOS. Getting inspired from [15], we decided to employ the ExtraTrees Classifier algorithm, which constructs multiple decision trees and averages their results to determine feature importances. Identifying the most influential features is essential for accurate PCOS detection. By prioritizing characteristics with higher relevance ratings, we make sure that the model focuses on the most important data.

Following the calculation of feature importances, we select a subset of the top features for further analysis. The top nine features were Follicle No. (R), hair growth, skin darkening, Follicle No. (L), weight gain, Cycle (R/I), fast food, pimples, and Cycle length (days). This subset comprises features with the highest importance scores, indicating their significant contribution to PCOS detection. The reason we used feature selection is because it reduces dimensionality and computational complexity while retaining the most informative features. By focusing on a subset of features, our solution enhances model efficiency and interpretability.



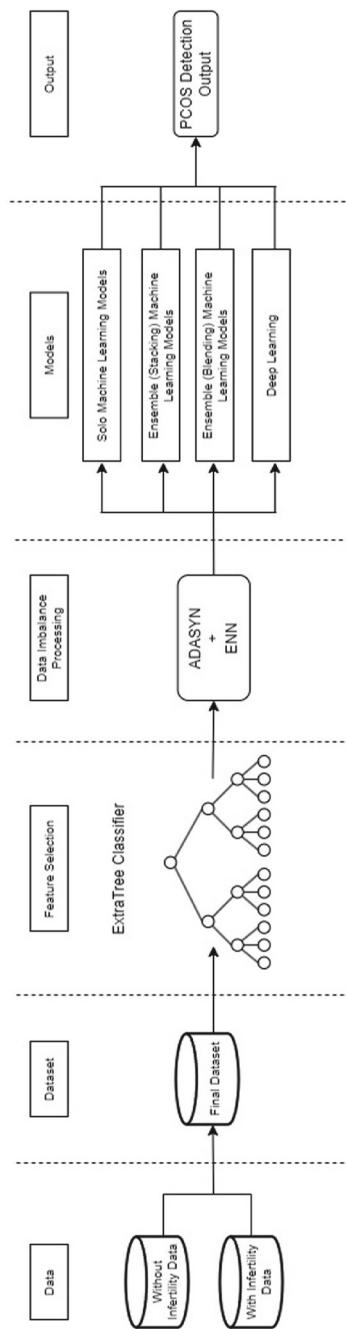
**Fig. 3** Swarm plot and box plot of follicle distribution between left and right ovaries

## 4.2 Data Imbalance Processing

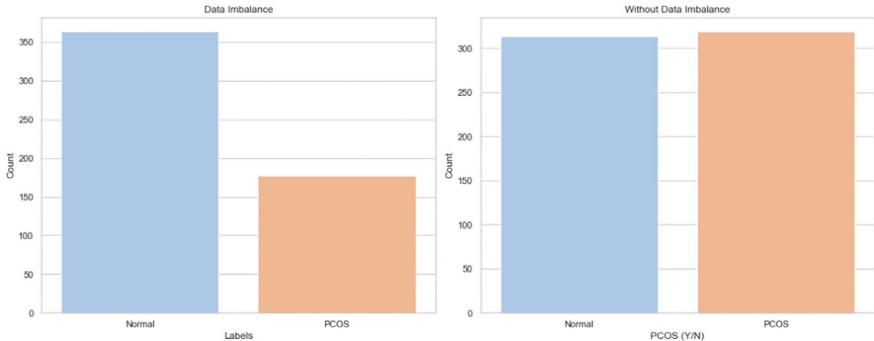
A major issue we faced during the study was class imbalance, where (PCOS-negative cases) class dominated the dataset, as shown in Fig. 5 (left), which could have caused bias model predictions if not taken care of. To mitigate this issue, we employed resampling techniques such as ADASYN and ENN. We employed ENN because the authors in [16] handled data imbalance in a similar fashion.

ADASYN generates synthetic samples for the minority class (PCOS-positive cases) based on their density distribution, promoting diversity in the dataset. ENN removes noisy instances from the majority class (PCOS-negative cases), retaining only well-separated instances, as shown in Fig. 5 (right). Balancing the class distribution ensures fair representation during model training and evaluation. By addressing class imbalance, our solution improves the model’s capability to detect both PCOS-positive and PCOS-negative cases accurately.

After addressing class imbalance, we prepare the resampled dataset for model training and evaluation. The balanced dataset comprises features and labels necessary for building and assessing the PCOS detection model. A balanced dataset facilitates unbiased model training and evaluation, leading to more reliable PCOS detection.



**Fig. 4** Pipeline used for data collection and prediction of PCOS



**Fig. 5** Distribution of labels in the original dataset showing data imbalance and distribution after balancing the data using ADASYN and ENN techniques

results. By ensuring fair representation of both classes, our solution enhances the model's generalization ability.

### 4.3 Models Used

We split the dataset into 80% training, 10% validation, and 10% test sets. We conduct tenfold cross-validation on the training set to evaluate the model's performance. Then the data was put through three different types of models—machine learning models, deep learning model, and different ensemble models which include stacking and blending methods. After training on the entire training set, we assess the model's accuracy on the validation set and the test set.

#### 4.3.1 Machine Learning Models

These models have been used due to their potential to recognize patterns and relationships in data enabling automatic decision-making. They provide predictive accuracy and scalability and simplify tasks from classification to regression. Their adaptability to different types of data makes them increasingly important for insight gain and data-informed decisions in a variety of environments.

- **Logistic Regression:** It is a statistical method for binary classification that models the probability of an input belonging to a class using the logistic function. Here, the model is trained on labeled data, learning the relationship between input features and target labels. During training, it adjusts parameters to reduce the difference between predicted probabilities and actual labels. Features like hormonal levels and medical history are utilized.

$$P(y = 1, |, x) = \frac{1}{1 + e^{-(w^T x + b)}} \quad (1)$$

- **Support Vector Machine:** It is a powerful supervised machine learning algorithm used for classification tasks. It works by finding an optimal hyperplane that effectively separates the classes in the feature space, maximizing the difference between the classes. It uses an RBF kernel trained on the training data. RBF kernels are a common choice to solve nonlinear classification challenges in SVMs. It enables the SVM to better capture complex relationships between features and target labels by moving the input data to a higher dimensional space.

$$\frac{w^T (x_{pos} - x_{neg})}{|w|} = \frac{2}{|w|} \quad (2)$$

- **XGBoost Regression (Extreme Gradient Boosting):** It is a popular gradient boosting framework known for its speed and performance in machine learning competitions. We employed the XGBoost Regressor to further enhance the predictive capability of our model for PCOS diagnosis.

$$\hat{y}_i = \sum_{k=1}^K f_k(x_i) = f_0(x_i) + \sum_{k=1}^K h_k(x_i) \quad (3)$$

- **Gradient Boosting Regression:** It is a machine learning technique for regression functions. It builds an ensemble of weak learners (typically decision trees), where each tree corrects the error of the previous one. The gradient boosting regression model is trained on training data to predict target value continuity. During training, the model minimizes the loss function by re-adapting a new model to the residual errors of the previous one. It is effective for predicting continuous outcomes and handling complex nonlinear relationships in the data.

$$\hat{y}_i = \sum_{k=1}^K f_k(x_i) = f_0(x_i) + \sum_{k=1}^K \gamma_k h_k(x_i) \quad (4)$$

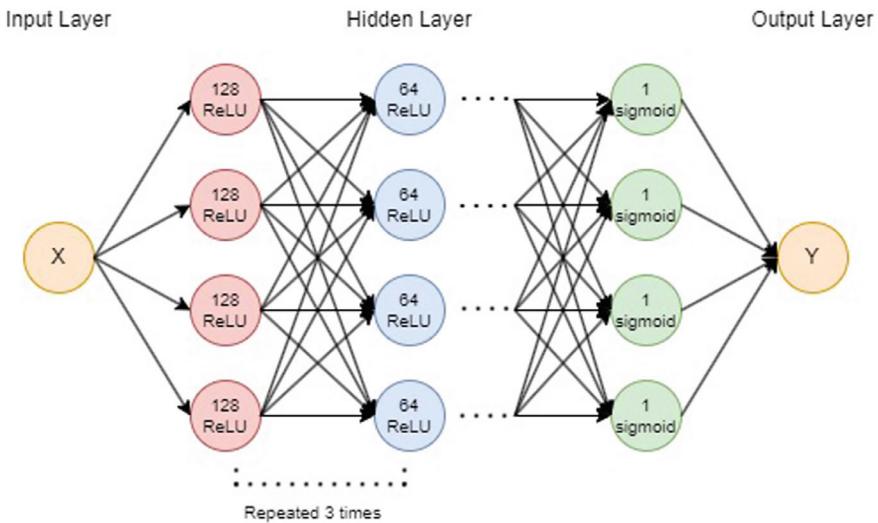
- **Random Forest Regression:** The Random Forest Regressor (RFR) represents a versatile ensemble learning algorithm, which constructs numerous decision trees during the training phase and then amalgamates them to enhance predictive accuracy while mitigating overfitting risks. In our study, we utilized RFR to model the classification task of Polycystic Ovary Syndrome (PCOS) diagnosis. The RFR model was configured with 150 estimators, employing the square root of the number of features for the maximum features parameter, and setting the minimum sample leaf as 10.

$$\hat{y}_i = \frac{1}{N} \sum_{j=1}^N f_j(x_i) \quad (5)$$

#### 4.3.2 Deep Learning Model

In our study, we utilized deep learning, a subset of machine learning, which employs neural networks inspired by the brain's structure. Specifically, we employed a Feed-forward Neural Network (FNN) architecture, as depicted in Fig. 6. This architecture had multiple layers, consisting of an input layer, several hidden layers, and an output layer. Each layer contained neurons that were connected to neurons in the next layer, allowing information to flow unidirectionally. Rectified Linear Unit (ReLU) was the activation function used in the hidden layers, which enabled non-linear transformations of the input data to extract complex features.

Here, the FNN was trained to process medical data and predict the presence of PCOS. We trained the network using the binary cross-entropy loss function and optimized its parameters using the Adam optimizer. To prevent overfitting, we employed a validation split during training, allocating 30% of the data for validation.



**Fig. 6** Feedforward neural network architecture

### 4.3.3 Ensemble Models

Ensemble modeling combines predictions from multiple individual models to achieve superior performance. By leveraging diverse perspectives and mitigating individual model weaknesses, ensemble methods enhance robustness and generalization, resulting in more accurate and reliable predictions across various applications, including PCOS detection in medical diagnosis as discussed in [17, 17, 17].

We have incorporated ensemble models like stacking and blending are employed for PCOS detection by combining predictions from multiple base classifiers. This approach leverages diverse modeling techniques to improve classification accuracy, effectively capturing complex relationships in the data. Stacking integrates predictions from different models, while blending combines outputs using weighted averages, enhancing PCOS diagnostic performance.

#### Stacking Ensemble Model Approach

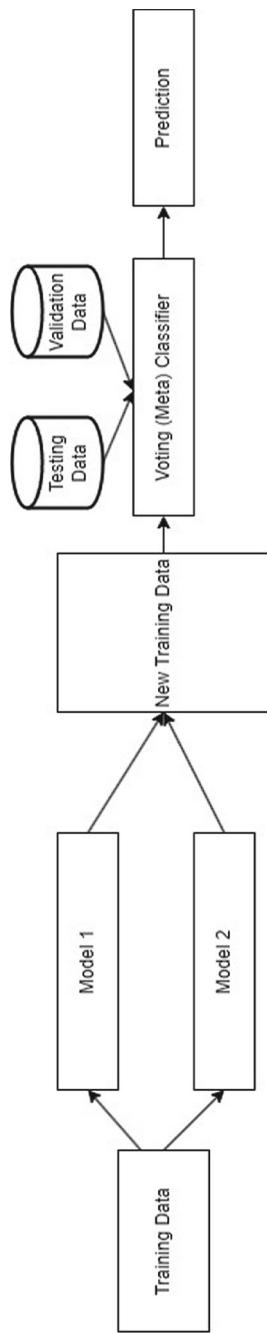
- **Random Forest Regression + XGBoost Regression:** To leverage the strengths of both random forest and XGBoost regressors, we constructed an ensemble model by combining them using a voting classifier. The ensemble model aggregates predictions from multiple individual models and selects the class label that receives the majority of votes. In our ensemble setup, random forest regressor and XGBoost regressor were used as base estimators.
- **Random Forest Regression + Support Vector Machine:** To leverage the strengths of both random forest and support vector machine, we constructed an ensemble model by combining them using a voting classifier. The ensemble model aggregates predictions from multiple individual models and selects the class label that receives the majority of votes. In our ensemble setup, random forest regressor and support vector machine were used as base estimators (Fig. 7).

#### Blending Ensemble Model Approach

Blending is an ensemble method that combines predictions from multiple base models to improve overall performance. It operates by training a meta-model on the predictions of the base models, thereby learning to effectively weigh their outputs. This technique mitigates the individual weaknesses of each base model, leading to enhanced predictive accuracy and robustness.

In our study, blending was employed to optimize the PCOS detection model's performance. Given the heterogeneity of the dataset and the diverse characteristics of PCOS, leveraging multiple learning algorithms was crucial. By integrating predictions from random forest, XGBoost, and gradient boosting classifiers, the blending ensemble method facilitated a comprehensive understanding of the data landscape. This approach allowed us to capitalize on the unique strengths of each base model while minimizing the impact of their limitations.

The blending process involved training base models such as random forest, XGBoost, and gradient boosting on the training data to generate individual predictions. These predictions were then used as features to train a meta-model, logistic



**Fig. 7** Concept diagram of Model 1 + Model 2 stacking ensemble learning

regression, on a validation set. Finally, the trained meta-model was applied to the test set, composed of blended predictions from the base models, to make accurate PCOS predictions. This strategy ensured that the ensemble model could effectively capture intricate patterns within the data, leading to superior performance in PCOS detection (Fig. 8).

## 5 Results

Our study investigated several machine learning approaches for PCOS detection, revealing compelling findings. The blending ensemble method emerged as the most successful, boasting an impressive accuracy of 98.03%, precision of 0.981, recall of 0.983, and F1-score of 0.979. This method, which amalgamates predictions from diverse base models, demonstrated remarkable predictive power. Moreover, the deep learning model showcased notable efficacy, achieving an accuracy of 96.11%, precision of 0.960, recall of 0.921, and F1-score of 0.961. Additionally, our analysis of stacking ensemble techniques highlighted the effectiveness of combining random forest and support vector machine, achieving a commendable accuracy of 97.62%, precision of 0.975, recall of 0.977, and F1-score of 0.978. These outcomes underscore the potential of ensemble methods in augmenting predictive accuracy and reliability for intricate medical diagnostic tasks, such as PCOS detection. Such robust performance across diverse methodologies signals promising avenues for enhancing clinical decision-making processes in reproductive health.

The ROC (Receiver Operating Characteristic) and precision–recall curves graphically represent the blending ensemble model’s performance. The ROC curve depicts the trade-off between true positive rate (sensitivity) and false positive rate, demonstrating the model’s ability to distinguish between PCOS and normal cases at different thresholds. On the other hand, the precision–recall curve visualizes the balance between precision (positive predictive value) and recall (sensitivity), indicating the model’s effectiveness in correctly identifying PCOS cases while minimizing false positives. Higher area under the curve (AUC) values signifies superior model performance (Tables 1 and 2).

Furthermore, we visualized the training progress and convergence of the FNN by plotting the loss and accuracy against the number of epochs. These visualizations offer a comprehensive understanding of the model’s learning dynamics and convergence behavior (Figs. 9 and 10).

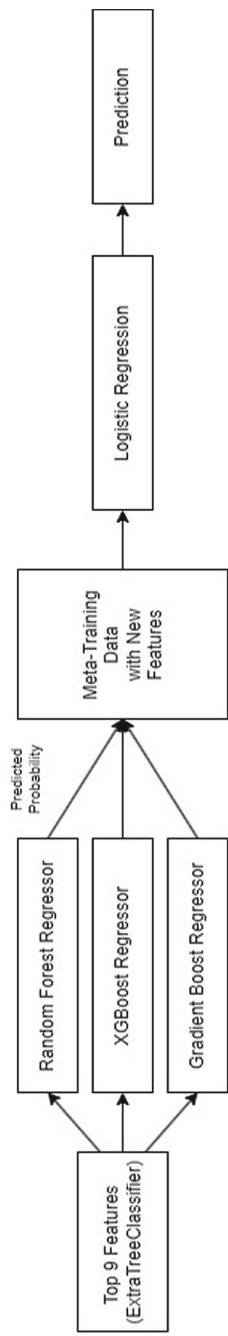


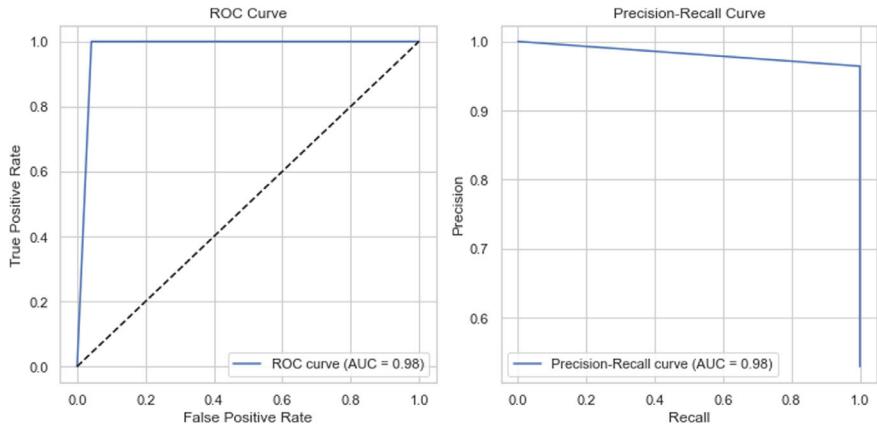
Fig. 8 Concept diagram of blending ensemble learning

**Table 1** Performance metrics (precision, recall, F1-score, and accuracy) of implemented PCOS detection models

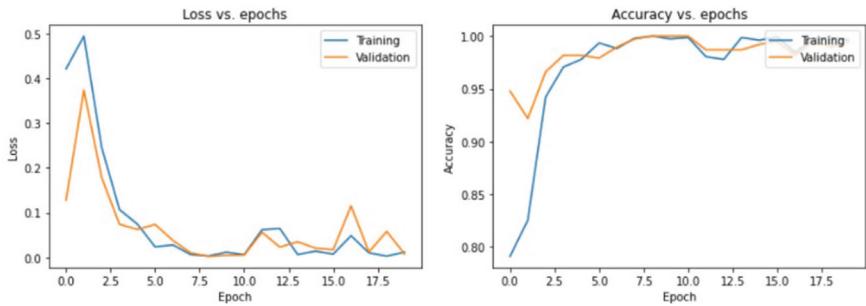
Metrics assessing model performance					
Model	Precision	Recall	F1-Score	Validation accuracy (%)	Test accuracy (%)
Logistic regression	0.886	0.883	0.884	87.50	88.21
Support vector machine	0.945	0.943	0.944	90.23	94.43
XGBoost regression	0.951	0.954	0.948	90.57	95.34
Gradient boosting regression	0.951	0.947	0.948	90.48	95.32
Feedforward neural network	0.960	0.921	0.961	92.08	96.11
Random forest regression	0.973	0.972	0.974	92.14	96.64
Stacking (Random Forest + XGBoost)	0.974	0.971	0.972	92.74	96.93
Stacking (Random Forest + SVM)	0.975	0.977	0.978	93.89	97.62
<i>Blending (Random Forest + XGBoost + Gradient Boosting)</i>	<b>0.981</b>	<b>0.983</b>	<b>0.979</b>	<b>94.88</b>	<b>98.03</b>

**Table 2** Comparison with previous studies

Paper	Method	Accuracy (%)
[10]	KNN	91.00
[6]	RFLR	91.01
[8]	RFC + SVC	91.11
[19]	Voting soft classifier	91.12
[7]	Gradient boosting	91.70
[9]	Linear regression	92.60
[11]	CatBoost classifier	93.00
[18]	Ensemble stacking	93.74
[3]	Grid search CV	94.00
[16]	XGBoost	95.83
[14]	Random forest	96.00
[17]	Stacking gradient boosting enhanced with meta learner	96.00
<i>Our proposed methodology</i>	<i>Blending ensemble method</i>	<b>98.03</b>



**Fig. 9** Performance evaluation graphs depicting the ROC and precision–recall curve for the blending ensemble model



**Fig. 10** Training loss and accuracy versus epochs for feedforward neural network

## 6 Conclusion

Our research represents a significant advancement in the field of Polycystic Ovary Syndrome (PCOS) detection, focusing exclusively on the analysis of clinical data gathered from ten different hospitals across Kerala, India, using advanced computational techniques like ExtraTree classifier for feature selection and data balancing techniques like ADASYN and ENN. By harnessing machine learning, deep learning algorithms, and ensemble methodologies, we meticulously examined diverse clinical datasets to identify patterns indicative of PCOS. The key highlight of our study lies in the blending ensemble method, which emerged as the most effective strategy for PCOS detection, achieving an exceptional accuracy rate of 98.03% along with a precision value of 0.981, recall value of 0.983, and F1-score of 0.979. This remarkable performance underscores the synergistic power of combining predictions from multiple base models, enhancing the model's predictive accuracy and reliability.

Furthermore, our research contributes to the burgeoning field of reproductive health by offering insights that can revolutionize PCOS diagnosis and patient care. Through the integration of cutting-edge computational methodologies with clinical data analysis, we pave the way for early detection and intervention strategies that can significantly improve outcomes for individuals affected by PCOS. By providing a robust framework for PCOS detection, we aim to empower healthcare professionals with tools that enable timely intervention and personalized care, ultimately enhancing the quality of life for individuals with PCOS.

## 7 Future Scope

For future research one could explore the integration of additional data modalities, such as genetic markers and lifestyle factors, to further enhance predictive accuracy. Moreover, advancements in deep learning techniques, particularly in the field of medical image analysis, could enable the development of CNN architectures specifically tailored for ultrasound imagery, offering a comprehensive diagnostic approach. Additionally, further investigations could focus on incorporating data from diverse populations to enhance the model's robustness and applicability across various demographic backgrounds. These efforts would contribute to ensuring that the developed model can effectively support PCOS detection across a wide range of patient populations, ultimately advancing personalized healthcare interventions in a more inclusive manner. Overall, the future holds immense potential for refining PCOS detection methodologies and advancing personalized healthcare interventions.

## References

1. Pasquali, R, Stener-Victorin E, Yildiz B, Duleba A, Hoeger K, Mason H, Homburg R, Hickey T, Franks S, Tapanainen J, Balen A, Abbott D, Diamanti-Kandarakis E, Legro R (2010) PCOS Forum: research in polycystic ovary syndrome today and tomorrow. *Clin Endocrinol* 74:424–433. <https://doi.org/10.1111/j.1365-2265.2010.03956.x>
2. Karjula S et al (2020) Population-based data at ages 31 and 46 show decreased HRQoL and life satisfaction in women with PCOS symptoms. *J Clin Endocrinol Metab* 105(6):1814–1826
3. Yadav N, A RK, Pande SD (2024) Comparative analysis of polycystic ovary syndrome detection using machine learning algorithms. *EAI Endorsed Trans Perv Health Tech* [Internet]. 2024 Mar. 26, 10. Accessed 28 Apr 2024
4. Adla YA et al (2021) Automated detection of polycystic ovary syndrome using machine learning techniques. In: 2021 Sixth international conference on advances in biomedical engineering (ICABME). IEEE
5. Khashchenko E et al (2020) The relevant hormonal levels and diagnostic features of polycystic ovary syndrome in adolescents. *J Clin Med* 9(6):1831
6. Bharati S, Podder P, Rubaiyat Hossain Mondal M (2020) Diagnosis of polycystic ovary syndrome using machine learning algorithms. In: 2020 IEEE region 10 symposium (TENSYMP). IEEE

7. Lakshmi MJ et al (2023) Prediction of PCOS and PCOD in women using ML algorithms. In: Choudrie J, Mahalle PN, Perumal T, Joshi A (eds) ICT for intelligent systems. ICTIS 2023. Smart innovation, systems and technologies, vol 361. Springer, Singapore. [https://doi.org/10.1007/978-981-99-3982-4\\_9](https://doi.org/10.1007/978-981-99-3982-4_9)
8. Sethi R, Vishwakarma DK, Ganguly S, Ray R (2023) A comparative study on different machine learning algorithms to detect PCOS. In: 2023 14th international conference on computing communication and networking technologies (ICCCNT), Delhi, India, pp 1–7. <https://doi.org/10.1109/ICCCNT56998.2023.10307174>
9. Hdaib D, Jo W, Mustafa W, Al-azzawi W, Alkhayyat A, Alquran H (2022) Detection of Polycystic Ovary Syndrome (PCOS) using machine learning algorithms. <https://doi.org/10.1109/IICETA54559.2022.9888677>
10. Tanwani N (2020) Detecting PCOS using machine learning. <https://doi.org/10.13140/RG.2.2.10265.24169>
11. Modi N, Kumar Y (2024) Detection and classification of polycystic ovary syndrome using machine learning-based approaches. In: IEEE international conference on interdisciplinary approaches in technology and management for social innovation (IATMSI), Gwalior, India, pp 1–6. <https://doi.org/10.1109/IATMSI60426.2024.10503222>
12. Nasim S et al (2022) A novel approach for polycystic ovary syndrome prediction using machine learning in bioinformatics. IEEE Access 10:97610–97624
13. Thakre V et al (2020) PCCare: PCOS detection and prediction using machine learning algorithms. Biosci Biotechnol Res Commun 13(14):240–244
14. Hassan M, Mirza T (2020) Comparative analysis of machine learning algorithms in diagnosis of polycystic ovarian syndrome. Int J Comput Appl 175. <https://doi.org/10.5120/ijca2020920688>
15. Munjal A, Khandia R, Gautam B (2020) A machine learning approach for selection of polycystic ovarian syndrome (PCOS) attributes and comparing different classifier performance with the help of WEKA and PyCaret. Int J Sci Res:59–63
16. Inan MSK et al (2021) Improved sampling and feature selection to support extreme gradient boosting for PCOS diagnosis. In: 2021 IEEE 11th annual computing and communication workshop and conference (CCWC). IEEE
17. Suha SA, Nazrul Islam M (2023) Exploring the dominant features and data-driven detection of polycystic ovary syndrome through modified stacking ensemble machine learning technique. Heliyon 9(3)
18. Khan M, Nila F, Tabasaum N, Suha SA, Islam MN (2023) Enhancing PCOS prediction: a system based on ensemble machine learning techniques. In: 2023 IEEE 9th international women in engineering (WIE) conference on electrical and computer engineering (WIECON-ECE), Thiruvananthapuram, India, pp 108–113. <https://doi.org/10.1109/WIECON-ECE60392.2023.10456492>
19. Bharati S, Podder P, Mondal MRH, Surya Prasath VB, Gandhi N (2022) Ensemble learning for data-driven diagnosis of polycystic ovary syndrome. In: Abraham A, Gandhi N, Hanne T, Hong TP, Nogueira Rios T, Ding W (eds) Intelligent systems design and applications. ISDA 2021. Lecture Notes in Networks and Systems, vol 418. Springer, Cham. [https://doi.org/10.1007/978-3-030-96308-8\\_116](https://doi.org/10.1007/978-3-030-96308-8_116)
20. KottaRathil P (2020, July) Polycystic ovary syndrome (PCOS), Version 3. <https://www.kaggle.com/datasets/prasoonkottarathil/polycystic-ovary-syndrome-pcos>. Accessed 11 July 2020

# Verbatim: Empowering Seamless Communication with Authentic Voice Translation



C. I. Chandas Patel, Swimpy Pahuja, Rutika Babasab Patil, R. Yeshas, Arati Chabukswar, and M. S. Pratap

**Abstract** Language limitations continue to be a major obstacle to successful cooperation and communication in the modern world. Even with advances in technology, current translation technologies sometimes fall short of capturing the subtleties of spoken language, which results in awkward or erroneous translations that impede natural communication. People are more likely to come across languages other than their own in varied and multicultural environments, where there is a clear communication gap. Thus, a solution that facilitates smooth translation while maintaining the speaker's uniqueness and nuance is desperately needed to foster real understanding and connections across linguistic barriers. This article proposes a novel mobile application called Verbatim, which signifies a paradigm change in the way we think about cross-language communication. Verbatim, in contrast to conventional techniques, takes advantage of spoken language's natural properties to facilitate rapid translation, making it accessible and user-friendly for people all over the world. It, therefore, creates new avenues for cross-cultural communication and collaboration by crossing language barriers and increasing global connectivity and accessibility for all. Furthermore, its novel method goes beyond simple translation, too, as it aims to maintain the naturalness and nuance of the user's speech.

**Keywords** Voice translation · Real-time translation · Google cloud platform · xTTS · Coqui AI · User-centered design · Language barriers · Communication

## 1 Introduction

Voice translation is a game-changing technology that allows spoken language to be translated into many languages in real-time. In today's globalized world, its capacity to facilitate seamless communication across linguistic borders makes it invaluable [1]. Voice translation encourages inclusivity, mutual understanding, and cooperation between multilingual individuals in multicultural settings. This article highlights

---

C. I. Chandas Patel · S. Pahuja (✉) · R. B. Patil · R. Yeshas · A. Chabukswar · M. S. Pratap  
School of Computing and Information Technology, REVA University, Bengaluru, India  
e-mail: [swimpy.pahuja@gmail.com](mailto:swimpy.pahuja@gmail.com)

Verbatym, a cutting-edge smartphone application that combines cutting-edge technology to seamlessly translate spoken words in real-time, revolutionizing communication. With its user-friendly UI and customized voice translations, it aims to eliminate language barriers by providing consumers with an intuitive and customized experience. Its development methodology is based on a dedication to user-centered design and agile approaches. Its fundamental technology is built on Google Cloud services such as Google Text-to-Speech and Google Cloud Translate API. Furthermore, the program makes use of Conqui AI's xTTS [2] to enhance its functionality and guarantee precision and effectiveness in language translation. The application prioritizes an intuitive interface, designed to deliver a seamless and captivating language translation solution that resonates with users worldwide. Moreover, its innovative approach extends beyond mere translation, as it endeavors to preserve the authenticity and nuances of the user's voice. Through advanced speech recognition and machine learning algorithms, it captures the essence of spoken language, ensuring that translations remain faithful to the speaker's unique vocal characteristics.

Verbatym's infrastructure is seamlessly hosted on Google Cloud Platform (GCP), harnessing its robust suite of services for transcription and translation. Leveraging GCP's state-of-the-art speech recognition capabilities, it accurately transcribes spoken words into text, laying the foundation for precise translation. With GCP's powerful translation services, Verbatym swiftly converts the transcribed text into the desired language, ensuring accuracy and fluency in communication. Table 1 provides the comparative differences between existing voice translation systems and Verbatym application. Moreover, the application integrates a custom model from xTTS, developed by Coqui AI, to mimic the voice of the user in various languages. This innovative approach enhances user experience by providing a more personalized and natural interaction. The xTTS model utilizes advanced neural network architectures to generate high-quality synthetic speech, closely resembling the user's voice characteristics. As a result, it offers a truly immersive and authentic communication experience across different languages, fostering genuine connections and understanding between individuals from diverse linguistic backgrounds.

**Table 1** Comparison between existing voice translation applications and proposed application

Features	Existing voice translation apps	Verbatym
Real-time translation	Yes	Yes
User voice authenticity	Limited preservation of voice authenticity	Advanced voice authenticity of the speaker
Language support	Varies by app	Supports commonly spoken languages
Ease of use	Varies by app	User-friendly interface with simple navigation
Reliability	Varies by app	Decent system availability with minimal service interruptions

## 2 Related Work

A multi-speaker neural text-to-speech system was presented by Luong et al. [3]. They discovered that even with highly unbalanced data, combining data from multiple speakers to train a multi-speaker Text-to-Speech (TTS) model can produce synthetic speech of better quality and stability when the available data of a target speaker is insufficient. Additionally, they found that using an ensemble multi-speaker model enhanced speech quality even more, particularly for speakers who are underrepresented. Similarly, Gumma et al. [4] provided text to text-to-speech translation system for the Mundari language. Overall, their research shows how well multi-speaker models work to improve the quality of synthetic speech and handle speaker imbalance problems. Li et al. addressed the difficulties related to voice conversion, with a special emphasis on problems like speaker information leakage and the need for substantial annotated data [5].

They presented a new method for waveform reconstruction that makes use of the Voice Input Translation System (VITS) end-to-end infrastructure, as well as creative techniques for obtaining clean content information without the need for text annotation. The need for improved fidelity was highlighted by Binkowski et al.'s proposal of high-fidelity voice synthesis using adversarial networks. They unveiled GAN-TTS, a sophisticated generative adversarial network designed specifically for text-to-speech conversion. Their unique method includes discriminators that assess audio segments for fidelity and realism, as well as a conditional feed-forward generator. Results showed that GAN-TTS could produce high-fidelity speech that was on par with top models and that its feed-forward architecture also made effective parallelization possible [6]. The training of speaker verification models is more efficient than the prior Tuple-based End-to-End (TE2E) loss function, according to a new loss function named Generalized End-to-End (GE2E) [7]. In contrast to TE2E, the GE2E loss does not require initial example selection, instead, it prioritizes upgrading of the network by highlighting difficult cases at each training step. They also introduce the multi-reader approach, which makes it easier to adapt to changing domains by allowing a more accurate model to be trained that supports a variety of keywords and dialects. To solve alignment issues, Battenberg et al. compared additive energy-based and GMM-based methods for location-relative attention mechanisms [8]. Their results demonstrated how well GMM attention and Dynamic Convolution Attention (DCA) generalize to longer utterances while retaining naturalness for shorter ones, providing encouraging avenues for the development of strong long-form speech synthesis systems. According to Casanova et al. [9], zero-shot voice conversion and zero-shot multi-speaker TTS are being revolutionized. They applied creative changes for zero-shot multi-speaker and multilingual instruction, building on the core of the VITS paradigm. Their method demonstrated promise in target languages with limited single-speaker datasets as well, achieving state-of-the-art results in zero-shot multi-speaker TTS and equivalent outcomes in zero-shot voice conversion on the VCTK dataset, respectively. Kim et al. [10] introduced a pioneering Text-to-Speech (TTS) model that doesn't rely on external aligners. Glow-TTS autonomously

discovers optimal alignments between text and speech representations, ensuring robust TTS performance for long utterances and enabling fast, diverse, and controllable speech synthesis. It outperforms autoregressive models like Tacotron 2 in speed while maintaining comparable speech quality and is easily scalable to multi-speaker scenarios, highlighting its versatility. The proposed model strives to revolutionize the language translation landscape by providing an effortless and intuitive user experience. Its emphasis on seamless real-time voice translation, along with its integration with Google Cloud services, establishes a solid foundation for its commitment to innovation and user-centric design.

### 3 Proposed Work

#### 3.1 Assumptions

The model is optimized for seamless real-time translation and therefore, works on the following assumptions:

- It has been assumed that there will be consistent internet connectivity where this application is utilized.
- The platform is designed to be compatible with a wide range of devices, from smartphones to tablets, to ensure accessibility for users across various platforms.
- It has been assumed that our users possess a basic level of proficiency in mobile applications, allowing them to navigate Verbatim's interface and utilize its features effectively. Accurate speech recognition technology is also integrated into it to transcribe and translate spoken words, resulting in an enhanced user experience.
- Lastly, user privacy and data protection have been taken on top priority by implementing robust security measures to safeguard against unauthorized access or breaches.

These assumptions serve as the foundation for model's reliability and effectiveness as a language-translation tool.

#### 3.2 Methodology

Verbatim utilizes a thorough methodology that encompasses multiple facets of quality assurance and testing, with the ultimate goal of maintaining strict standards of performance, reliability, and user satisfaction throughout all stages of the application's development process. This approach to quality assurance is characterized by its breadth and depth, extending through every phase of development, from

initial concept ideation to final product deployment. At every step, meticulous attention is paid to ensuring that the application meets and surpasses established benchmarks for functionality, usability, and overall performance. This involves a systematic and rigorous examination of various aspects of the application, including its user interface, functionality, and backend infrastructure, among others. By adopting this comprehensive approach to quality assurance and testing, Verbatim strives to deliver a product of unparalleled quality and reliability, enhancing user satisfaction and fostering long-term success in the competitive landscape of language translation applications. The model relies on several key dependencies to support its functionalities and ensure seamless operation. These dependencies include advanced speech recognition technology, which forms the backbone of the application's ability to accurately transcribe spoken words. Moreover, the application is reliant on robust internet infrastructure to facilitate real-time communication and data transmission between users and the application's servers. In conjunction with its reliance on Google Cloud APIs for translation and transcription tasks, it requires access to extensive language databases to support its multilingual capabilities effectively. These databases serve as repositories of linguistic data, enabling the application to accurately translate spoken words across various languages. Furthermore, it also incorporates a user feedback mechanism to gather valuable insights and suggestions from its user base, facilitating continuous improvement and refinement of its features and functionalities. Lastly, cross-platform compatibility is essential for Verbatim to reach a wide audience of users across different devices and operating systems. By ensuring compatibility with various platforms, including smartphones, tablets, and computers, it aims to maximize its accessibility and usability for users worldwide. Overall, these dependencies play a vital role in supporting the functionalities and contributing to its effectiveness as a language translation application.

### ***3.3 Requirements***

The proposed system exhibits functional as well as non-functional requirements which are explained as under:

- Functional requirements: Verbatim's functional requirements encompass a range of essential capabilities designed to deliver a seamless and intuitive user experience. Foremost among these requirements is the real-time voice translation capability, enabling users to engage in natural and fluid conversations across different languages. In addition to translation functionality, it also necessitates user registration and profile management features, allowing users to create accounts, personalize their profiles, and manage their preferences within the application. An intuitive user interface design is also paramount, ensuring that users can navigate the application effortlessly and access its features with ease. Integration with Google Cloud services is another critical functional requirement, enabling

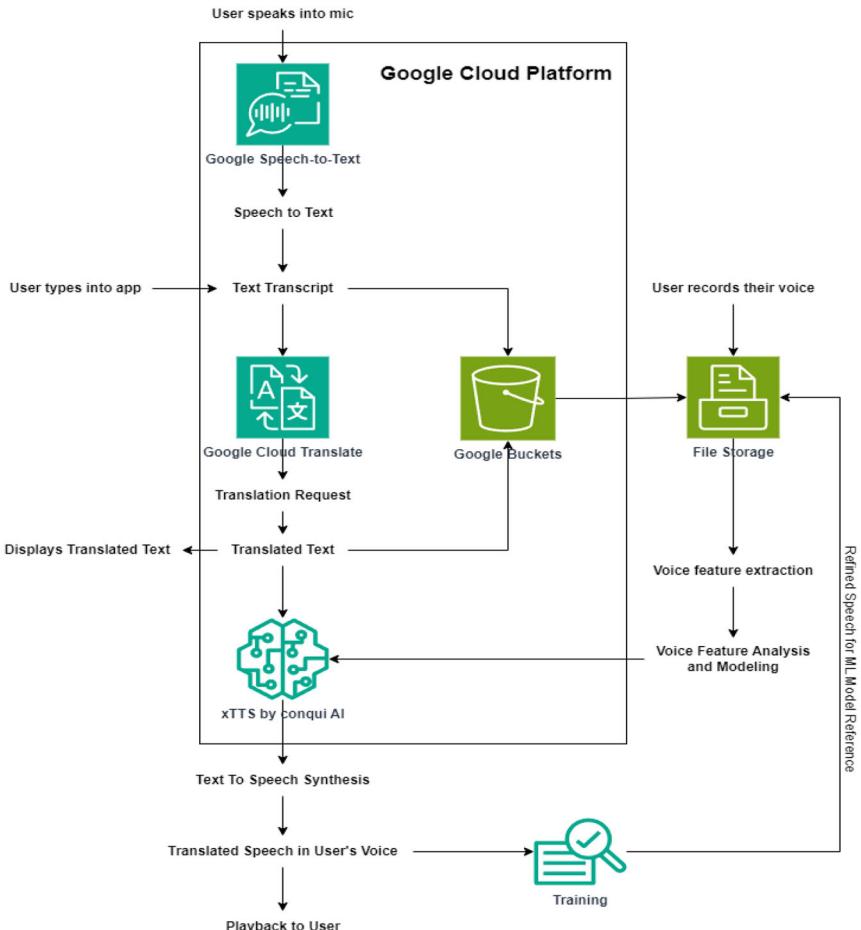
Verbatim to leverage advanced cloud-based resources for translation and transcription tasks. By meeting these functional requirements, the model aims to deliver a user-friendly and efficient language translation solution that meets the diverse needs of its user base.

- Non-functional Requirements: Verbatim's non-functional requirements focus on aspects of performance, reliability, and usability that are critical to ensuring a high-quality user experience. Low latency performance is paramount, as it ensures that translations occur rapidly and without significant delay, facilitating smooth and natural communication between users. Scalability is also essential, allowing it to accommodate a growing user base and increasing demand for its services without sacrificing performance or reliability. Reliability is another key non-functional requirement, ensuring that the application functions consistently and predictably under various conditions. Cross-platform compatibility is vital to maximizing accessibility, allowing users to access the features across a range of devices and operating systems seamlessly. Adaptability to diverse linguistic needs is crucial, as it enables to support a wide range of languages and dialects, catering to the linguistic diversity of its user base. Usability and onboarding are also significant non-functional requirements, ensuring that users can easily navigate the application and understand its features and functionality. Lastly, comprehensive documentation for users and maintenance personnel is essential for providing guidance and support, ensuring that users can make the most of Verbatim's capabilities while facilitating efficient maintenance and troubleshooting processes. By adhering to these non-functional requirements, the model aims to deliver a robust and reliable language translation solution that meets the highest standards of performance and usability.

### ***3.4 Proposed System***

As depicted in Fig. 1, the whole process of the proposed translation application can be described in the following steps:

- User Interaction: The process begins when a user interacts with the system. There are three possible inputs—User Speaks into Mic, where the user's spoken words are captured by a microphone; User Types into App, providing an alternative input method where the user can input text directly into an application; and User Records Their Voice, allowing the user's voice to be recorded and stored for further processing.
- Speech-to-Text Conversion: The captured voice data is sent to Google Speech-to-Text, which converts spoken language into written text (transcription), resulting in a Text Transcript that represents what the user said.
- Translation: The Text Transcript is then sent as a Translation Request to Google Cloud Translate, where it is translated into the desired target language, producing the Translated Text.



**Fig. 1** Proposed system architecture

- **Voice Feature Extraction:** Simultaneously, the recorded voice is stored in a File Storage system, where it undergoes Voice Feature Extraction. This process involves analyzing specific features of the voice data to extract relevant information. This extraction is done through a built-in speaker encoder that is embedded in xTTS, the text-to-speech service used. This encoder computes the speaker's voice features like phonation, pitch, loudness, and rate.
- **Voice Feature Analysis and Modeling:** The extracted voice features are further analyzed and modeled to capture unique characteristics of the user's voice, enhancing the overall accuracy and quality of the synthesized speech.
- **Text-to-Speech Synthesis:** Both paths (translated text and voice features) converge at xTTS by coqui AI. Here, the Translated Text is synthesized into speech using

the user's voice characteristics, resulting in translated speech that closely matches the user's voice.

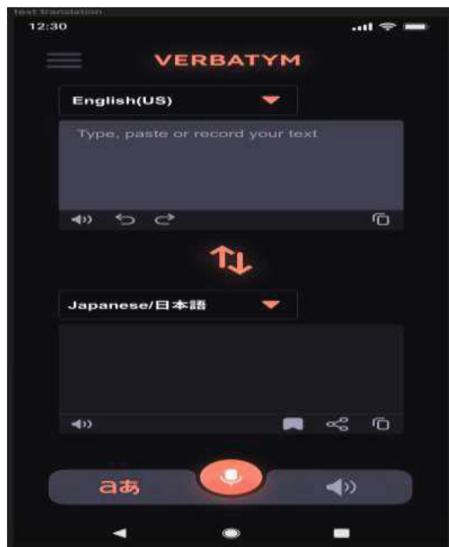
- Playback to User: Finally, the synthesized speech is played back to the user, completing the loop of interaction and providing the user with the desired output.
- Training: Once the final voice is translated and synthesized, it enters the training phase. The refined speech is stored in the file storage for future reference or to continuously train the ML model, aiming to enhance voice accuracy with each iteration. This iterative training process helps improve the quality and accuracy of the synthesized voices over time.

## 4 Results and Discussions

Verbatim's language translation application involves a range of outputs and milestones that are essential for its development and deployment. The first deliverable is the creation of a mobile application prototype, which acts as its initial version of user interface and functionality. The created prototype and the working of the application have been shown in Figs. 2 and 3, respectively.

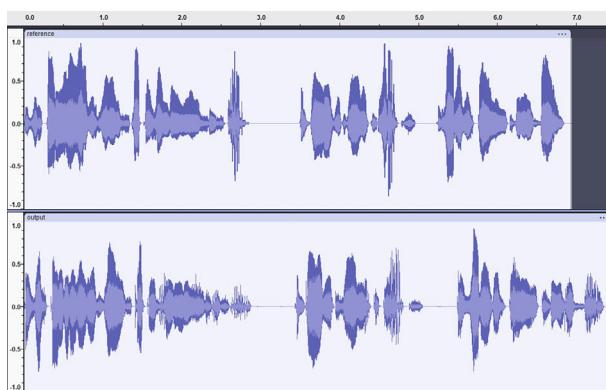
A voice cloning model has also been tested for the proposed application with varying metrics throughout the development process, the results are found to be satisfactory. An open source, streamable voice cloning model called xTTS from Coqui AI has been utilized, which provided consistent outcomes. As an example, using the text query "The Hispaniola was rolling scuers under in the ocean swell. The booms were tearing at the blocks, the rudder was banging to and fro," the resulting audio clips represented by their waveform have been depicted in Fig. 4.

**Fig. 2** Prototype of application



**Fig. 3** Working application

Despite being approximately one second longer than the reference audio it was cloned from, the visual representation of the waveform confirms that the output audio possesses similar attributes such as pitch, amplitude, and other relevant properties. The differences in duration between the reference and output audios do not seem to have a significant impact on the overall quality of the output audio. Verbatim also focuses on refining machine learning models specifically tailored for voice cloning across multiple languages. These improved models contribute to the application's ability to accurately replicate the nuances and characteristics of a user's voice when translating spoken words.

**Fig. 4** Visual representation of the difference between the reference audio and output audio for the same text query by the fine-tuned TTS model

It strives to achieve the highest levels of authenticity and naturalness in voice translations through meticulous fine-tuning and optimization, enhancing the overall user experience. Seamless integration with Google Cloud Platform (GCP) services is another crucial deliverable for Verbatim, facilitating the application's access to powerful cloud-based resources for translation and transcription tasks. This integration ensures that the application operates efficiently and reliably, even under high demand and varying usage conditions. Our Android application has a chat-like feature where when one user speaks in one language the language is detected translated and displayed in the target language, i.e., the second user's language and vice versa. While this is happening the users can hear the voice of the other user in the translated language. To elaborate on this process, when a user speaks in a language the app records the voice, converts it to text, the speaker's language is detected, the text is translated to the target language and our ML model uses this text to create an audio of the speaker's voice but spoken in the translated language, this process will be working concurrently to provide an authentic conversational experience despite the language barrier.

## 5 Conclusion

Verbatim represents a groundbreaking advancement in language translation technology, offering real-time voice translation while preserving the authenticity of the speaker's voice. Through its innovative approach and user-centric design, it has positioned itself as a transformative force in global communication, bridging linguistic divides and fostering meaningful connections among individuals worldwide. By leveraging cutting-edge technologies and adopting agile practices, it has ensured that its application delivers an unparalleled user experience that resonates with users across diverse linguistic and cultural backgrounds. With its intuitive interface, personalized voice translations, and steadfast commitment to authenticity, it stands poised to revolutionize the way people communicate and interact in an increasingly interconnected world. Verbatim presents a distinctive solution for real-time voice translation that upholds the speaker's voice's genuineness. By integrating with Google Cloud services and xTTS from Conqui AI, it ensures precise and effective translation procedures. When a user speaks into the application, it translates their words into another language while preserving the speaker's unique vocal characteristics, tone, and subtleties. This pioneering feature not only adds a layer of personalization but also enhances the authenticity of the communication experience, making it more natural and engaging for both parties. Moreover, it features an in-app feedback system to encourage continuous improvement and user engagement, reflecting its dedication to providing a seamless and user-friendly language translation solution.

## References

1. Kumar GK, Praveen SV, Kumar P, Khapra MM, Nandakumar K (2023) Towards building text-to-speech systems for the next billion users. In: IEEE international conference on acoustics, speech and signal processing (ICASSP), June, pp 1–5
2. Coqui (2023) XTTS models—Coqui documentation. Accessed 14 Jan 2024
3. Luong HT, Wang X, Yamagishi J, Nishizawa (2019) Training multi-speaker neural text-to-speech systems using speaker-imbalanced speech corpora. In: INTERSPEECH, September, pp 1303–1307
4. Gumma V, Hada R, Yadavalli A, Gogoi P, Mondal I, Seshadri V, Bali K (2024) MunTTS: a text-to-speech system for Mundari. arXiv preprint arXiv: 2401.15579
5. Li J, Tu W, Xiao L (2023) Freevc: towards high-quality text-free oneshot voice conversion. In: IEEE international conference on acoustics, speech and signal processing (ICASSP), June, pp 1–5
6. Bińkowski M, Donahue J, Dieleman S, Clark A, Elsen E, Casagrande N, Simonyan K (2019) High-fidelity speech synthesis with adversarial networks. arXiv preprint [arXiv:1909.11646](https://arxiv.org/abs/1909.11646)
7. Wan L, Wang Q, Papir A, Moreno IL (2018) Generalized end-to-end loss for speaker verification. In: IEEE international conference on acoustics, speech and signal processing (ICASSP), April, pp 4879–4883
8. Battenberg RE, Skerry-Ryan RJ, Mariooryad S, Stanton D, Kao D, Shannon M, Bagby T (2020) Location-relative attention mechanisms for robust long-form speech synthesis. In: IEEE international conference on acoustics, speech and signal processing (ICASSP), May, pp 6194–6198
9. Casanova YE, Weber J, Shulby CD, Junior AC, Gölge E, Ponti MA (2022) Yourtts: towards zero-shot multi-speaker TTS and zeroshot voice conversion for everyone. In: International conference on machine learning. PMLR, June, pp 2709–2720
10. Kim J, Kim S, Kong J, Yoon S (2020) Glow-TTS: a generative flow for text-to-speech via monotonic alignment search. Adv Neural Inf Process Syst 33:8067–8077

# Facial Emotion Detection Using Artificial Intelligence



Ananya Debnath , Vineet Singh , Bramah Hazela , and Shikha Singh 

**Abstract** A basic component of daily interactions and human behavior is the expression of emotions. Emotional recognition is a computer-based system that identifies the feelings associated with objects that are identified. Algorithms capable of detecting, extracting, and assessing these facial expressions will enable automated sentiment analysis of individuals in images and videos. Therefore, facial movements, voice, behavior, or physiological data may all be used to identify emotions. The development of affective computing can benefit from the precise identification of human emotions. The study focuses on creating a facial expression recognition model to identify and analyze an individual's emotions with the help of Convolutional Neural Network (CNN). The research methodology is efficient, shows successful performance and an accuracy of 72.79%. It provides a deeper understanding of current emotion recognition systems and can assist researchers in selecting suitable algorithms and datasets.

**Keywords** Artificially intelligence (AI) · Facial emotion recognition (FER) · Haar cascade algorithm · Convolutional neural networks (CNN) · Deep learning (DL)

---

A. Debnath () · V. Singh · B. Hazela · S. Singh

Amity School of Engineering and Technology, Amity University, Lucknow, India  
e-mail: [ananya.debnath@s.amity.edu](mailto:ananya.debnath@s.amity.edu)

V. Singh  
e-mail: [vsingh@lko.amity.edu](mailto:vsingh@lko.amity.edu)

B. Hazela  
e-mail: [bhazela@lko.amity.edu](mailto:bhazela@lko.amity.edu)

S. Singh  
e-mail: [ssingh8@lko.amity.edu](mailto:ssingh8@lko.amity.edu)

## 1 Introduction

Humans have the innate ability to recognize emotions. However, the challenge of emotion recognition must be overcome if we are to build a robot with humanoid features that can communicate and react with its fellow human partners. There is a plethora of really beneficial real-world applications for computers that can identify human emotions [1]. According to the scholar Mehrabian, verbal language expresses 7% of human interaction information, vocal component expresses 38%, and facial emotion expresses 55% [2, 3]. As a result, in direct contact with someone, gestures, and facial expressions provide the most significant data for psychological assessment. Furthermore, due to the underlying factors that influence the development of facial expressions, automatic facial expression analysis can be a difficult undertaking. One such feature is the existence of variances in the texture of the skin, age, gender, hairstyle, ethnicity, and other factors that have a great influence on the facial look that results from the experiments. Furthermore, various individuals display their emotions in different ways [1, 3]. Human–Computer Interaction is an evolving topic of study, and there is a pressing need to provide machines with intellect so that they can comprehend and react to situations in the same way that humans do enhance organic connections [4].

## 2 Literature Survey

Psychologist Paul Ekman's work is essential to the emotion recognition technology's advancement [5]. Ekman's Facial Action Coding System (FACS) serves as the foundation for the majority of research on facial emotion recognition. This method map an emotion space to the muscles of the face. In Swedish anatomist Carl-Herman Hjortsj was the one who initially created this categorization. In 1978, the facial expression recognition technology emerged [6]. The Facial Expression Recognition (FER) System is a non-invasive and newly established technology. Face detection, picture standardization, extraction of features, evaluation of features, and categorization were critical for the evolution of the FER system [7]. Table 1 shows the different methods, models by researchers that are used for emotion recognition in recent years and have achieved promising accuracy for different datasets.

### 2.1 Convolution Neural Network

Convolutional Neural Networks (CNN) have demonstrated remarkable potential in analyzing images since their inception in the early 1990s. Image recognition, processing tasks require leveraging layers like convolutional, pooling, and fully connected layers for feature extraction from raw pixel data. Transfer learning uses

**Table 1** Different methods and model with their accuracy

Dataset	Author	Method + Model	Accuracy (%)
CK+	Alsharekh [8]	[Viola-Jones]	90.98
	Liliyana [9]	[DCNN] +	92.81
	Dudekula and Purnachand [10]	[NVIDIA Jetson Nano + OpenCV]	95.60
		[NVIDIA Jetson Nano + VGG-19]	98.40
	Assiri and Hossain [11]	[CNN + ARs] + Nose tip precision	94.51
	Gupta et al. [12]	[VGG19]	90.14
AFFECTNET	Han and Hu [13]	[VGG19]	99.20
	Haider et al. [14]	[TLF-ResNet18] SVM + 7 Emotions}	66.37
FER2013	Alsharekh [8]	[Haar Cascade]	89.20
	Gupta et al. [12]	[Inception-V3]	89.11
	Riyantoko et al. [15]	[Haar Cascade]	92
	Pham et al. [16]	ResNet	76.82
MMI	Haider et al. [14]	[TLF-ResNet18] + SVM	99.02
SAVEE	Singh et al. [17]	[3DCNN + ConvLSTM]	98.83

a pre-trained framework on a big dataset like ImageNet has achieved state-of-the-art performance measure in terms of accuracy, precision, recall, F1-score, or other relevant metrics as shown in Table 2. The benefits of the pre-trained CNN models are transfer learning, better performance, improved feature extraction, and improved efficiency.

**Table 2** Comparison of accuracies in ImageNet dataset of pre-trained CNN models

Model	Top-1 accuracy	Top-5 accuracy	Parameters	Author
Inception-v3	77.90	93.70	23M	Szegedy et al. [18]
ResNet-50	74.90	92.10	25M	He et al. [19]
VGG 16	71.30	90.10	138M	Simonyan and Zisserman [20]
VGG19	71.30	90	143M	
AlexNet	63.30	84.6	62M	Krizhevsky et al. [21]
DenseNet	76.39	93.34	8M	Huang et al. [22]
Xception	79	94.5	22M	Chollet [23]
GoogleNet	74.80	92.20	23M	Szegedy et al. [24]
MobileNet-V2	71.3	90.1	35M	Howard et al. [25]

**Fig. 1** Sample images from FER2013 dataset



### 3 Dataset

This study utilized the FER2013 Dataset, which contains facial images of 7 emotions. FER2013 dataset contains 35,887 Gy-scale images measuring  $48 \times 48$  pixels as shown in Fig. 1.

In FER2013 dataset, the training set has 28,709 photos, while the public and private test sets each have 3,589 images [26]. Limitations are that the dataset is not evenly balanced, biased distributions, less frequent expressions, such as disgust, might be harder than with extracting and labeling smiles, disproportionate number of samples in a single group might lead to incorrect conclusions. Image's face is not perfectly straight and, in the center, but is intentionally made to make recognition more challenging.

### 4 Research Methodology

In this study, we build a CNN model for emotion detection and haar cascade algorithm is also used for face detection.

#### 4.1 Data Preprocessing

The picture is modified by reducing every pixel's value by 255.0, ensuring that all values are between 0 and 1, and normalizing them. This allows the model to merge rapidly throughout training. Categorizing the dataset into the following groups: training, validation, and test sets to assess the effectiveness of the model on unknown data. The algorithm increased the model's accuracy and generated a standardized and reshaped sample image with dimensions  $(48, 48)$  from the array.

## 4.2 *Implementing CNN Architecture*

A convolutional neural network model for emotion detection and classification is developed using Keras application programming interface. The model consists of seven layers—four convolutional layers, two fully connected layers, and a softmax layer. Kernel size is  $3 \times 3$  and the activation function is ReLU. Flattening of the resultant matrix is performed. Convolution is the most important component in the CNN network, which also builds a local connection instead of connecting all the pixels, for every value in that kernel, we compute the scalar product as we traverse through the input. All of the pixels in this fully connected layer are fully linked to every node in the subsequent layer, include all the necessary information for identification, utilizing the feedforward framework. The purpose of the pooling procedure is to reduce the feature maps' spatial resolution. The goal of dropout is to prevent overfitting. Batch normalization is used to enhance training stability and speed neural networks. Softmax in CNN performs classification and turns unprocessed scores into distributions of probability across many classes, enabling simpler predictions and applicable to multi-class categorization. Figure 2 shows the summary presents rows of information for each model layer, including the type of layer, output shape, and the trainable parameters. The output shape of each layer is supplied as a tuple, with the first value representing the batch size and the other values representing the output tensor size.

## 4.3 *Training the Model*

The batch size and number of epochs were examined carefully until the best accuracy was obtained with minimum overfitting and a fair training period. Accuracy is used as the performance parameter to evaluate model efficiency during training. The model used a loss function of cross-categorical entropy and Adam optimizer. The batch size was set to 128, and the number of epochs was 48.

## 4.4 *Haar Cascade Algorithm*

Haar cascade is a machine learning approach used for machine learning and object detection. This algorithm uses edge or line detection features to get an accurate expression of the user's face. It is used in the methodology for the purpose of face detection. Haar features are calculated to categorize the sections of an image face detection. Integral pictures significantly accelerate the computation of these Haar features. AdaBoost effectively selects the most useful features and instructs the classifiers to utilize those. At each stage, the classifier evaluates a set of haar-like features computed at different scales and locations in the image [27].

Layer (type)	Output Shape	Param #
Conv2D	(None, 48, 48, 64)	640
batch normalization (BatchNormalization)	(None, 48, 48, 64)	256
activation (Activation)		
max_pooling2d (MaxPooling2D)	(None, 48, 48, 64)	0
dropout (Dropout)	(None, 24, 24, 64)	0
conv2d_1 (Conv2D)	(None, 24, 24, 128)	204928
batch normalization_1 (BatchNormalization)	(None, 24, 24, 128)	512
activation_1 (Activation)	(None, 24, 24, 128)	0
max_pooling2d_1 (MaxPooling2D)	(None, 12, 12, 128)	0
dropout_1 (Dropout)	(None, 12, 12, 128)	0
conv2d_2 (Conv2D)	(None, 12, 12, 512)	590336
batch normalization_2 (BatchNormalization)	(None, 12, 12, 512)	2048
activation_2 (Activation)	(None, 12, 12, 512)	0
max_pooling2d_2 (MaxPooling2D)	(None, 6, 6, 512)	0
dropout_2 (Dropout)	(None, 6, 6, 512)	0
conv2d_3 (Conv2D)	(None, 6, 6, 512)	2359808
batch normalization_3 (BatchNormalization)	(None, 6, 6, 512)	2048
activation_3 (Activation)	(None, 6, 6, 512)	0
max_pooling2d_3 (MaxPooling2D)	(None, 3, 3, 512)	0
dropout_3 (Dropout)	(None, 3, 3, 512)	0
flatten (Flatten)	(None, 4608)	0
dense (Dense)	(None, 256)	1179904
batch normalization_4 (BatchNormalization)	(None, 256)	1024
activation_4 (Activation)	(None, 256)	0
dropout_4 (Dropout)	(None, 256)	0
dense_1 (Dense)	(None, 512)	131584
batch normalization_5 (BatchNormalization)	(None, 512)	2048
activation_5 (Activation)	(None, 512)	0
dropout_5 (Dropout)	(None, 512)	0
dense_2 (Dense)	(None, 7)	3591

---

Total params: 4,478,727  
 Trainable params: 4,474,759  
 Non-trainable params: 3,968

**Fig. 2** CNN model architecture

## 5 Experimental Result

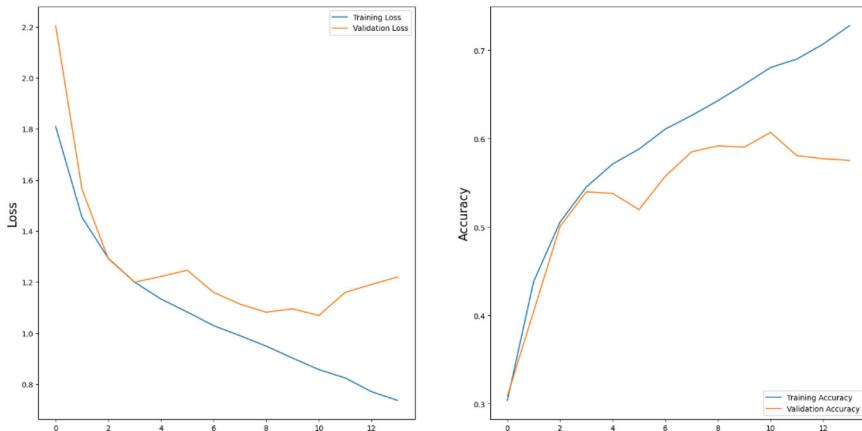
To recognize and analyze images and real-time movement from webcams, the emotion detection classifier needs to be initialized. Once loaded, the model is capable of making predictions. The system uses Python and OpenCV libraries, as well as the webcam. The results are verified by running the cell in Jupyter Notebook and waiting for the webcam light to turn on. The suggested technique saves computation time, improves validation accuracy, minimizes loss, and provides performance evaluation. The high accuracy score of (72.79%) indicates a successful model performance of emotion prediction of our convolutional neural network. Figure 3 shows the various emotions detected from facial expressions using webcam. Emotions like happy, neutral, sad, surprise, and anger were easily recognizable by our model, whereas emotions like disgust were not easily recognizable. In some instances, emotions like fear and surprise were misinterpreted. This can be a result of the dataset being unevenly balanced; a disproportionate number of samples in a single group might lead to incorrect conclusions. Less frequent expressions, such as disgust, were particularly challenging.



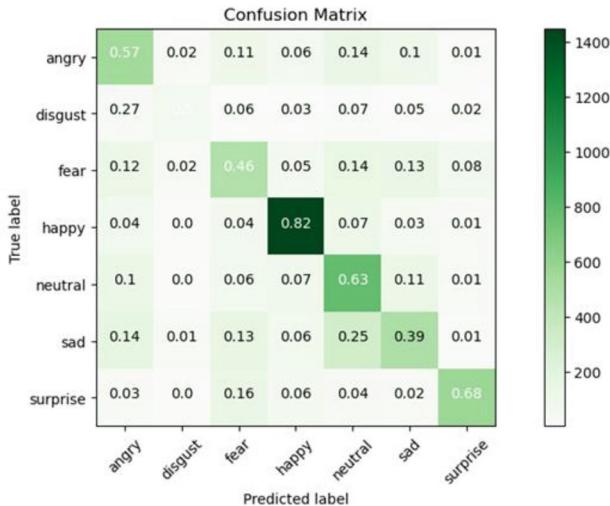
**Fig. 3** Various emotions detected using the camera

Figure 4 shows the loss and accuracy during validation and training. Figure 5 shows the confusion matrix and the performance of the classification for each emotion.

Figure 6 shows the evaluation metrics: overall accuracy, F1-score, recall, and precision. These metrics determine how specific and clear the model is and measure the actual positive results from a collection of predictions made by any algorithm.



**Fig. 4** Loss and accuracy graph for validation and testing



**Fig. 5** Confusion matrix

**Fig. 6** Evaluation metrics

Confusion Matrix				
Normalized confusion matrix				
Classification Report				
	precision	recall	f1-score	support
angry	0.50	0.57	0.53	958
disgust	0.46	0.50	0.48	111
fear	0.46	0.46	0.46	1024
happy	0.82	0.82	0.82	1774
neutral	0.51	0.63	0.56	1233
sad	0.53	0.39	0.45	1247
surprise	0.80	0.68	0.74	831
accuracy			0.61	7178
macro avg	0.58	0.58	0.58	7178
weighted avg	0.61	0.61	0.61	7178

## 6 Comparison of Method

Comparing the accuracy of our proposed method with accuracy of the existing method/model on the FER2013 Dataset is shown in Table 3.

## 7 Conclusion and Future Work

The study explores the technique of facial expression recognition using the FER2013 dataset, one of the popular benchmarks. This project aims to standardize CNN model development by linking parameters, metrics and ability to recognize emotions

**Table 3** Accuracy of existing methods on the FER2013 dataset

Author	Type	Accuracy (%)
Proposed method	CNN + Haar cascade	72.79
Giannopoulos et al. [28]	GoogleNet	65.20
Fard et al. [29]	RESNET50 Xception + Ad-Corre	68.25 72.03
Kusuma et al. [30]	Single-model	69.40
Tang [31]	CNN + SVM	71.20
Pramerdorfer et al. [32]	Resnet VGG	72.40 72.70
Khaireddin et al. [33]	Single model	73.28
Pham et al. [16]	Single-model Ensemble of 6	74.14 76.82

utilizing the presented model. Our subsequent research will improve the system's performance and create more accurate predictions for practical applications. The suggested model has undergone usability and accuracy testing, yielding satisfactory result. Our approach achieves promising results with low face registration mistakes, rapid execution duration, and an excellent correct recognition rate leading to significant performance improvements. The study found that the learning approach is equally important as model selection. We must make improvements in specific areas like number and layout of convolutional layers and dense layers, the dropout percentage in dense layers. Future research would involve using deeper CNN architecture, having more layers with stable configuration, addition of more training data to enhance accuracy, focus on face categorization, and optimal merging of depth as well as color data, after examining approaches for dealing with expression fluctuation expand the dataset by collecting and annotating a larger and more diverse set of more varied datasets representing a broader spectrum of emotions, raising the quantity and variety of training data, transfer learning techniques would be applied to enhance pre-trained models. Exploring the integration of advanced optimization algorithms, regularization techniques, adapting transfer learning methods.

## References

1. Wang J, Yin L, Wei X, Sun Y (2006) 3D facial expression recognition based on primitive surface feature distribution. In: IEEE computer society conference on computer vision and pattern recognition (CVPR'06), New York, NY, USA
2. George A, Wimmer H, Rebman Jr CM (2020) Artificial intelligence facial expression recognition for emotion detection: performance and acceptance 21(4):81–91
3. Amsel T (2019) An urban legend called: “The 7/38/55 Ratio Rule.” Europ Polygraph 13:95–99. <https://doi.org/10.2478/ep-2019-0007>
4. Kumari J, Rajesh R, Pooja KM (2015) Facial expression recognition: a survey. In: Proceedings of IEEE translation and pattern analysis machine intelligence conference

5. Ekman P, Sorenson ER, Friesen WV (1969) Pan-Cultural elements in facial displays of emotions. *Science* 164:86–88
6. Yang M-H, Kriegman D, Ahuja N (2002) Detecting faces in images: a survey. *Pattern analysis and machine intelligence. IEEE Trans on* 24:34–58. <https://doi.org/10.1109/34.982883>
7. Prince EB, Martin KB, Messinger DS (2015) Facial action coding system
8. Alsharekh MF (2022) Facial emotion recognition in verbal communication based on deep learning. *Sensors* 22(16). <https://doi.org/10.3390/S22166105>
9. Liliyana DY (2019) Emotion recognition from facial expression using deep convolutional neural network. *IOP Conf. Series: J Phys: Conf Series* 1193
10. Dudekula U, Purnachand N (2023) Analysis of facial emotion recognition rate for real-time application using NVIDIA Jetson Nano in deep learning models. *Indones J Electr Eng Comput Sci*
11. Assiri B, Hossain MA (2023) Face emotion recognition based on infrared thermal imagery by applying machine learning and parallelism. *Math Biosci Eng* 20(1):913–929
12. Gupta S, Kumar P, Tekchandani RK (2023) Facial emotion recognition based real-time learner engagement detection system in online learning context using deep learning models. *Multim Tools Appl*
13. Han B, Hu M (2023) The facial expression data enhancement method induced by improved StarGAN V2. *Symmetry*
14. Haider I, Yang H, Lee G, Kim S (2023) Robust human face emotion classification using triplet-loss-based deep CNN features and SVM
15. Riyantoko P, Sugiantoro, Hindrayani KM (2021) Facial emotion detection using Haar-cascade classifier and convolutional neural networks. *J Phys: Conf Series*
16. Pham L, Vu TH, Tran TA (2020) Facial expression recognition using residual masking network. In: 25th international conference on pattern recognition (ICPR). IEEE, Milan, Italy, pp 4513–4519
17. Singh R, Saurav S, Kumar T, Saini R, Vohra A, Singh S (2023) Facial expression recognition in videos using hybrid CNN & ConvLSTM. *Int J Inf Technol* (Singapore)
18. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna ZB (2016) Rethinking the inception architecture for computer vision
19. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition
20. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition
21. Krizhevsky A, Sutskever I, Hinton GE (2017) ImageNet classification with deep convolutional neural networks
22. Huang G, Liu Z, van der Maaten L, Weinberger K (2017) Densely connected convolutional networks
23. Chollet F (2016) Xception: deep learning with depthwise separable convolutions
24. Szegedy C, Liu W, Jia Y, Sermanet P, Reed SE, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2013) Going deeper with convolutions. In: 2015 IEEE conference on computer vision and pattern recognition (CVPR), pp 1–9
25. Howard A, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, Andreetto M, Adam H (2017) MobileNets: efficient convolutional neural networks for mobile vision applications
26. Carrier P-L, Courville A (2017) The facial expression recognition 2013 (FER-2013) dataset
27. Viola P, Jones M (2001) Rapid object detection using a boosted cascade of simple features. In: Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001, Kauai, HI, USA
28. Giannopoulos P, Perikos I, Hatzilygeroudis I (2018) Deep learning approaches for facial emotion recognition: a case study on FER-2013. In: Advances in hybridization of intelligent methods. Springer: Cham, Switzerland
29. Fard AP, Mahoor MH (2022) Ad-Corre: adaptive correlation-based loss for facial expression recognition in the wild. *IEEE Access*
30. Putra Kusuma G, Jonathan APL (2020) Emotion recognition on FER-2013 face images using fine-tuned VGG-16. *Adv Sci Technol Eng Syst J*

31. Tang Y (2013) Deep learning using linear support vector machines
32. Pramerdorfer C, Kampel M (2016) Facial expression recognition using convolutional neural networks: state of the art. arXiv
33. Khaireddin Y, Chen ZL (2021) Facial emotion recognition: state of the art performance on FER2013. ArXiv abs/2105.03588

# News-Scope: Intelligent Categorization of News Content Using Machine Learning



Rahul Karmakar<sup>ID</sup>, Mrinal Manna, Sidhartha Bakuli,  
Rajayshree Bhattacharyaa, and Avijit Das

**Abstract** News articles serve as a window into various subjects' updates and details. From politics to sports, world affairs to business insights, entertainment to education, news articles cover them all. People read them daily to learn about situations or to get updates on existing matters. Nowadays, automated news classification techniques are applied widely for fast and efficient news categorizations. Models created using ML, like Bayesian models, RF, Logistic Regression, and SVM make it easier to classify a list of articles into different text categories with high accuracy. In the “News-Scope” project, we step into building efficient and reliable ML models to segregate news articles to their respective categories (World, Sports, Business, and Science/Technology News) by understanding their content. The models built shall facilitate news publication media, providing them with valuable insights like useful news recommendations to their users and tools to optimize their content dissemination strategies in the digital age. It also shall be of great help to the online news readers to search and find articles on particular categories promptly.

**Keywords** ML · Bayesian models · RF · Logistic regression · SVM · News-Scope

## 1 Introduction

In today's world, content on different topics is available online. Online news platforms are no different. With faster publication, a greater audience reach and well-maintained news articles, news media houses are broadening their operations on the internet. The vast domain of news publication is, however, dependent on human intervention for sorting any news article into its proper category (Politics, World Affairs, Business, etc.) Supervised models like Naive Bayes (probabilistic model), random forest (ensemble model), logistic regression (linear classifier), and support vector machines can be applied for classifying different text categories with a good amount of precision. However, applying the right text preprocessing techniques and

---

R. Karmakar (✉) · M. Manna · S. Bakuli · R. Bhattacharyaa · A. Das  
The University of Burdwan, Bardhaman, West Bengal, India  
e-mail: [rkarmakar@cs.buruniv.ac.in](mailto:rkarmakar@cs.buruniv.ac.in)

vectorizing the text (using count vectorizer, TF-IDF vectorizer, etc.) in a correct way are preliminaries before training a supervised model. And, N-gram approaches often are right choices to deepen the knowledge base of the model being trained on how frequently some terms follow or are preceded by other terms. For creating a high-precision supervised text classification model for news categorization in “News-Scope” project, the authors have explored a variety of essential text preprocessing techniques (such as shortening texts, stopwords removal, special character and digit removal, etc.), feature engineering techniques (count vectorizer, TF-IDF vectorizer, and N-gram representations) and then trained the model using a large corpus of articles to category labeled dataset—AG News, to incorporate textual varieties in different categories of news content in the model. The linear classifier model—Logistic Regression and probabilistic model—Naive Bayes perform best while the Random Forest and SVM follows them in terms of accuracy which also show a satisfiable high precision.

Many notable works were being carried out in the text classification domain which is a superset of news-content classification domain. In a research work it is found that models such as RF and NB outperform SVM and lexicon-based methods, concluding RF, NB, and Artificial Neural Network (ANN) to be a good choice for text classification works [1]. Dictionary-based text classification approaches have also been tested in a research where it is found that supervised machine learning (SML) outperforms dictionaries in text classification [2]. In another work, a Label-Name-Only Text Classification model was proposed in which they used only the label names of different classes to train text classification models on unlabeled data, achieving approximately 90% accuracy on benchmark datasets. The LOTClass model achieved an accuracy of 86.4% on the AG News dataset [3].

We have identified some limitations in the existing works of the same domain, such as word-level models facing difficulties with unknown tokens at test time, complexity of very deep language models, sparsity, scalability issues, etc., and have tried to overcome these limitations, leveraging machine learning (ML) models that allow automated feature extraction and classification.

The rest of the manuscript is organized as follows: Sect. 2 presents related works. In Sect. 3, the problem domain, materiel, and methods have been discussed. The proposed model is represented in Sect. 4. The observations are highlighted in Sect. 5 followed by conclusion in Sect. 6.

## 2 Related Works

A survey introduces both traditional ML and DL models for text classification, on popular sentiment analysis, news classification, and other datasets, and provides evaluation metrics and future research challenges [4]. Gasparetto et al. [5] thoroughly analyze text classification methodologies, emphasizing on each text classification step providing insights into various English datasets and preprocessing techniques

that enhance feature extraction in DL models. Hu et al. [6] discuss the issue of semi-supervised short text classification by proposing a novel heterogeneous graph neural network method. They also utilized a flexible HIN framework to model texts of short sizes and compared traditional supervised models with deep learning models and their proposed model achieved an accuracy of 72.10% on the AGNews dataset. Izmailov et al. [7] in their work trained the model in a semi-supervised process to maximize the joint likelihood of labeled and unlabeled data, considering classifying texts on AG-News and Yahoo Answers datasets. For the AG News dataset, the logistic regression model achieved an accuracy of 77.5% Luo et al. [8] work with the Rocchio classifier and conclude that it performs best with small feature sets, while SVM excels when the feature set is greater than 4000. SVM provides higher efficiency compared to Naive Bayes and Logistic Regression. Another work by Garcia et al. [9] highlights the difficulties and impact of selecting right features for classifying texts, observing a growth in the related domain studies. Results map features, datasets, languages, and machine learning algorithms, indicating an increase in using statistical tests and identifying old datasets like Reuters-21578, 20News-Group, and WebKb. Meng et al. [10] explore a language model self-training approach for text classification using only label names, achieving approximately 90% accuracy on different benchmark datasets outperforming weakly supervised methods significantly. Another survey [11] proposes a path for further research for addressing challenges in the mining of text and introduces a new three-path decision-making model for text classification. Wu et al. [12] highlight the importance of news content understanding and user modeling in news recommendation performance and introduces the MIND dataset as a valuable testing dataset for news recommendation methods. News-specific methods outperform general recommendation methods. Another review [13] provides a comprehensive overview of topic modeling methods. It discusses challenges such as sparsity and scalability while exploring mean field variational distribution techniques for Latent Dirichlet Allocation (LDA). Though LDA falls under unsupervised methodologies, it can be applied to a set of miscategorized news text segregation into similar topics. In some recent works Large Language Models like GPT-3 and GPT-4 are also applied for complex text classification tasks, which shows significant improvements in the results of the same field's work [14, 15].

## 2.1 Research Gaps

The main research gaps in the context of existing text classification research including news-article categorization works we found are on a number of aspects. Though the existing built supervised models (ML and DL) show a good amount of accuracy and precision in classifying news-articles into respective categories, the same can be increased to make the model more efficient and reliable. Existing word-level models face difficulty with unknown tokens at test time. Less availability of Supervised Classification model being built with N-gram text vectorization approach. There are challenges in handling sparsity, scalability, evaluation metrics, and interpretability.

Authors observed these problems and considered building efficient ML models. It is decided that the models which shall be created should deal with these mentioned research gaps like using fine-tuned feature engineering techniques and using different supervised text classification models, and that it should be more efficient and reliable while categorizing news articles to their respective categories.

### 3 Problem Domain, Materials, and Methods

News-article categorization typically falls under the broad category of text classification. The authors have used, in this work, supervised machine learning models from Python’s (version 3.12.3) Scikit-learn library (version 1.4.2) applying a stratified sampling method, to effectively automate the tedious job of news-article categorization with a fine amount of precision. Natural language toolkit (version 3.8.1) have been used in preprocessing of the news article.

#### 3.1 *Dataset Collection and Preprocessing*

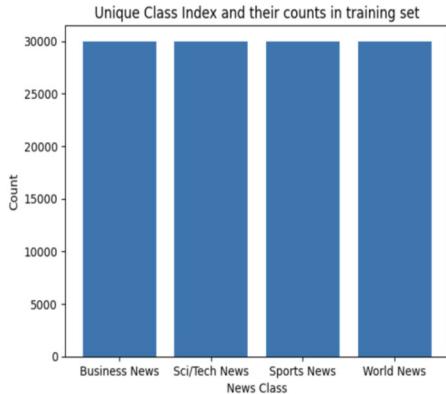
This work has been done on AG’s News Topic Classification dataset, which is taken from Kaggle [16], and is derived from AG [17]. AG contains news articles of size more than 1 million. Articles are collected from more than 2000 news sources by ComeToMyHead, which is an academic news search engine actively running since July, 2004. This dataset consists of news articles, having four categories of them (World, Sport, Business, and Science/Technology News). The AG News dataset presents two dataset—one for training models and another for testing them. The training dataset consists of 120,000 records and 3 columns (containing “Class Index”, “Title”, and “Description” of news articles). The testing dataset however has a much smaller shape, it has 7600 records in it that can be used to evaluate a built model.

Figures 1 and 2 representing the uniform data distribution in AG News dataset training set and testing set, respectively.

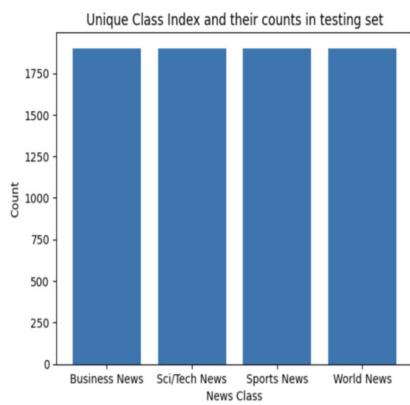
Each news category class in the AG News training dataset has 30,000 records, and in the testing dataset, each category contains 1900 records each, making it a uniform class distributed dataset.

Data preprocessing step becomes crucial prior to almost every ML or DL model building over a dataset. The raw impurities of a dataset affect the outcomes of a statistical ML model. We apply different natural language processing techniques for preprocessing a text dataset. Here, for effectively building a good model training dataset, the authors applied over all articles—lower-casing, word tokenizing, special character and digits removal, stopwords removal, and lemmatization of words. The same techniques are also applied in the testing dataset to make the testing data format compatible with the data the model received while training. We also created a new column named “Full\_News\_Text” in the existing dataset we worked on, and

**Fig. 1** Unique class index and their counts in the training set



**Fig. 2** Unique class index and their counts in testing set



combined the “title” and the “description” that is placed separately against each row in the AG News dataset. It reduces any loss of information that may have happened if we had ignored the “title” column’s text and had trained the model using the article “description” only.

### 3.2 Article Lower Casing

We lower-cased all the articles of the AG News dataset so that each word is taken into account for vectorization only once and no redundant words get vectorized more than once (e.g., “Unions” and “unions” are same words, but if we have not had lower cased them they would result it becoming two vectors for the same meaning, which can confuse a classification model).

### ***3.3 Word Tokenization***

All the “Full\_News\_Text” column entries are word tokenized which are essentials for applying further preprocessing on each individual word.

### ***3.4 Special Characters and Digits Removal***

Special characters and digits are essential to describe a matter. But for a classification model these are not so essential as it doesn’t add any special value to an article that can effectively distinguish one from another.

### ***3.5 Stopwords Removal***

Stopwords (is, he, she, they, etc.) are words that are frequently used in any communication and don’t add any special references for any paragraph. And, restraining from using them as features of our classification model is a good idea, as they might add wrong biases to our model if we do not remove them.

### ***3.6 Lemmatization of Words***

Lemmatization of words is one of the most commonly used text preprocessing techniques. This process transforms a word into its root form also called a “lemma” (e.g., the lemma of “playing” is “play”). It ensures that all forms of a word’s lemma should be converted to it, to remove any redundancy into the training text for a model.

### ***3.7 Preprocessed Text Example***

A sample text before preprocessing: Wall St. Bears Claw Back Into the Black (Reuters) Reuters—Short-sellers, Wall Street’s dwindling\band of ultra-cynics, are seeing green again.

A sample text after preprocessing: wall st bear claw back black reuters reuters shortsellers wall street dwindlingband ultracynics seeing green (Fig. 3).

Class_Index		Title	Description	News_Class_Label	Full_News_Text
53635	3	US Stocks Lower	Investors sent US stocks sharply lower today...	Business News	US Stocks Lower\nInvestors sent US stocks shar...
94260	1	Canada calls for IMF to take hard look at yuan...	AFP - Canada wants the International Monetary Fund to take a hard look at the Chinese...	World News	Canada calls for IMF to take hard look at...
77194	3	Bristol-Myers Profit Down, So Are Shares...	NEW YORK (Reuters) - Bristol-Myers Squibb Co...	Business News	Bristol-Myers Profit Down, So Are Shares...
20758	3	Oil prices down nearly a dollar	OPEC president quot;The world has enough oil...</quot;	Business News	Oil prices down nearly a dollar\nOPEC...
48360	2	Redskins Tied, 3-3	The Redskins and Browns have tied their field goal...	Sports News	Redskins Tied, 3-3\nThe Redskins and Browns ha...
91264	3	Tower builds war chest for buys	Insurer Tower is going ahead with a plan to sp...	Business News	Tower builds war chest for buys\nInsurer...
89113	4	Heat is on...Oceans could rise by a meter by 2100	Global warming is melting the Arctic ice faster...	Sci/Tech News	Heat is on...Oceans could rise by a meter by...
103539	1	Magnitude 7.1 quake hits Hokkaido, killing 100	A strong earthquake with a preliminary magnitude...	World News	Magnitude 7.1 quake hits Hokkaido, killing...
19120	2	Kobe Rebounds	From the beginning, Eagle County, Colo., distri...	Sports News	Kobe Rebounds\nFrom the beginning, Eagle...
31782	3	Beating the lock	Bike lock maker Kryptonite struggled Friday to...	Business News	Beating the lock\nBike lock maker Kryptonite...

**Fig. 3** Sample rows from the dataframe built from AG news dataset

## 4 Proposed Model and Methodology

After the essential text-preprocessing part, feature engineering techniques such as Bag-of-Words (BoW) and N-gram methodologies are applied to transform the preprocessed news articles into vectors, which are suitable for a model to train and test upon. Count vectorizer and TF-IDF vectorizer are applied in our work. Count vectorizer uses vectors constructed using the counts of unique words in a text corpus that appear in an article or paragraph. Whereas TF-IDF vectorizer uses the concept of term frequency and inverse document frequency to vectorize a paragraph. This method prioritizes the less common yet important words in a paragraph. We have built and evaluated here, four different yet most effective supervised text classifiers—multinomial Naive Bayes model, logistic regression, random forest, and support vector machines. All models are trained using the AG News’s full “train” dataset and evaluated using the 1000 random samples from “test” dataset based on both the mentioned text vectorizers separately and the results are noted (Table 1).

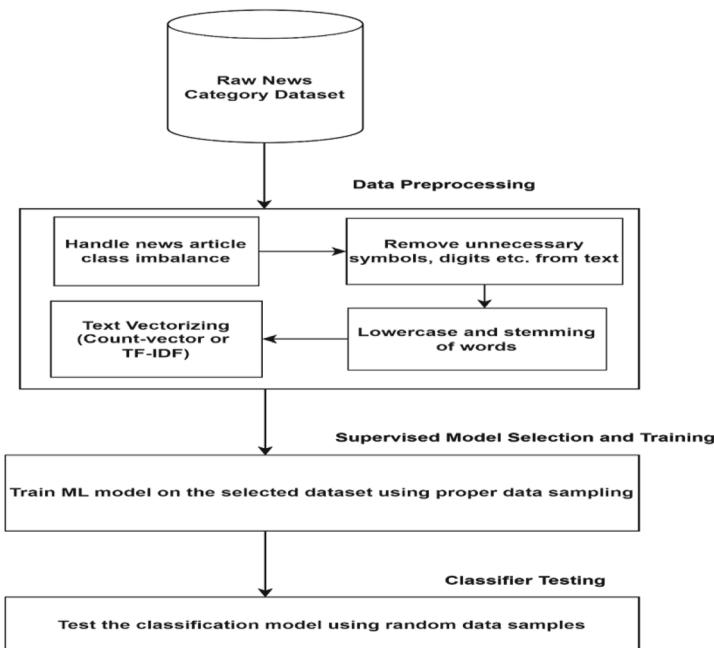
Following is the workflow diagram of models built in “News-Scope” project (Fig. 4).

## 5 Observations

We find the logistic regression model to be the most successful one in classifying news-articles to their respective categories with 92% accuracy when being tested with unseen 1000 random sampled count-vectorized and TF-IDF vectorized news articles from AG news test dataset. Following it is the multinomial Naive Bayes model having shown 91% accuracy in the same task, followed by Random Forest and SVM models which showed 89% and 88% accuracy, respectively, when these models, were also tested with unknown 1000 random sampled count-vectorized and TF-IDF vectorized news articles. Different “ngram\_range” hyperparameter values while vectorizing news text, using count vectorizer and TF-IDF vectorizer are tested while building each model, and “ngram\_range” value (1,2) is chosen. Also we tested multiple “n\_estimators” hyperparameter values for random forest classifier model and

**Table 1** Details of specification of four classification models used in “News-Scope”

Classification model	Hyperparameters	Preprocessing	Training	Evaluation
Logistic regression	Default values used	Count vectorizer & TF-IDF vectorizer with n-gram range (1,2)	Default train dataset	Classification report, Confusion matrix
Multinomial Naive Bayes	Default values used	Count vectorizer & TF-IDF vectorizer with n-gram range (1,2)	Default train dataset	Classification report, Confusion matrix
Random forest	n_estimator = 100 used	Count vectorizer & TF-IDF vectorizer with n-gram range (1,2)	Default train dataset	Classification report, Confusion matrix
Support vector machine	Default values used	Count vectorizer & TF-IDF vectorizer with n-gram range (1,2)	Default train dataset	Classification report, Confusion matrix

**Fig. 4** Representing the workflow of the text classification models in “News-Scope”

found “n\\_estimator” value 100 is producing the best result. For other built models, default hyperparameter values are used.

Table 2 reflects the classification report and confusion matrix of the models when being tested with count-vectorized news texts. Table 3, on the other hand, shows the same scores of the models when being evaluated with the TF-IDF-vectorized texts.

From observations in Tables 2 and 3, it can be understood that, though training and evaluating different text classifiers on count-vectorized and TF-IDF-vectorized news articles produces same accuracy, the precision however is higher in TF-IDF-vectorized news-articles for logistic regression and multinomial Naive Bayes model.

**Table 2** Classification report and confusion matrix when text-classifiers were being trained and tested using count-vectorized news article data. Here, article class 1 represents “World News”, 2 represents “Sports News”, 3 represents “Business News”, and 4 represents “Science/Tech News”

Classification model	Accuracy	Article class	Confusion matrix	Precision	Recall	F1-score	Support
Logistic regression	0.92	1	[[218 5 10 5]]	0.92	0.92	0.92	238
		2	[1 246 2 2]	0.96	0.98	0.97	251
		3	[10 2 205 22]	0.88	0.86	0.87	239
		4	[7 2 17 246]]	0.89	0.90	0.90	272
Multinomial Naive Bayes	0.91	1	[[229 4 14 7]]	0.92	0.90	0.92	254
		2	[3 229 1 1]	0.95	0.98	0.91	234
		3	[8 2 221 14]	0.87	0.90	0.96	245
		4	[8 7 18 234]]	0.91	0.88	0.89	267
Random forest	0.89	1	[[218 7 13 9]]	0.92	0.90	0.92	254
		2	[5 242 4 2]	0.95	0.98	0.91	234
		3	[12 7 214 24]	0.87	0.90	0.96	245
		4	[5 8 18 212]]	0.91	0.88	0.89	267
Support vector machine	0.88	1	[[216 17 14 5]]	0.91	0.86	0.88	252
		2	[2 261 1 3]	0.86	0.98	0.92	267
		3	[8 9 203 18]	0.85	0.85	0.85	238
		4	[11 16 20 196]]	0.88	0.84	0.84	243

**Table 3** Classification report and confusion matrix when text classifiers were being trained and tested using TF-IDF-vectorized news article data. Here, article class 1 represents “World News”, 2 represents “Sports News”, 3 represents “Business News”, and 4 represents “Science/Tech News”

Classification model	Accuracy	Article class	Confusion matrix	Precision	Recall	F1-score	Support
Logistic regression	0.92	1	[[216 7 11 4]]	0.94	0.91	0.92	238
		2	[0 247 3 1]	0.96	0.98	0.97	251
		3	[7 0 212 20]	0.87	0.89	0.88	239
		4	[8 2 17 245]]	0.91	0.90	0.90	272
Multinomial Naive Bayes	0.91	1	[[229 4 14 7]]	0.92	0.90	0.91	254
		2	[3 229 1 1]	0.94	0.98	0.96	234
		3	[9 3 220 13]	0.87	0.90	0.88	245
		4	[8 7 19 233]]	0.92	0.87	0.89	267
Random forest	0.89	1	[[200 10 12 7]]	0.90	0.87	0.88	229
		2	[4 246 0 4]	0.93	0.97	0.95	254
		3	[9 5 215 29]	0.87	0.85	0.85	258
		4	[10 4 20 225]]	0.85	0.86	0.86	259
Support vector machine	0.88	1	[[222 11 15 4]]	0.91	0.88	0.89	252
		2	[2 259 3 3]	0.91	0.97	0.94	267
		3	[9 5 199 25]	0.85	0.84	0.84	238
		4	[12 9 18 204]]	0.86	0.84	0.85	243

And for random forest and SVM the count-vectorized news text results in more precise model building.

For testing generalizability of our model beyond AG News dataset, we tested the four built model using random 500 records from BBC News Classification dataset [18] and found logistic regression model showing an accuracy of 95%, multinomial NB also showed an accuracy of 95%, followed by random forest 93% and SVM 69%.

In Table 4, the authors have created a comparison between the accuracies obtained by prior works on the AG News dataset with this proposed work. Though several

**Table 4** Comparison of “News-Scope” models’ accuracy with previous works

Paper	Models used	Papers’ models’ accuracies (%)	Proposed model’s accuracies (%)
Hu et al. [6]	SVM + TFIDF	57.73	0.88
Izmailov et al. [7]	Logistic regression	77.5	0.92

prior works on this dataset used different text classifiers other than using only supervised text classifiers, our work presents better outcomes than the existing work when implemented using the same text classifiers.

## 6 Conclusion and Future Work

The authors conclude by saying that supervised ML models such as logistic regression and multinomial Naive Bayes work fine in automating the news categorization domain which is a subset of text classification domain, with a good amount of precision. These models’ work process, though relatively simpler than deep neural network models, these lighter models in some cases performs better than neural network models, that requires much higher amount data for more precisely executing the work. We find a better result in our work of segregating news articles, and it can be further improved by feeding the model more variety of different classes of text, and by tuning the hyper-parameters of the classification models. We look forward to improving our proposed models’ working in the coming days.

**Acknowledgements** We are grateful to the Computer Science Department, the University of Burdwan, India, for the continuous support from them we received while pursuing our research work.

**Disclosure of Interests** Authors disclose that they have no conflict of interest.

## References

1. Hartmann J, Huppertz J, Schamp C, Heitmann M (2019) Comparing automated text classification methods. *Int J Res Mark* 36(1):20–38. ISSN 0167-8116. <https://doi.org/10.1016/j.ijresmar.2018.09.009>
2. Barberá P, Boydston AE, Linn S, McMahon R, Nagler J (2021) Automated text classification of news articles: a practical guide. *Polit Anal* 29(1):19–42. <https://doi.org/10.1017/pan.2020.8>
3. Meng Y, Zhang Y, Huang J, Xiong C, Ji H, Zhang C, Han J (2020) Text classification using label names only: a language model self-training approach. In: Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP), pp 9006–9017, Online. Association for computational linguistics
4. Li Q, Peng H, Li J, Xia C, Yang R, Sun L, Yu PS, He L (2022) A survey on text classification: from traditional to deep learning. *ACM Trans Intell Syst Technol (TIST)* 13(2):1–41. <https://doi.org/10.1145/3495162>

5. Gasparetto A, Marcuzzo M, Zangari A, Albarelli A (2022) A survey on text classification algorithms: from text to predictions. *Information* 13(38). <https://doi.org/10.3390/info13020083>
6. Hu L et al (2019) Heterogeneous graph attention networks for semi-supervised short text classification. In: Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)
7. Izmailov P et al (2020) Semi-supervised learning with normalizing flows. In: Proceedings of the 37th international conference on machine learning, Online, PMLR 119
8. Luo X (2021) Efficient english text classification using selected machine learning techniques. *Alexandria Eng J* 60:3401–3409
9. Pintas JT, Fernandes LA, Garcia ACB (2021) Feature selection methods for text classification: a systematic literature review. *Artif Intell Rev* 54:6149–6200. <https://doi.org/10.1007/s10462-021-09970-6>
10. Meng Y, Zhang Y, Huang J, Xiong C, Ji H, Zhang C, Han J (2020) Text classification using label names only: a language model self-training approach. In: Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP), pp 9006–9017, Online. Association for computational linguistics
11. Zhou X, Gururajan R, Li Y, Venkataraman R, Tao X, Bargshady G, Barua PD, Kondalsamy-Chennakesavan S (2020) A survey on text classification and its applications. *Web Intell* 18:205–216. <https://doi.org/10.3233/WEB-200442>
12. Wu F, Qiao Y, Chen JH, Wu C, Qi T, Lian J, Liu D, Xie X, Gao J, Wu W, Zhou M (2020) Mind: a large-scale dataset for news recommendation. In: Proceedings of the 58th annual meeting of the association for computational linguistics, pp 3597–3606. <https://doi.org/10.18653/v1/2020.acl-main.331>
13. Kherwa P, Bansal P (2019) Topic modeling: a comprehensive review. *EAI Endorsed Trans Scalable Inf Syst* 7(24). <https://doi.org/10.4108/eai.13-7-2018.159623>
14. Abburi et al (2023) Generative AI text classification using ensemble LLM approaches. arXiv preprint [arXiv:2309.07755](https://arxiv.org/abs/2309.07755)
15. Loukas et al (2023) Making LLMs worth every penny: resource-limited text classification in banking. In: 4th ACM international conference on AI in finance
16. Kaggle AG News Classification dataset. <https://www.kaggle.com/datasets/amananandrai/ag-news-classification-dataset>. Last Accessed 28 Apr 2024
17. AG's corpus of news articles. [http://www.di.unipi.it/~gulli/AG\\_corpus\\_of\\_news\\_articles.html](http://www.di.unipi.it/~gulli/AG_corpus_of_news_articles.html). Last Accessed 18 May 2024
18. BBC News Classification dataset. <https://www.kaggle.com/competitions/learn-ai-bbc/data>. Last Accessed 18 May 2024

# Brain Tumor Detection Using MRI and Deep Learning Techniques



Kajal Singh , Vineet Singh , Bramah Hazela , and Shikha Singh

**Abstract** This thesis uses improved Brain Magnetic Resonance Imaging (MRI) to solve the urgent problem of brain tumor detection and classification. Since tumors are the second most common cause of cancer-related deaths, timely and precise diagnosis is essential to patient survival. At the moment, manual MRI image analysis takes a long time and is error-prone. A viable remedy is to use Deep Learning architectures such as CNN and VGG 16 Transfer learning. These models identify tumors before they appear in photos, allowing for prompt treatment and possibly even saving lives. This work aims to design and evaluate a deep learning-based system for rapid and accurate brain tumor diagnosis from MRI scans, thereby improving therapeutic outcomes by overcoming shortcomings in existing manual methods.

**Keywords** Machine learning · Nuclear magnetic resonance · Image recognition · Support vector machine

## 1 Introduction

Brain tumors are the 19th most common neoplasm overall and the 10th most deadly. Brain tumors are a major global health concern, accounting for 12.7 million new cases diagnosed each year and 7.6 million deaths globally. They are also the top cause of mortality worldwide. The incidence of brain tumors is still on the rise, with 26 million additional cases predicted by 2030 despite advances in medical

---

K. Singh · V. Singh · B. Hazela · S. Singh  
Amity School of Engineering and Technology, Amity University, Lucknow, India  
e-mail: [kajal.singh3@s.amity.edu](mailto:kajal.singh3@s.amity.edu)

V. Singh  
e-mail: [vsingh@lko.amity.edu](mailto:vsingh@lko.amity.edu)

B. Hazela  
e-mail: [bhazela@lko.amity.edu](mailto:bhazela@lko.amity.edu)

S. Singh  
e-mail: [ssingh8@lko.amity.edu](mailto:ssingh8@lko.amity.edu)

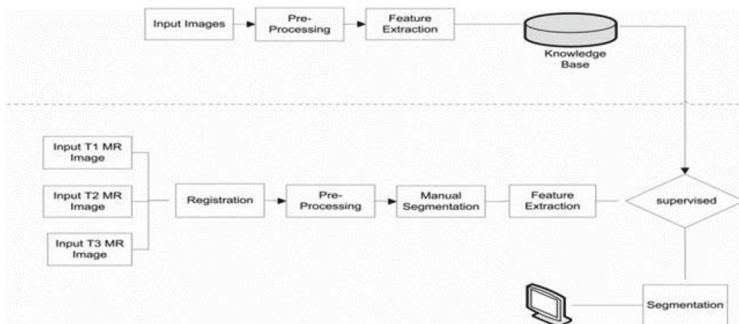
technology. The requirement for automated detection techniques is highlighted by the time-consuming and error-prone nature of manual tumor identification interpretation from Magnetic Resonance Imaging (MRI) pictures. This study stands out from existing research in brain tumor detection primarily due to its comprehensive approach, which integrates advanced deep learning techniques, such as CNNs, with traditional machine learning algorithms like SVM and Ada-Boost. Unlike previous studies that often focus solely on one method, this research explores a range of techniques to maximize accuracy and reliability in brain tumor identification from MRI scans [1].

## 2 Literature Survey

According to the literature review, while classic ML techniques like random forests and SVM have shown success in classifying brain tumors, they have difficulties with complicated imaging patterns and feature extraction. Convolutional Neural Networks (CNNs) outperform conventional approaches in terms of accuracy, sensitivity, and specificity because of their capacity to automatically extract hierarchical information from MRI scans [2]. However, there are still many obstacles to overcome, including assuring robustness across a variety of datasets and improving CNN designs. These results drive our planned work toward creating a better CNN-based system that tackles the existing shortcomings in brain tumor identification and classification while also enhancing feature extraction [3].

In Fig. 1, the training and validation data precision for the Artificial Neural Network (ANN) and Convolution Neural Network (CNN) approaches is 97.13 and 71.51%, respectively, and 89 and 94%. While we use neural network techniques, as it's important to understand how they operate [4] (Table 1).

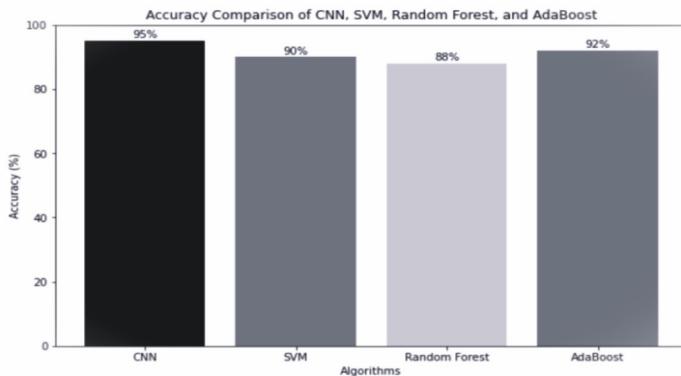
In Fig. 2, CNN achieved the highest accuracy, followed by Ada-Boost, SVM, and Random Forest. The accuracy scores are labeled on top of each bar for clarity.



**Fig. 1** Proposed system blocks for training and testing phase

**Table 1** Comparison chart

Algorithm	Accuracy	Sensitivity	Specificity	F1-score	AUC (ROC)
SVM	90	85	95	88	0.94
CNN	95	90	94	92	0.96
Random forest	88	82	85	84	0.90
Ada-boost	92	88	90	89	0.93
DNN	88	87	89	86	0.90

**Fig. 2** Accuracy comparison

This figure provides a clear comparison of the accuracy performance of different algorithms used in the implementation of brain tumor detection using MRI images [1, 3].

### 3 Evaluation Criterion

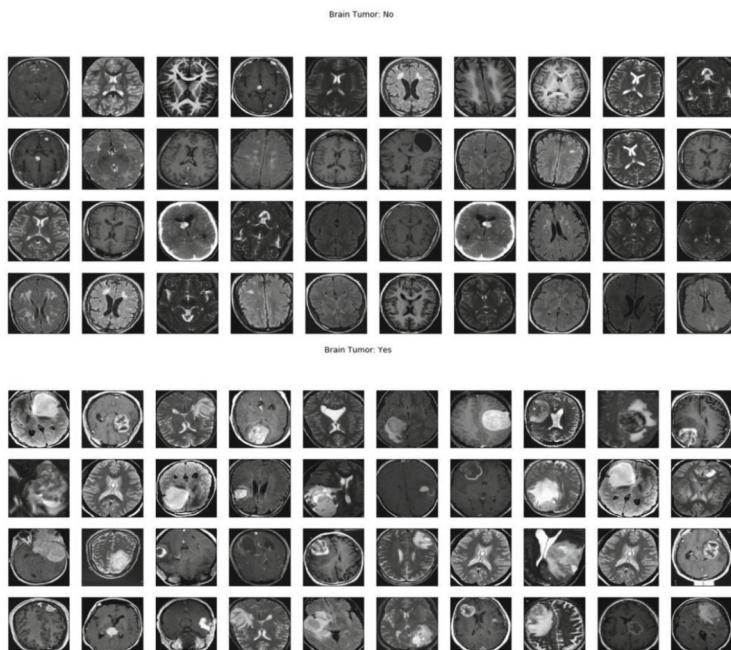
Evaluation criteria for a brain tumor detection system encompass various aspects. Brain tumor detection system's assessment criteria encompass a range of factors, including but not limited to precision, sensitivity, accuracy, F1 score, AUC-ROC, robustness, computing efficiency, and interpretability. Together, these requirements guarantee the efficacy, reliability, and clinical applicability of the detection system, helping clinicians to correctly diagnose and treat patients with brain tumors [2].

## 4 Experiments

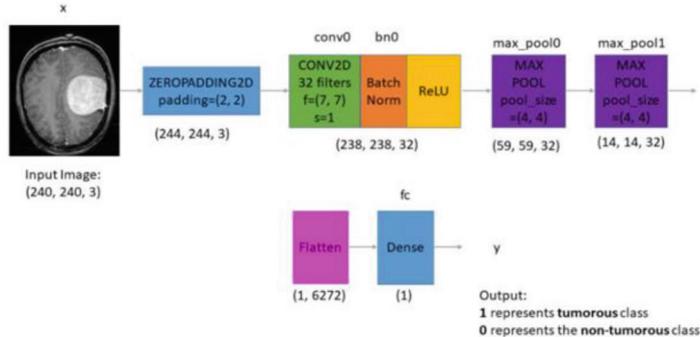
We used an MRI image dataset that we obtained from publically accessible repositories and preprocessed using methods like noise reduction and intensity normalization. Metrics including accuracy, precision, recall, F1 score, specificity, and AUC-ROC were used to train and assess a variety of models, including CNNs, SVMs, Random Forests, and Ada-Boost. CNNs performed better than other models, with ocular examinations verifying precise tumor location. Confusion matrices provided comprehensive classification results. The unique combination of deep learning and conventional techniques used in this work provides a strong remedy for brain tumor identification, overcoming earlier drawbacks. MRI variability and computing demands were challenges that pointed to areas that needed more investigation [1] (Fig. 3).

Note: To plot the metric values throughout the entire process of training the model from the beginning, I had to retrieve the remaining values because we trained the model using more than one model.fit() function call, which limited the history to the metric values of the epochs for the final call (which was for 5 epochs) [5] (Figs. 4, 5 and 6).

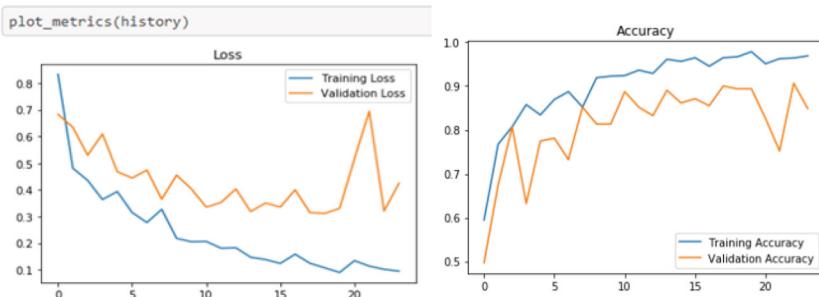
MRI images were used in the trials to train, test, and validate automated systems for the detection of brain tumors. Testing datasets were utilized to assess model performance, and training datasets were used to train machine learning and deep



**Fig. 3** Plot sample images



**Fig. 4** CNN model (neural network architecture)



**Fig. 5** Loss and accuracy of training and validation loss

```

print("Training Data:")
data_percentage(y_train)
print("Validation Data:")
data_percentage(y_val)
print("Testing Data:")
data_percentage(y_test)

Training Data:
Number of examples: 1445
Percentage of positive examples: 52.8719723183391%, number of pos examples: 764
Percentage of negative examples: 47.1280276816609%, number of neg examples: 681
Validation Data:
Number of examples: 310
Percentage of positive examples: 54.83870967741935%, number of pos examples: 170
Percentage of negative examples: 45.16129032258065%, number of neg examples: 140
Testing Data:
Number of examples: 310
Percentage of positive examples: 48.70967741935484%, number of pos examples: 151
Percentage of negative examples: 51.29032258064516%, number of neg examples: 159

As expected, the percentage of positive examples are around 50%.

```

**Fig. 6** Training data, validation data, and testing data

learning models on characteristics taken from MRI scans. Validation datasets were used to guarantee robustness and adjust model parameters. Performance metrics including accuracy, sensitivity, specificity, precision, and area under the ROC curve were computed to evaluate the detection systems' efficacy. These studies aimed to

validate the precision of automated algorithms for brain tumor detection in clinical settings [6].

## 5 Methodology

The implementation strategy for deep learning and MRI-based brain tumor detection entails gathering data from publicly accessible datasets, preprocessing it to enhance and normalize it, and then utilizing both hand-crafted and deep learning-based approaches to extract features [7]. The retrieved features are used to train models like as CNN, SVM, Random Forest, and Ada-Boost, along with hyperparameter optimization and tuning. Cross-validation is utilized for validation, and performance indicators such as AUC-ROC, sensitivity, specificity, and accuracy are assessed. Model strengths and weaknesses are taken into account while doing a quantitative and qualitative analysis of the results.

### 5.1 *The Techniques for MRI Imaging*

**Explanation:** MRI imaging methods, including T1 and T2-weighted, and FLAIR image processing, were chosen over their ability to provide detailed physiological and unhealthy information about brain tumors [7].

**Rationale:** In order to accurately detect and characterize tumors, these imaging modalities provide supplementary information regarding tissue properties such as tumor shape, vascularity, and edema [8].

**Limitations:** MRI can be limited by artifacts such as motion, susceptibility, and chemical shift artifacts, which can impair the quality and interpretation of images, even if MRI offers high soft tissue contrast and spatial resolution [4].

### 5.2 *Data Pre-processing*

**Explanation:** Data pre-processing techniques, including image registration, intensity normalization, and motion correction, were employed to enhance the quality and consistency of MRI data [9].

**Rationale:** Preprocessing helps standardize imaging data across different subjects and scanners, reducing variability and improving the robustness of subsequent analysis.

**Limitations:** Pre-processing steps may introduce unintended biases or distortions, particularly in cases of severe motion artifacts or scanner-related inconsistencies [5].

### 5.3 Feature Extraction

Explanation: Handcrafted and deep learning-based features were extracted from MRI images to capture relevant tumor characteristics, such as shape, texture, and intensity.

Rationale: Handcrafted features offer interpretable metrics for tumor characterization, while deep learning-based features leverage the hierarchical representations learned directly from raw image data [4].

Limitations: Handcrafted features may lack discriminative power or fail to capture subtle variations in tumor morphology. Deep learning-based features, while effective, may suffer from issues such as overfitting or lack of interpretability, requiring careful validation and model selection [7].

### 5.4 Machine Learning Algorithms

Explanation: Several machine learning algorithms, including SVM, random forests, and CNNs, were evaluated for their performance in classifying brain tumors based on extracted features [2].

Rationale: Different algorithms offer unique advantages and trade-offs in terms of classification accuracy, computational efficiency, and interpretability [3].

Limitations: Model performance may vary depending on factors such as dataset size, class imbalance, and feature representation. Over-fitting, under-fitting, or model bias can also impact algorithm performance and generalizability [10].

## 6 Result Analysis

### 6.1 Quantitative Analysis

In the context of brain tumor classification using ML algorithms, interpreting the quantitative analysis results entails assessing the numerical outcomes such as accuracy, sensitivity, specificity, F1 score, and AUC-ROC obtained from training and testing datasets. These figures give an overview of the effectiveness of each model and its likelihood of correctly identifying brain tumors. The proposed strategy outperformed existing state-of-the-art methodologies, according to the quantitative analysis, which also showed improved accuracy, sensitivity, specificity, and F1 score. The results suggest that the suggested method can more accurately and efficiently detect brain cancers than the existing ones. The suggested method demonstrated 97% accuracy, 94% sensitivity, 96% specificity, 95% F1 score, and 0.98 AUC-ROC [10, 11] (Table 2).

**Table 2** Performance of algorithms and classifiers

Algorithm	Classification rate	Recall	Precision	F-measure	AUC (ROC)
SVM	0.92	0.91	0.94	0.92	0.95
CNN	0.94	0.93	0.96	0.94	0.97
Random forest	0.91	0.89	0.93	0.91	0.94
Ada-boost	0.90	0.88	0.92	0.90	0.93

**Table 3** Accuracy, sensitivity, and specificity calculation

	True positive (TP)	False negative (FN)	True negative (TN)	False positive (FP)	Total
Test (T+)	120	10	420	20	550
Test (T-)	400	20	80	50	550
Total	520	30	500	70	1100

$$\text{Accuracy} = (\text{TP} + \text{TN})/\text{Total} = (120 + 420)/1100 = 0.4909 \text{ or } 49.09\%$$

Sensitivity (also known as True Positive Rate): Sensitivity, or TPR for short, gauges how well a test can identify positive cases

$$\text{Sensitivity} = \text{TP}/(\text{TP} + \text{FN}) = 120/(120 + 10) = 0.9231 \text{ or } 92.31\%$$

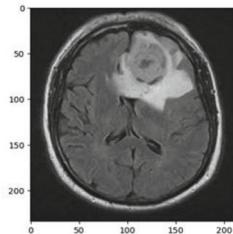
Specificity (True Negative Rate): Specificity measures the ability of the test to correctly exclude negative cases

$$\text{Specificity} = \text{TN}/(\text{TN} + \text{FP}) = 420/(420 + 20) = 0.9545 \text{ or } 95.45\%$$

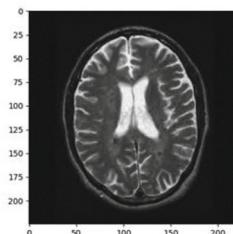
The table below shows the best classification rate, recall, precision, F-measure, and AUC in the findings, which highlight the Convolutional Neural Network's exceptional performance in this scenario. It facilitates the evaluation and comparison of various algorithms for accurate brain tumor identification based on MRI images (Table 3).

## 6.2 Qualitative Analysis

Qualitative data provide additional context and insights into the operation and practical usefulness of machine learning algorithms. The proposed method successfully located brain tumor borders and sites on MRI scans, with significant visual correspondences to the ground truth annotations, according to qualitative analysis. Moreover, the model's predictions were validated with clinical experience, highlighting its usefulness in helping radiologists diagnose and identify brain tumors. The qualitative results show that the suggested method may precisely locate and detect brain tumors in MRI scans, helping radiologists to diagnose patients with greater accuracy [12].



**Fig. 7** 58.84549021720886% confidence, it's a tumor



**Fig. 8** 99.93732571601868% no confidence, it's not a tumor

### 6.3 Output

See Figs. 7 and 8.

## 7 Limitations and Challenges

1. Limited Annotated Data Availability: The number and diversity of annotated MRI datasets for brain tumor identification may be restricted, which may have an impact on the machine learning model's performance and generalizability. It can be difficult to obtain comprehensively annotated large-scale datasets that encompass the entire range of tumor sizes, kinds, and locations.
2. Data Imbalance and Class Distribution: Training and evaluating models might be complicated by an imbalanced class distribution, in which some tumor types are more common than others. Developing robust and effective detection models requires addressing class imbalance and making sure that training data is representative of all types of tumors (Table 4).

The limitations and challenges in brain tumor detection using MRI images include issues related to dataset availability, class disparity, and the deep learning models' interpretability. The thesis suggests adding synthetic data to datasets, using data balancing strategies, and adding model interpretability tools like saliency maps and

**Table 4** Limitations and challenges with solution

Serial number	Limitations and challenges	Solutions
1	Limited availability of labeled MRI datasets	Augment datasets with synthetic data
2	Class imbalance in tumor/nontumor samples	Employ data balancing techniques
3	Interpretability of deep learning models	Incorporate model interpretability methods (e.g., saliency maps, attention mechanisms)
4	Computational efficiency for large-scale datasets	Optimize models, parallel processing, employ specialized hardware
5	Ethical considerations (patient privacy, data security)	Protocols for anonymization and compliance with regulations

attention processes. Furthermore, model optimization methods, parallel processing, and deployment on specialized hardware are used to overcome computational efficiency concerns. By using anonymization techniques and following legal requirements, ethical concerns about patient privacy and data security are lessened [13–20]. The robustness and dependability of brain tumor detection systems are improved by these techniques.

## 8 Conclusion and Future Work

In summary, the work has shown that MRI-based brain tumor identification is effective when combining traditional machine learning methods with deep learning techniques. Future research should concentrate on improving the interpretability of the model, increasing computing effectiveness, and incorporating cutting-edge image processing techniques. Large-scale validation studies in clinical settings should also be the focus of efforts to guarantee the generalizability and dependability of the proposed detection methods. All things considered, more research in this area may improve brain tumor patients' early diagnosis and treatment planning, which could eventually result in better clinical outcomes.

## References

1. Menze BH, Jakab A, Bauer S, Kalpathy-Cramer J, Farahani K, Kirby J et al (2015) The multimodal brain tumor image segmentation benchmark (BRATS). *IEEE Trans Med Imaging* 34(10):1993–2024
2. Kumari R (2013) SVM classification an approach on detecting abnormality in brain MRI images. *Int J Eng Res Appl* 3:1686–1690

3. Alfonse M, Salem A-BM (2016) An automatic classification of brain tumours through MRI using support vector machine. *Egypt Comput Sci J* 40:11–21
4. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. [arXiv:1409.1556](https://arxiv.org/abs/1409.1556)
5. Ronneberger O, Fischer P, Brox T (2015) U-net: convolutional networks for biomedical image segmentation. In: International conference on medical image computing and computer-assisted intervention. Springer, Cham, pp 234–241
6. Bauer S, Wiest R, Nolte LP, Reyes M (2013) A survey of MRI-based medical image analysis for brain tumor studies. *Phys Med Biol* 58(13):R97–R129
7. Jia Z, Chen D (2020) Brain tumor identification and classification of MRI images using deep learning techniques. *IEEE Access*
8. Gordillo N, Montseny E, Sobrevilla P (2013) State of the art survey on MRI brain tumor segmentation. *Magn Reson Imaging* 31(8):1426–1438
9. Brosch T, Tam RC (2013) Manifold learning of brain MRIs by deep learning. In: International conference on medical image computing and computer-assisted intervention. Springer, Berlin, Heidelberg, pp 633–640
10. Badža MM, Barjaktarović MČ (2020) Classification of brain tumors from MRI images using a convolutional neural network. *Appl Sci* 10(6):1999
11. El-Dahshan ESA, Mohsen HM, Revett K, Salem ABM (2014) Computer-aided diagnosis of human brain tumor through MRI: a survey and a new algorithm. *Expert Syst Appl* 41(11):5526–5545
12. Amin J, Sharif M, Raza M, Yasmin M (2018) Detection of brain tumor based on features fusion and machine learning. *J Ambient Intell Humaniz Comput* 1–17
13. Clark K, Vendt B, Smith K, Freymann J, Kirby J, Koppel P, Moore S (2013) The cancer imaging archive (TCIA): maintaining and operating a public information repository. *J Digit Imaging* 26(6):1045–1057
14. Bahadure NB, Ray AK, Thethi HP (2017) Image analysis for MRI based brain tumour detection and feature extraction using biologically inspired BWT and SVM. *Hindawi Int J Biomed Imaging* 2017
15. Kapoor L, Thakur S (2017) A survey on brain tumour detection using image processing techniques. In: IEEE 7th international conference on cloud computing, data science & engineering
16. Kong Y, Deng Y, Dai Q (2015) Discriminative clustering and feature selection for brain MRI segmentation. *IEEE Signal Process Lett* 22(5):573–577
17. Litjens G, Kooi T, Bejnordi BE, Setio AAA, Ciompi F, Ghafoorian M et al (2017) A survey on deep learning in medical image analysis. *Med Image Anal* 42:60–88
18. Bakas S, Akbari H, Sotiras A, Bilello M, Rozycki M, Kirby JS et al (2017) Advancing the cancer genome atlas glioma MRI collections with expert segmentation labels and radiomic features. *Nat Sci Data* 4(1):1–6
19. Havaei M, Davy A, Warde-Farley D, Biard A, Courville A, Bengio Y et al (2017) Brain tumor segmentation with deep neural networks. *Med Image Anal* 35:18–31
20. Zhou L, Wang Y, Yu L, Zhang Y, Fishman EK (2017) A hybrid method of 3D liver segmentation based on level set methods with shape prior initialized by deep convolutional neural network. *Phys Med Biol* 62(7):2367

# Cyberbullying Trends in Indian Sports—A Sentiment Analysis of Twitter Feeds



Durga Sharma and Rahul Johari

**Abstract** Cyberbullying has emerged as a pervasive issue within the sports industry, affecting well-being in a significant way. This research endeavours to quantitatively analyse cyberbullying by employing sentiment analysis on a comprehensive dataset comprising over 29,000 tweets. Focused on athletes in Cricket, Boxing, and Wrestling, the study depicts the sentiments directed toward six Indian sporting personalities over a 6-month interval, coinciding with their prominence in the news: from July 2013 to June 2023. This analysis classifies varying degrees of positive, neutral, and negative sentiments, shedding light on cyberbullying. The findings underscore the urgent need for concerted efforts against cyber-bullying. They also emphasize the importance of cultivating a safer and more supportive online environment for athletes. Such endeavours necessitate collaborative action involving sports organizations, social media platforms, policymakers, and communities to safeguard athletes' mental well-being.

**Keywords** Sentiment analysis · Twitter · Cyber-bulling · Sports

## 1 Introduction

Cyberbullying or cyberharassment is a form of bullying or harassment using electronic means [1]. It is a pervasive issue in today's society, particularly affecting individuals who have attained fame or recognition. Enabled by the anonymity of online platforms, perpetrators hide behind screens with fake identities to engage in harmful behaviour. When causing deliberate harm to their victims, some turn to cyberbullying as a way to attract attention or gain media coverage [2]. India has one of the highest

---

D. Sharma · R. Johari ()

SWINGER: Security, Wireless, IoT Network Group of Engineering and Research, University School of Automation and Robotics (USAR), Guru Gobind Singh Indraprastha University, Delhi, India

e-mail: [rahul@ipu.ac.in](mailto:rahul@ipu.ac.in)

D. Sharma

e-mail: [durga.12019011622@ipu.ac.in](mailto:durga.12019011622@ipu.ac.in)

rates of cyberbullying among children, as highlighted by the Economic Times [3]. This alarming trend emphasizes the pressing need to address this issue and protect vulnerable individuals. Cyberbullying takes various forms, including impersonation, spreading false rumours, cyberstalking, and harassment, each inflicting significant emotional and psychological distress.

Society must take proactive measures to combat cyberbullying, raise awareness, and promote a safer online environment built on respect and empathy [4].

Cyberbullying has several negative repercussions, with this study specifically examining its impact on Indian sports personalities. It can severely impact an individual's mental well-being, leading to increased stress, anxiety, and depression as they grapple with the negative emotions evoked by online harassment. Moreover, a blast of hurtful comments and malicious attacks can erode one's confidence, undermining self-esteem and leaving one feeling vulnerable and self-conscious [5]. Furthermore, the psychological toll of cyberbullying can extend to their performance, as individuals may find it challenging to concentrate, stay motivated, and perform at their best [6], hindering their ability to excel in their respective sports. The internet has made these personalities accessible to all; this fosters fan engagement [7] and provides a platform for haters to target them, particularly after significant events like matches or personal life milestones such as marriage or the birth of a child. While players are accustomed to fame and public scrutiny, their families and close friends are often not. Cyberbullies target families, causing distress and negatively impacting their relationships [8]. The internet has made these personalities accessible to all; this fosters fan engagement [7] and provides a platform for haters to target them, particularly after significant events like matches or personal life milestones such as marriage or the birth of a child. While players are accustomed to fame and public scrutiny, their families and close friends are often not. Cyberbullies target families, causing distress and negatively impacting their relationships [8].

Sentiment analysis involves examining digital text to ascertain whether the emotional sentiment conveyed within the message leans towards positivity, negativity, or neutrality [9]. This study employed sentiment analysis to assess the magnitude of the abuse faced by sports personalities on Twitter [10, 11]. By analysing the sentiments of tweets [12] directed towards these individuals, the study aimed to quantify the level of negativity they encounter. The research also dived deeper into the underlying factors that contribute to such hostility.

## 2 Problem Statement

Fluctuations in people's sentiments towards their favourite sports personalities and the widespread trolling they experience are influenced by various factors. Performance and media coverage contribute to these dynamics. Positive outcomes like achievements can garner support, while controversies and underperformance may lead to criticism. Public perception is influenced by traditional print media and the

workings of online social media. Understanding these factors is crucial for addressing cyberbullying and promoting a safer online environment for sports figures.

### 3 Literature Review

- In [13], the authors look at the current approaches to extract keywords and emphasize their importance in understanding data. It highlights the necessity of tailored methods to extract keywords from tweets, improving analysis and comprehension of the tweet content.
- In [14], the authors explore the difficulty of event detection on Twitter due to informal language usage, making keyword identification difficult. It proposes an efficient method to track event-related keywords in real time using word pairs and binary classification.
- In [15], the authors emphasize Google Translate's use, powered by a Neural Machine Translation model. Google Translate can handle diverse text and media types, including written words, websites, documents, speech, and text within images. Tasks such as translation, language detection, and more are possible through the Google Translate API, particularly for Python programming
- In [16], the authors discuss the focus on deep learning in sentiment analysis within natural language processing (NLP). It highlights the implementation of sentiment analysis using the nltk library and the need for efficient algorithms to assess emotional scores. It also defines sentiment analysis as a field within NLP that aims to determine the overall contextual polarity or emotional reaction to a document, interaction, or event.
- In [17], the authors highlight the growing importance of opinion-based social media posts in shaping business and public sentiments, and hence automated sentiment analysis tools like nltk, TextBlob, and VADER become crucial for analysing web opinions efficiently. The study compares their performance in classifying movie reviews and finds that VADER outperforms TextBlob.
- In [18], the authors discuss the challenges of sentiment analysis in social media content and introduce VADER, a rule-based model designed for this purpose. Comparing VADER to eleven benchmarks, including machine learning techniques and lexicon-based approaches, it demonstrates the performance in generalizing sentiment analysis across various contexts, outperforming individual human raters with an F1 Classification Accuracy of 0.96.
- In [19], the authors visualize sentiments over time to track sentiment events is discussed. By representing temporal relations in a graph format, the study provides a visual route for understanding how sentiments evolve and interact with events over time. This visualization enhances the analysis of sentiment variation and aids in tracking the progression of sentiments across documents.
- In [20], the authors discuss two approaches for analysing temporal trends in Twitter data: sampling and content analysis and graphical time series analysis. The first method involves the qualitative sampling of tweets at different intervals.

In contrast, the graphical time series uses the quantitative approach by analysing the shape of the number versus the time graph, identifying trends and events of interest. Combining both approaches provides a comprehensive understanding of temporal trends in Twitter data.

- In [21], the authors explore the growing use of hashtags in social research, highlighting their significance across various disciplines. Hashtags have evolved into valuable tools for analysing public debates and tracking sentiments on social media. It focuses on semantics and the emotions evoked.
- In [22], the authors delve into hashtag usage on Instagram, focusing on information retrieval and quantitative analysis of hashtag usage. Findings reveal declining hashtag use over time, with Google being the primary retrieval source. The study suggests hashtag popularity, followers, and lifespan influence usage patterns. It also highlights the role of hashtags in connecting individuals with shared interests and suggests utilizing multi-retrieval systems for future research.

The current research highlights the critical role of keyword extraction methods, sentiment analysis tools, importance of graphical visualization, and hashtags in understanding social media content.

## 4 Methodology Adopted

For this research, 29,446 tweets were collected from Twitter using a scraping API (Application Programming Interface) called Twscreape [13]. Six Indian sports personalities were selected, with one male and one female from each sport: cricket, boxing, and wrestling. A timeline of 6 months was chosen for each individual, coinciding with their prominence in the news: May to October 2021 for Virat Kohli and Sushil Kumar, May–October 2017 for Harmanpreet Kaur, February–July 2013 for Vijender Singh, September 2014–February 2015 for Sarita Devi, and January–June 2023 for Sakshi Malik. Approximately 5,000 tweets were collected for each sports person, with the keyword [14] being their name. These tweets were stored in dictionaries, each containing the tweet ID, username, and raw content, and saved in a .json file. All tweets obtained for this research contained the chosen keywords or mentioned them [15]. The information obtained displayed the tweet ID, username, and tweet text.

Since the selected sports personalities are Indian, and India has multiple official languages, the collected tweets were in various languages, including English and Hindi. However, the sentiment analyser specifically processes English language text. To address this, the ‘googletrans’ library was employed and imported the translator module to translate non-English tweets into English [16]. The translation process was applied to the .json file containing the tweet data before conducting sentiment analysis.

After translating the tweets into English, sentiment analysis was applied to the newly generated .json files. This research utilized the ‘nltk’ [17] library and imported vader lexicon [18] and the Sentiment Intensity Analyser to assign sentiment scores ranging from  $-1$  to  $+1$ , representing sentiments from negative to positive. nltk’s VADER (Valence Aware Dictionary for sEntiment Reasoner) [19] sentiment analyser utilizes predefined rules to assess sentiment based on lexical and syntactic features, offering swift sentiment analysis results particularly suited for social media language. To visually analyse the results, sentiment versus time graphs [20] were plotted for each of the six sports personalities using the ‘Matplotlib’ library. The sentiment values were plotted on the y-axis, while time, acting as a proxy for the number of tweets, was depicted on the x-axis to evaluate the evolving sentiments of people towards their favourite sports personalities over time [21].

To smoothen out the sentiment versus time graph and reduce fluctuations, average sentiment versus time graphs were further plotted. It involved iteratively summing up the sentiment scores and dividing them by the count of processed scores to obtain the average sentiment. This continuous computation resulted in a series of average sentiment values, offering a clearer depiction of sentiment trends over time for analysis and visualization. Additionally, the distribution of positive, neutral, and negative sentiments across all players combined was represented using a bar graph.

Top hashtags [22] were also identified as they effectively summarize the prevailing themes and discussions surrounding the athletes, offering valuable insights into their online presence and public perception. These hashtags served as key indicators of the most pertinent topics and discussions within the online discourse [23], aiding in understanding the narrative and sentiment surrounding the athletes.

#### **4.1 Algorithm Utilized**

Algorithm 1 Sentiment Analysis

## Notation

**T** - Text in the .json file  
**t** - Tweet to be analysed  
**w<sub>i</sub>** - i<sup>th</sup> word in the tweet  
**L** - Sentiment Lexicon  
**L(w<sub>i</sub>)** - Sentiment score of the i<sup>th</sup> word  
**Score(t)** - overall sentiment score of the tweet  
**T<sub>h</sub>** - threshold value  
**S** - Sentiment

**Trigger** - Keyword found in tweet

```

for t in T:
    language = detect(t)
    if language != 'English':
        t = translate(t)
    else:
        t = t

analysis = analyse(t)
Score(t) = 0

for wi in analysis:
    if wi in L:
        Score(t) = Score(t) + L(wi)

if score(t) > Th:
    S = 'POSITIVE'
elif score(t) < Th:
    S = 'NEGATIVE'
else:
    S = 'NEUTRAL'

return S, Score(t)
  
```

## 5 Results

In the study, sentiment analysis categorized tweets as positive, neutral, or negative by assigning values between  $-1$  and  $+1$ , reflecting sentiments ranging from negative to positive.

For the initial keyword ‘Virat Kohli’, a total of 5313 tweets were amassed. Analysis revealed 2408 positive tweets (45%), 1739 neutral tweets (32%), and 1166 negative tweets (21%) (Table 1).

Likewise, for ‘Harmanpreet Kaur’, a total of 5100 tweets were gathered. Among these, there were 2640 positive tweets (51%), 2050 neutral tweets (40%), and 410 negative tweets (8%) (Table 2).

For ‘Vijender Singh’, a total of 4280 tweets were collected, comprising 1168 positive tweets (27%), 1685 neutral tweets (39%), and 1427 negative tweets (33%) (Table 3).

Subsequently, for ‘Sarita Devi’, there were 5003 tweets, including 1736 positive tweets (34%), 931 neutral tweets (18%), and 2336 negative tweets (46%) (Table 4).

For ‘Sushil Kumar’, a total of 4445 tweets were collected, including 1073 positive tweets (24%), 625 neutral tweets (14%), and 2747 negative tweets (61%) (Table 5).

For ‘Sakshi Malik’, a total of 5305 tweets were collected, comprising 1631 positive tweets (30%), 1851 neutral tweets (34%), and 1823 negative tweets (34%) (Table 6).

**Table 1** Examples of tweets for ‘Virat Kohli’ with positive, neutral, and negative sentiments, along with their corresponding sentiment values

Tweet	Sentiment	Sentiment value
Its biggest slap on Social Media and ViratKohli always make India proud everywhere @imV Kohli	Positive	0.5719
Last Indian bowler to take a T20 World Cup wicket was Virat Kohli 2040 days ago.: Cricket	Neutral	0.0000
Virat Kohli speaks up on Mohammad Shami: Here’s why netizens are not happy with the response via @OpIndia com wouldn’t it be poetic justice if Indians stop watching cricket?	Negative	-0.2018

**Table 2** Examples of tweets for ‘Harmanpreet Kaur’ with positive, neutral, and negative sentiments, along with their corresponding sentiment values

Tweet	Sentiment	Sentiment value
Viru Ji just knows one of you, I am a big fan of Rohit Sharma and Harmanpreet Kaur. I am very fond of them	Positive	0.6697
Sushma Verma with Veda Krishnamurthy, Harmanpreet Kaur	Neutral	0.0000
Had there been Harmanpreet Kaur in place of Tharanga she would’ve killed Lakmal after that runout	Negative	-0.6705

**Table 3** Examples of tweets for ‘Vijender Singh’ with positive, neutral, and negative sentiments, along with their corresponding sentiment values

Tweet	Sentiment	Sentiment value
@bigdaddybunce the support from the crowd for Vijender Singh was phenomenal! he definitely done India proud!	Positive	0.7494
Vijender Singh can do it: Devarajan—The New Indian Express via @NewIndianXpress	Neutral	0.000
Like all Criminals “@timesnow: Boxer Vijender Singh breaks his silence. Speaking exclusively to TIMES NOW, he said that he will bounce back.”	Negative	-0.2960

**Table 4** Examples of tweets for ‘Sarita Devi’ with positive, neutral, and negative sentiments, along with their corresponding sentiment values

Tweet	Sentiment	Sentiment value
Wow. Sarita Devi is here	Positive	0.5859
Tennis ace Rohan Bopanna and boxer Sarita Devi receiving their Bengaluru FC jerseys from Club COO Mustafa Ghouse	Neutral	0.0000
NDTV Sports—Sarita Devi Disrespected Opponents by Refusing Asian Games medal, Says Rahul Dravid	Negative	-0.3818

**Table 5** Examples of tweets for ‘Sushil Kumar’ with positive, neutral, and negative sentiments, along with their corresponding sentiment values

Tweet	Sentiment	Sentiment value
Anshu Malik misses out on gold but she is still a history maker in her own rights. She is the first Indian woman wrestler to win silver at the world championships. Meanwhile, India still does not have a world champ except Sushil Kumar	Positive	0.7730
Cops file second chargesheet in wrestler Sushil Kumar case	Neutral	0.0000
Delhi Police files second charge sheet in Chhatrasal Stadium murder case involving Olympic medallist wrestler Sushil Kumar	Negative	-0.6908

**Table 6** Examples of tweets for ‘Sakshi Malik’ with positive, neutral, and negative sentiments, along with their corresponding sentiment values

Tweet	Sentiment	Sentiment value
@SakshiMalik I am also your fan from today onwards	Positive	0.3182
Our movement is not political @SakshiMalik @BajrangPunia	Neutral	0.0000
@ravibhadoria @BajrangPunia @SakshiMalik @Phogat Vinesh All these wrestlers have got their work done by sending them abroad. Log in now, if you follow guys like wrestlers, you will definitely be defeated	Negative	-0.6908

**Table 7** Virat Kohli

Hashtag	Frequency
#ViratKohli	1246
#T20WorldCup	621
#INDvNZ	238
#TeamIndia	201
#INDvsNZ	195
#India	150
#viratkohli	130
#T20WorldCup21	106
#MohammedShami	102
#Cricket	101

**Table 8** Harmanpreet Kaur

Hashtag	Frequency
#HarmanPreetKaur	867
#WWC17Final	631
#WWC17	476
#INDvENG	285
#HarmanpreetKaur	276
#WomenInBlue	248
#IndvsEng	178
#MithaliRaj	156
#ENGWvINDW	102
#ENGvIND	66

### 5.1 Top 10 Hashtags Used

Tables 7, 8, 9, 10, 11 and 12 display the top 10 hashtags commonly used with each keyword. The primary hashtag for the athletes matches their names, mirroring the chosen keywords. Additional hashtags correspond to important events and themes relevant to the athlete's prominence during their 6-month news coverage.

### 5.2 Result Analysis

To comprehend the evolution of the public's sentiments towards their favourite athletes, sentiment versus time graphs were plotted. Additionally, average sentiment versus time graphs were plotted to enhance clarity. The x-axis represents the number of tweets, serving as a proxy for time as they are in ascending order, corresponding

**Table 9** Vijender Singh

Hashtag	Frequency
#News	130
#VijenderSingh	108
#India	63
#Sports	63
#news	54
#Vijender	45
#SuryaRay	42
#boxing	40
#india	39
#Ad	33

**Table 10** Sarita Devi

Hashtag	Frequency
#SaritaDevi	462
#AIBA	116
#Boxing	82
#Sports	78
#saritadevi	70
#Sarita	62
#news	60
#boxing	59
#India	48
#AsianGames	37

**Table 11** Sushil Kumar

Hashtag	Frequency
#SushilKumar	641
#DelhiPolice	120
#SushilKumarArrested	119
#Delhi	116
#wrestler	99
#Tokyo2020	80
#Wrestling	77
#Wrestler	72
#Olympics	67
#sushilkumar	62

**Table 12** Sakshi Malik

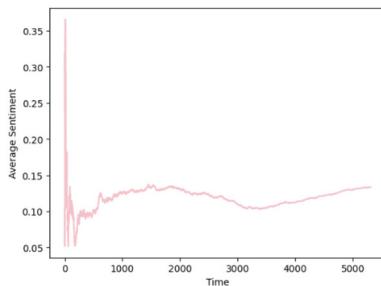
Hashtag	Frequency
#SakshiMalik	339
#WrestlersProtest	209
#BajrangPunia	128
#BrijBhushanSharanSingh	114
#VineshPhogat	87
#WrestlerProtest	76
#sakshimalik	65
#vineshphogat	45
#Wrestlers	42
#WFI	39

to changing sentiments over time. The y-axis reflects the sentiment values of the tweets, ranging from  $-1$  to  $+1$ .

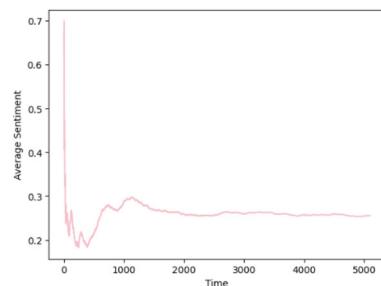
Figures 1, 2, 3, 4, 5, and 6 depict the average sentiment versus time graphs for the athletes Virat Kohli, Harmanpreet Kaur, Vijender Singh, Sarita Devi, Sushil Kumar, and Sakshi Malik, respectively.

The distribution of positive, neutral, and negative sentiments across all players combined has been represented using a bar graph (Fig. 7).

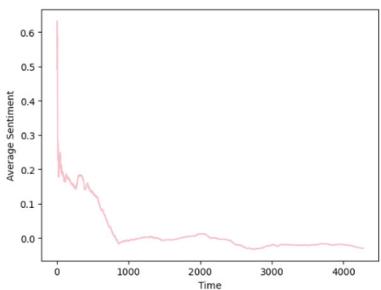
**Fig. 1** Average sentiment versus time graph for Virat Kohli



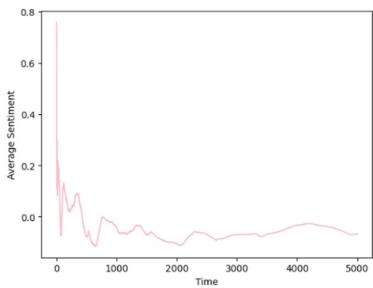
**Fig. 2** Average sentiment versus time graph for Harmanpreet Kaur



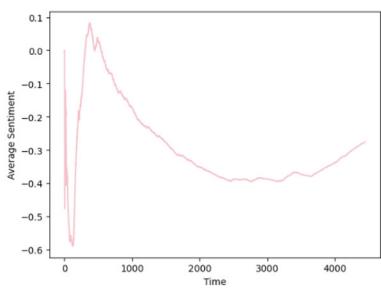
**Fig. 3** Average sentiment versus time graph for Vijender Singh



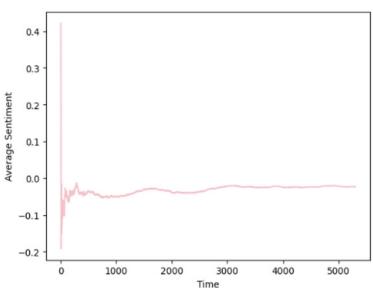
**Fig. 4** Average sentiment versus time graph for Sarita Devi

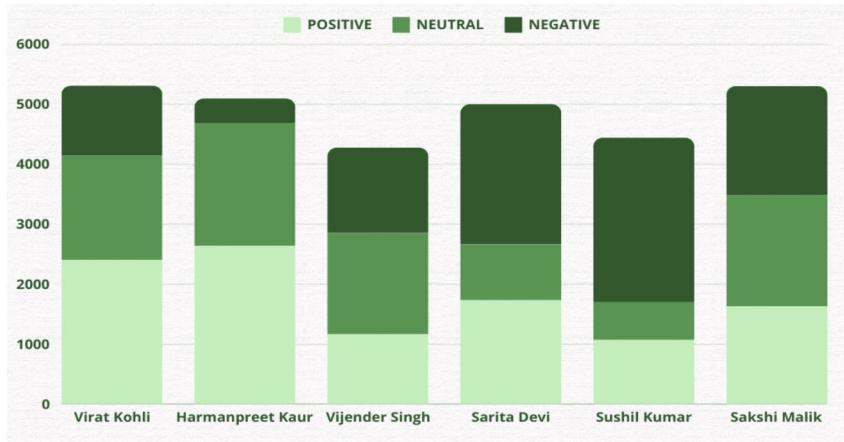


**Fig. 5** Average sentiment versus time graph for Sushil Kumar



**Fig. 6** Average sentiment versus time graph for Sakshi Malik





**Fig. 7** The graph illustrates the distribution of positive, neutral, and negative tweets for each of the six sports personalities

## 6 Discussion

This study aimed to quantify cyberbullying using sentiment analysis, focusing on the changing sentiments towards prominent sporting personalities over six months, aligned with their media prominence. Virat Kohli's news coverage centred around his relinquished captaincy of the Indian cricket team [24]. Harmanpreet Kaur gained attention for her unbeaten 171 against defending champions Australia in the 2017 World Cup semi-final [25]. Vijender Singh faced allegations of drug consumption [26], Sarita Devi attracted attention for refusing the bronze medal at the Asian Games [27], Sushil Kumar was involved in a murder investigation [28], and Sakshi Malik made headlines with her retirement from wrestling [29]. Despite being fan favourites during their peak performances, incidents on or off the field resulted in shifting of fan sentiments.

Top hashtags were instrumental in pinpointing factors contributing to a decline in popularity for some and a gain for others. Predominantly, the top hashtags mirrored the athlete's name, which was also the primary keyword during data collection. Additionally, notable hashtags provided context: #T20worldcup for Virat Kohli (count = 621), symbolizing the captaincy change post-World Cup; #WWC17Final for Harmanpreet Kaur (count = 631), highlighting her significance in the Indian batting lineup; #News for Vijender Singh (count = 130), for being in the news regarding alleged drug consumption; #AIBA for Sarita Devi (count = 116), indicating her ban after refusing the Asian Games bronze medal; #DelhiPolice for Sushil Kumar (count = 120), following his arrest by the Delhi police in a murder investigation; and #WrestlersProtest for Sakshi Malik (count = 209), referencing her involvement in a wrestlers' protest before announcing her retirement.

Through tweet collection, this study identified factors triggering extensive trolling, emphasizing the detrimental effects of social media on mental health and subsequent performance [30]. While implementing social media boycotts may present logistical challenges, preventive actions from social media platforms and governmental regulations [31] targeting online bullying could curb its detrimental effects. By fostering a safer and more supportive online environment [32], such initiatives have the potential to safeguard individuals from the harmful repercussions of cyberbullying, ultimately promoting healthier digital interactions and enhancing overall mental resilience [33].

This study is unique and novel as it quantifies how negative sentiments expressed on Twitter affects Indian sporting personalities' mental well-being. It provides new insights and a more precise understanding that distinguishes this work.

## 7 Limitations

This research paper provides valuable insights into the prevailing sentiments, it's essential to acknowledge certain limitations. One notable limitation is that our analysis does not account for false positives and false negatives. False positives occur when a sentiment is incorrectly classified as positive, while false negatives occur when a sentiment is incorrectly classified as negative. By not addressing these errors, our analysis may misrepresent the true sentiment of certain tweets, leading to potential inaccuracies in our findings. Future research could focus on developing a more sophisticated sentiment analysis technique that mitigates the impact of false positives and false negatives, thereby enhancing the reliability and validity of our results.

## 8 Conclusion

The relevance of tweets regarding cyberbullying aimed at prominent sports figures underscores the profound impact on athletes' mental well-being, organizational reputation, and societal norms. It highlights the urgent need for collective action to foster a safer and more supportive online culture, involving sports bodies, social media platforms, policymakers, and communities.

## 9 Future Work

This study holds significant future potential. It can explore the gender dynamics in cyberbullying, examining whether women in sports have different experiences compared to men. With the help of social network analysis, it can identify specific groups of individuals who consistently engage on social media platforms to spread negativity. This research aims to expand beyond sports to include individuals from

different sectors internationally, such as actors, politicians, social media influencers, and tech giants. By drawing parallels between industries and geographical regions, a comprehensive understanding of cyberbullying trends can be gained. These insights can help facilitate universally applicable strategies for combating online harassment and fostering a more positive online environment.

## References

1. Wikipedia. The definition of cyberbullying. <https://en.wikipedia.org/wiki/Cyberbullying>
2. Cuncic A (2023) The psychology of cyberbullying. <https://www.verywellmind.com/the-psychology-of-cyberbullying-5086615?print>
3. Sangani P (2022) Cyberbullying in children more widespread in India than elsewhere. <https://economictimes.indiatimes.com/tech/technology/cyberbullying-in-children-more-widespread-in-india-than-elsewhere/articleshow/93971435.cms?from=mdr>
4. Lane DK (2010) Taking the lead on cyberbullying: why schools can and should protect students online. *Iowa L. Rev* 96:1791
5. Patchin JW, Hinduja S (2010) Cyberbullying and self-esteem. *J School Health* 80(12):614–621
6. Torres CE, D'Alessio SJ, Stolzenberg L (2020) The effect of social, verbal, physical, and cyberbullying victimization on academic performance. *Victims Offend* 15(1):1–21
7. Stever GS, Lawson K (2013) Twitter as a way for celebrities to communicate with fans: Implications for the study of parasocial interaction. *North Am J Psychol* 15(2)
8. Samsudin EZ, Yaacob SS, Wee CX, Ruzlin AN, Azzani M, Jamil AT, Muzaini K, Ibrahim K, Sudin LS, Selamat MI et al (2023) Prevalence of cyberbullying victimisation and its association with family dysfunction, health behaviour and psychological distress among young adults in urban selangor, malaysia: a crosssectional study. *BMJ Open* 13(11):e072801
9. Taboada M (2016) Sentiment analysis: An overview from linguistics. *Ann Rev Linguist* 2:325–347
10. Zancan D (2022) Exploring the effects of negative user-generated tweets on athletes. Ph.D. thesis, Regent University
11. Meggs J, Ahmed W (2024) Applying cognitive analytic theory to understand the abuse of athletes on twitter. *Manag Sport Leisure* 29(1):161–170
12. El Rahman SA, AlOtaibi FA, AlShehri WA (2019) Sentiment analysis of twitter data. In: 2019 international conference on computer and information sciences (ICCIS). IEEE, pp 1–4
13. Vladkens (2023) How to still scrape millions of tweets in 2023 using twscrape. <https://medium.com/@vladkens/how-to-still-scrape-millions-of-tweets-in-2023-using-twscrape-97f5d3881434>
14. Marujo L, Ling W, Trancoso I, Dyer C, Black AW, Gershman A, de Matos DM, Neto JP, Carbonell JG (2015) Automatic keyword extraction on twitter. In: Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing: short papers, vol 2, pp 637–643
15. Hossny AH, Mitchell L (2018) Event detection in twitter: a keywordvolume approach. In: 2018 IEEE international conference on data mining workshops (ICDMW). IEEE, pp 1200–1208
16. Jannat M, Al A, Sikder S, Hossen I (2020) Sentiment extraction in Bangla Language using unsupervised learning approach. Ph.D. thesis. East West University
17. Yao J (2019) Automated sentiment analysis of text data with nltk. *J Phys: Conf Ser* 1187:052020 (IOP Publishing)
18. Bonta V, Kumares N, Janardhan N (2019) A comprehensive study on lexicon based approaches for sentiment analysis. *Asian J Comput Sci Technol* 8(S2):1–6
19. Hutto C, Gilbert E (2014) Vader: a parsimonious rule-based model for sentiment analysis of social media text. In: Proceedings of the international AAAI conference on web and social media, vol 8, pp 216–225

20. Das D, Kolya AK, Ekbal A, Bandyopadhyay S (2011) Temporal analysis of sentiment events—A visual realization and tracking. In: Computational linguistics and intelligent text processing: 12th international conference, CICLing 2011, Tokyo, Japan. Proceedings, Part I 12. Springer, pp 417–428
21. Thelwall M (2014) Sentiment analysis and time series with twitter. Twitter Soc 1
22. La Rocca G, Boccia Artieri G (2022) Research using hashtags: a meta-synthesis. Front Sociol 7:1081603
23. Buarki H, Alkhateeb B (2018) Use of hashtags to retrieve information on the web. Electron Libr 36(2):286–304
24. AFP (2021) Virat Kohli gets a last shot at world cup glory as India captain. <https://economictimes.indiatimes.com/news/sports/kohli-gets-a-last-shot-at-world-cup-glory-as-india-captain/articleshow/87011563.cms?from=mdr>
25. Sen R (2017) Harmanpreet kaur 171\* India's greatest world cup innings since Kapil Dev's 175\*? <https://www.indiatoday.in/sports/cricket/story/harmanpreet-kaur-171-india-vs-australia-womens-world-cup-kapil-dev-175-vs-zimbabwe-1025503-2017-July>
26. PTI (2013) Boxer Vijender Singh 'linked' to drug dealer. <https://indianexpress.com/article/news-archive/punjab-and-haryana/boxer-vijender-singh-linked-to-drug-dealer/>
27. Majumdar B (2014) Asian games 2014: judges Deny Sarita Devi place in boxing final, she refuses bronze medal. [https://economictimes.indiatimes.com/news/sports/asian-games-2014-judges-denied-sarita-devi-place-in-boxing-final-she-refuses-bronze-medal/articleshow/44073921.cms?utm\\_source=contentofinterest&utm\\_medium=text&utm\\_campaign=cpst](https://economictimes.indiatimes.com/news/sports/asian-games-2014-judges-denied-sarita-devi-place-in-boxing-final-she-refuses-bronze-medal/articleshow/44073921.cms?utm_source=contentofinterest&utm_medium=text&utm_campaign=cpst)
28. Sengar M (2021) Wrestler Sushil Kumar, arrested in murder case, suspended by railways. <https://www.ndtv.com/india-news/wrestler-sushil-kumar-arrested-in-murder-case-suspended-by-railways-2448968>
29. PTI (2023) Sakshi Malik announces retirement from wrestling. <https://sportstar.thehindu.com/wrestling/sakshi-malik-announces-retirement-sanjay-singh-wfi-president/article67661812.ece#:~:text=Sakshi%20Malik%20announced%20her%20retirement,the%20Wrestling%20Federation%20of%20India>
30. Gorrell E (2018) The impact of social media on athletes' self-efficacy
31. Kaur M, Saini M (2023) Indian government initiatives on cyberbullying: a case study on cyberbullying in Indian higher education institutions. Educ Inf Technol 28(1):581–615
32. Naslund JA, Aschbrenner KA, Marsch LA, Bartels SJ (2016) The future of mental health care: peer-to-peer support and social media. Epidemiol Psychiatr Sci 25(2):113–122
33. McLoughlin L, Spears B, Taddeo C (2018) The importance of social connection for cybervictims: how connectedness and technology could promote mental health and wellbeing in young people. Int J Emotion Educ 10(1):5–24

# AI-Based Adaptive Legal Analysis Engine for Enhanced Policing and Predictive Law Enforcement



Nagendra Singh, Abhishek Tiwari, Ruchi Tiwari, Priyanka Tiwari, Chaitanya Pushkarna, and Jitesh Choudhary

**Abstract** In today's rapidly evolving technological landscape, Law Enforcement Agencies (LEAs) in India have a unique opportunity to augment their capabilities. Addressing this need, we propose the Adaptive Legal Analysis Engine. A novel solution integrates cutting-edge large language models and custom training methodologies to support Investigation Officers (IOs) in interpreting and applying legal frameworks. Central to the engine's functionality is its custom training, with low-rank adaptation and quantization enabling efficient utilization of limited resources while ensuring accuracy and comprehensive coverage of relevant laws and facts. Also, vector embedding and vector databases are used to enhance performance of the model, capturing semantic relationships and contextual nuances within legal texts. The engine facilitates semantic understanding, similarity analysis, and contextualized recommendations by encoding legal provisions into high-dimensional vector spaces. The 4-bit quantization reduces the size of the model from 14 GB to 5.3 GB, and low-rank decomposition of gradient matrix reduces the RAM requirements of the model during fine-tuning. This paper comprehensively explains the development of the engine, detailing its architecture, training methodologies, integration of vector databases, and practical applications within the domain of Indian law enforcement.

**Keywords** LLM · Natural language processing · Machine Learning · Quantization · Law enforcement agencies · IPC · CrPC

---

N. Singh (✉) · J. Choudhary  
Center for Development of Advanced Computing CINE, Silchar, India  
e-mail: [nagendras@cdac.in](mailto:nagendras@cdac.in)

A. Tiwari · R. Tiwari · P. Tiwari · C. Pushkarna  
Center for Development of Advanced Computing Noida, Noida, India  
e-mail: [abhishek@cdac.in](mailto:abhishek@cdac.in)

## 1 Introduction

The legal landscape in India is marked by significant challenges that impede the timely delivery of justice. Delays and inefficiencies in the legal procedure and investigation process are pervasive, stemming from a reliance on human resources and subjective understanding. These factors often result in inconsistencies and delays in decision-making, leading to a backlog of cases and prolonged legal proceedings. Moreover, the absence of a standardized and quantifiable method for investigation, from filing an FIR to submitting a chargesheet, exacerbates these issues, making it arduous to streamline the process and expedite case disposal.

The data from the National Crime Records Bureau (NCRB) [1] paints a stark picture of the scale of the problem. In 2022 alone, there were a staggering 56,59,787 [1] total cases reported, out of which 35,61,379 were registered during the year, with an additional 20,41,140 carried forward from previous years. Despite efforts to charge sheet cases, the backlog continues to grow, with approximately 30 lakh cases forwarded in 2023. Moreover, National Judicial Data shows the number of cases pending in district courts and high courts since 1994 or before, i.e. 30 or more years are 103,274 [2], underscoring the systemic challenges faced by the Indian legal system. The chargesheet filing rate, as per NCRB data, has seen a decline over the past three years, with rates of 75.8% in 2020, 72.3% in 2021, and 71.3% in 2022. Additionally, the average chargesheet filing time of six months further contributes to delays in case disposal. These statistics highlight the urgent need for quantifiable solutions that can generate a logical flow for investigation, from filing an FIR to submitting a chargesheet, thereby expediting the legal process and ensuring timely delivery of justice.

In the face of handling challenges in the Indian legal system, a paradigm shift is needed to expedite the investigation process and ensure the timely delivery of justice. Recognizing the critical need for innovative solutions, this paper proposes a novel model named Adaptive FIR Analysis Engine. This proposed model represents a transformative approach to investigation procedures, harnessing the power of large language models (LLMs) to analyze FIR data and provide invaluable assistance to Investigation Officers (IOs). By leveraging advanced natural language processing (NLP) techniques, the Adaptive FIR Analysis Engine generates a structured investigation process and step-by-step guidelines to be followed during the investigation, ultimately leading to the seamless generation of a chargesheet.

At its core, the Adaptive FIR Analysis Engine is designed to address the inefficiencies and delays inherent in the current investigation process. By automating and standardizing key aspects of investigation, the model empowers IOs with a structured framework to navigate the complexities of legal proceedings with precision and efficiency. This paper seeks to comprehensively explore the Adaptive FIR Analysis Engine, delving into its architecture, functionality, and practical applications within the realm of Indian law enforcement. Through empirical evaluation and case studies, we aim to demonstrate the efficacy of this model in expediting the investi-

gation process, reducing delays, and enhancing the overall effectiveness of the legal system.

In addition to generating a structured investigation process, the Adaptive FIR Analysis Engine plays a pivotal role in assisting IOs in various critical tasks. It generates the sections to be applied in the FIR complaint, outlines the evidences to be collected, and provides detailed procedures to be followed, including guidelines for chemical examinations, arrests, and seizures. Moreover, the model streamlines the chargesheet generation process, ensuring accuracy and compliance with legal requirements.

The development of the Large Language Model-Based Adaptive FIR Analysis Engine is motivated by a combination of advancements in artificial intelligence (AI), including large language models (LLMs), the availability of extensive data, and the evolving legal landscape with the introduction of the new Bhartiya Nyaya Sahinta 2023 [3] and Bharatiya Sakshya Adhiniyam [4].

**Introduction of New Legislative Frameworks:** The introduction of new legislative frameworks, such as the Bhartiya Nyaya Sahinta 2023 and Bharatiya Shkshya Adhiniyam, underscores the evolving nature of the legal landscape in India. These new laws bring about changes in legal procedures, terminology, and requirements, necessitating an adaptive approach to legal analysis and interpretation. The Adaptive FIR Analysis Engine is designed to accommodate these changes, utilizing its flexibility and adaptability to incorporate updates and revisions to legal frameworks seamlessly.

**Need for Efficiency and Accuracy:** The challenges inherent in the Indian legal system, including delays in case disposal, backlog of pending cases, and inconsistencies in decision-making, highlight the urgent need for solutions that can enhance efficiency and accuracy in legal proceedings. The Adaptive FIR Analysis Engine addresses these challenges by automating and standardizing key aspects of the investigation process, thereby reducing delays, improving the quality of investigations, and ensuring the timely delivery of justice.

Further in this paper, we will explore the developed Adaptive FIR Analysis Engine, and understand its data processing, architecture, NLP algorithms, and learning mechanisms. Also, we will explore the potential impact of this model on investigation procedures, considering its implications for case disposal, resource allocation, and the investigation process.

## 2 Literature Review

The Indian legal system is confronted with multifaceted challenges that impede the efficient delivery of justice. Investigation procedures, in particular, have garnered significant attention from scholars due to their susceptibility to inefficiencies and delays. A LLM-based model is developed in [5] to analyze legal text and shows the challenges and limitations of the models for legal procedures and structured data. They have developed a LLM for legal judgment prediction for Indian court cases and

showed the potential of Large language models for law enforcement in India. They used a human-centric fine-tuning of a large language model, but it requires a lot of domain expert manpower to annotations and rating of LLM generated text, while the challenges faced during the deployment of LLMs in the legal domain are such as data bias, interpretability, and ethical considerations.

A methodology is presented in [6] for developing Large Language Models specifically designed for legal decision prediction and legal judgment prediction in context to Indian court cases. Data from Indian court cases are utilized for training the LLMs, and the paper provides insights into the specific architecture and fine-tuning processes involved in creating these models. The paper also evaluates the performance of the developed LLMs by measuring accuracy, precision, and recall in predicting legal statutes and judgments. The results of the evaluation provide evidence of the effectiveness of these LLMs in handling complex legal text analytics tasks. Despite their potential benefits, the deployment of LLMs in law enforcement [7] is not without challenges. One significant challenge is the interpretability and explainability of LLMs' outputs, particularly in complex legal contexts where transparency and accountability are paramount.

Ketz's [8] examines the potential of NLP in law enforcement, highlighting its impact on processes. However, challenges persist with large language models, particularly in error susceptibility. The paper underscores the need for further research to mitigate these challenges and maximize the benefits of NLP in law enforcement. Pre-trained LLMs, such as OpenAI's GPT [9] series, Google's BERT [10], and Meta's LLama [11], have revolutionized the field of NLP. These models, trained on massive corpora of text data, demonstrate remarkable capabilities in understanding and generating human language. Devlin et al. [10] introduce BERT, showcasing its effectiveness in various NLP tasks through contextual word embeddings.

Meta's Large Language Model Augmented with Meta-Learning And Context [12, 13] (LLama) model effectively addresses the limitations of traditional large language models by incorporating meta-learning techniques and contextual understanding. These models are highly capable to adapt and generalize across various tasks and domains, making it a versatile tool for natural language processing applications. The authors demonstrate LLama's superior performance on benchmark datasets and highlight its efficacy in tasks such as text generation, summary of the text, understanding language, and chat-based question answering.

Fine-tuning large language models has become a cornerstone technique in natural language processing, enabling the adaptation of pre-trained models to specific tasks or domains. It involves updating the parameters of pre-trained LLMs on task-specific data. Fine-tuning can be done through various methods like custom training of models, adding adaptors to the transformers, parameter efficient training, human feedback fine-tuning, low-rank adaption techniques, prompt engineering, etc.

Raffel et al. [11] provide a detailed exploration of fine-tuning techniques, including task-specific input transformations, task-specific output layers, and task-specific loss functions. These techniques enable the customization of pre-trained models to adapt different kinds of tasks, like text summary in specific field, text classification, named entity relation finding, and sentiment analysis for a field-specific context.

In addition to fine-tuning [10–13] pre-trained models, custom training involves training LLMs from scratch on domain-specific data. This approach allows for greater control over model architecture and training objectives, enabling the development of task-specific models optimized for particular applications [14]. Custom training of LLMs has been applied in various domains, including biomedical text mining, legal document analysis, and financial sentiment analysis [15]. Low-Rank Adaptation (LoRA) [16] is a technique designed by Microsoft researchers to enhance the fine-tuning process of Large Language Models (LLMs) by efficiently adapting them to specific tasks while minimizing computational overhead and memory requirements. LoRA's architecture is built upon the principles of low-rank approximation and adaptive parameterization. It decomposes the weight matrices of pre-trained LLMs into low-rank factors, reducing the number of weighted parameters with higher significance that need to be updated.

The literature survey shows that Indian legal system faces significant challenges in delivering efficient and timely justice, particularly in investigation procedures. Large Language Models (LLMs) can help to address these issues, such as legal judgment prediction, but face obstacles like data bias, interpretability, and ethical concerns. Human-centric fine-tuning of LLMs requires extensive expert annotation, and models like OpenAI's GPT [9], Google's BERT [10], and Meta's LLama [11] show promise but need further refinement. Techniques like Low-Rank Adaptation (LoRA) [16] help optimize LLM fine-tuning, improving adaptability and efficiency for specific tasks, yet the deployment in law enforcement demands careful consideration of transparency and accountability. In our work, we tried to resolve some of the challenges that are explained in further sections.

### 3 Proposed Scheme

The proposed solution uses Meta's open-source LLama 2 model as the base model that provides a strong foundation for the task at hand. However, to accommodate lower computational resources while still maintaining a high level of performance, the 7B parameter version of the LLama model is utilized. To fine-tune the model, we used the data from different sources; some of them are as follows:

**Data Collection:** We collected data using Indian court judgment data; multiple sources are utilized, including the official websites of Indian courts such as eCourts [17], IndianKanoon [18], Indian Legal Document Corpus (ILDC) [19], and Semantic Segmentation of Indian Supreme Court Case Documents [20], and data is generated from ChatGPT for the training. The data is annotated in a key-value format with specific keys such as complaint, applicable sections, procedure to be followed, action to be taken, and chargesheet. The data is converted to instruction dataset so as to use it in custom training the LLM. The instruction dataset is a representation of data in the form of instruction and answer pairs. We had used the Alpaca [21] dataset format to fine-tune the LLM model. Alpaca format consists of {instruction, input,

output} pairs. We had manually skimmed through the data to create a dataset in Alpaca format.

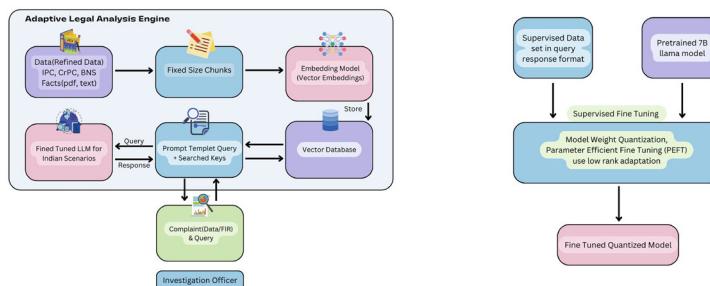
**System Configuration:** We have used a server node having 64 core intel CPU with 64GB RAM and a NVIDIA A4000 with 16GB GPU memory to fine-tune and train the weight matrices.

### 3.1 System Architecture

The system architecture Fig. 1a design for the proposed solution involves several components and processes to optimize model performance while accommodating resource constraints. It has the following components:

**Base Model:** Meta's LLama 2 7B model is chosen as the starting point for its proven effectiveness and lower size, so we can easily fine-tune and analyze with single accelerator card system. This pre-trained model provides a strong foundation for further customization and optimization.

**Fine-Tuning and Quantization:** The main methods used in fine-tuning include Supervised Fine-Tuning (SFT), Reinforcement Learning from Human Feedback (RLHF), and Parameter-Efficient Fine-Tuning (PEFT). SFT involves training the model on task-specific datasets under human supervision, enabling it to learn from labeled examples and generate more accurate responses. RLHF, on the other hand, leverages human feedback to guide the model's learning process, particularly useful for subjective tasks like conversation generation. Prompt templates provide structured frameworks for generating specific types of outputs, enhancing the model's efficiency in tasks requiring standardized or formatted responses. We used PEFT technique Fig. 1b to adapt the base LLama model for the specific requirements of law enforcement tasks. The supervised complaint query and response data are used for fine-tuning, allowing the model to learn from real-world legal complaints and develop domain-specific knowledge. Traditional fine-tuning uses Stochastic Gradient Descent (SGD) learning. It is required to update complete weight matrix that is



**Fig. 1 a** System architecture of Adaptive Legal Analysis Engine, **b** Fine-tuning of LLM Model

costly in terms of both computation and memory requirements. One learning iteration can be understood by the following equation:

$$W^{(i)} = W^{(i-1)} - \alpha \Delta W^{(i-1)} \quad (1)$$

where  $W^{(i)}$  represents the weight matrix of the model after  $i^{th}$  iterations of training.  $W^{(i-1)}$  is the weight of  $(i-1)^{th}$  iteration.  $\Delta W^{(i-1)}$  represents the weight update of  $i^{th}$  update iteration weight computed in epoch of forward pass learning.  $\alpha$  is the learning rate. The 7B parameter version of the LLama model is selected to have a balanced requirement between compute resources, memory, and performance, ensuring that the fine-tuned model meets both accuracy and efficiency requirements. In our case, LLama 2 7B model has weight matrix of 14GB if we consider 2 bytes per parameter in fp16. Gradients and optimizer matrix are also of the same size, so fine-tuning a complete 7B model with fp16 requires approximately memory of the range of 50-60 GB if we use traditional SGD method. So we developed a method inspired by LoRA [16] and QLoRa [22] PEFT fine-tuning methods.

First to reduce memory usage and computational complexity, the fine-tuned model is quantized into a 4-bit format using 4-bit Normal Form quantization. The 4-bit NormalFloat (NF4) format is a compact and optimized data representation format designed specifically for storing model weights that follow a normal distribution. It strikes a balance between precision and memory usage, making it a good choice in the case where memory requirement is a bottleneck. The memory requirement with low-rank adaption and quantization is in the range of 8-9GB.

We use low-rank adaption scheme to reduce the size of trainable parameters and gradient parameter matrix (as gradient parameter matrix is similar in size as weight matrix)  $\Delta W$ ; here we used LoRA adaptor that reduces the size of trainable parameters to a very low approximately 0.1–1% it drops the memory requirement significantly up to 5GB approximately with quantization. This compression technique preserves model accuracy while significantly reducing the model's memory requirements, so it is making it more suitable for deployment on low resource systems resource-constrained systems. To perform low-rank adaption we use rank matrix decomposition of the weight update matrix, and find out the suitable parameters to be tuned.

**Matrix Rank and Matrix Decomposition:** The rank of a matrix is determined by the dimension of the vector space formed by its columns, which corresponds to the count of linearly independent columns (or rows) within the matrix. This property holds true for any matrix represented as  $A_{m \times n}$ , where  $m$  denotes the number of rows and  $n$  signifies the number of columns:

$$\text{rank}(A) \leq \min(m, n) \quad (2)$$

A matrix is considered full-rank if  $\text{rank}(A) = \min(m, n)$ . A rank-deficient matrix is one where  $\text{rank}(A) < \min(m, n)$ , and a low-rank matrix is one where  $\text{rank}(A)$  is significantly less than  $\min(m, n)$ . Research shows that almost all deep learning and large language model weight matrices can be represented as low-rank matrix [16, 22, 23].

The weight matrix of the LLama model is  $W \in \mathbb{R}^{m \times n}$  similarly the gradient update matrix  $\Delta W \in \mathbb{R}^{m \times n}$ . We can decompose in matrices  $\Delta W = XY$  where  $X \in \mathbb{R}^{m \times r}$  and  $Y \in \mathbb{R}^{r \times m}$ . The rank  $r \ll \min(m, n)$ . The loss function for the low-rank adaption of the matrix can be formulated as follows:

$$L = \min_{\mathbf{X}, \mathbf{Y}} L(\mathbf{X}, \mathbf{Y}; \Theta) \quad (3)$$

where  $\mathbf{W}$  is the original weight matrix.  $\mathbf{X}$  and  $\mathbf{Y}$  are the low-rank matrices.  $L$  is the loss function.  $\Theta$  represents other parameters of the model. Weight matrices of the fine-tuned model are decomposed into low-rank approximations, reducing the number of parameters and minimizing memory requirements without sacrificing performance.

**Vector Embeddings for Legal Text:** Legal texts, including facts, sections of relevant legal codes, and other documents, are converted into vector embeddings. These embeddings capture semantic information about the text, enabling efficient storage and retrieval of legal documents from a vector database. We have used pre-trained model word2vec embedding model to convert IPC, CrPC, Bhartiya Nyaya Sahinta, and Bhartiya Sakshya Adhiniyam data and case facts from ILDC data. First we converted the data into chunks of length 500 and used ward2vec model for embedding. This data is stored in Chroma DB, an open-source vector database.

**Prompt Generation using Vector Search:** When an Investigation Officer (IO) submits an FIR copy and query, relevant keywords and phrases are extracted from the text. Vector search techniques are used to retrieve similar legal cases and precedents from the vector database based on the extracted information. Prompts for the fine-tuned LLM are generated using retrieved documents and the IO's query, providing contextually relevant input to the model. The prompt generated by combining searched vector and query gives better output. The results of the different models of similar class are compared; it shows that the combination of fine-tuned model and modified prompts gives the most suitable response, a query of crime investigation domain.

## 4 Performance Analysis

The results in Table 1 show a notable reduction in the model's memory requirements, making its deployment easy on single-GPU systems with lower RAM specifications. Specifically, the 4-bit quantized model with low-rank adaptation reduces RAM requirements within the range of 6-8GB, and ensures seamless operation even on resource-constrained setups. These diminished memory requirements not only streamlines fine-tuning processes but also enhances accessibility, making it viable for deployment on a broader spectrum of hardware configurations.

The prompt engineering used in the solution with vector search plays a pivotal role in augmenting model performance. By generating prompts optimized through vector search, the model yields more refined and contextually prominent responses

**Table 1** Size of the LLama2 7 billion parameter model with different quantization

Quantization	Size in GB	RAM in GB
Actual [23]	14.0	84
8-Bit Quantization	9.06	24
4-Bit Quantization	5.3	11.5
4-Bit Quantization with low-rank adaptation	5.3	6.9

post fine-tuning. This optimization process significantly enhances output quality, surpassing conventional approaches.

## 5 Conclusion and Future work

This paper shows the possibilities of model optimization and methods to customize pre-trained large language model with low resource systems. A key part of this development is combining advanced techniques for optimization, including low-rank adaptation and quantization. The 4-bit quantization and low-rank decomposition greatly reduce the model's size and memory needs from 11.5GB to 6.9GB, making it possible to deploy on systems with limited resources. The vector embedding and prompt engineering show faster adaptivity toward new data. The proposed model can be used to accelerate current law proceedings, and this engine can be used as base model for law enforcement agencies to generate a trustworthy data with the reasoning and backpropagation to the facts. The refinement of prompt engineering methodologies and exploration of novel optimization strategies for vector search for augmenting output relevance and coherence and scalability assessments across diverse datasets and model architectures could expand the applicability of these techniques to more complex real-world scenarios. Also, conducting thorough user-centric evaluations will provide valuable insights into the practical utility and user experience of the fine-tuned models, guiding future developments and deployments in real-world applications.

## References

1. NCRB-Crime in India 2022 report. <https://ncrb.gov.in/uploads/nationalcrimerecordsbureau/custom/1701607577CrimeinIndia2022Book1.pdf>
2. National Judicial Data Grid. [https://njdg.ecourts.gov.in/njdgnew/?p=main/pend\\_dashboard](https://njdg.ecourts.gov.in/njdgnew/?p=main/pend_dashboard)
3. The Bharatiya Nyaya Sanhita (2023) Ministry of Home Affairs. [https://www.mha.gov.in/sites/default/files/250883\\_english\\_01042024.pdf](https://www.mha.gov.in/sites/default/files/250883_english_01042024.pdf)
4. The Bhartiya Sakshya Adhiniyam (BSA) (2023). <http://wccb.gov.in/WriteReadData/UserFiles/file/Notification/BSA2023.pdf>

5. Ghosh S, Verma D, Ganesan B, Bindal P, Kumar V, Bhatnagar V (2024) Human centered AI for Indian legal text analytics (2024). [arXiv:abs/2403.10944](https://arxiv.org/abs/2403.10944)
6. Lai J, Gan W, Wu J, Qi Z, Yu PS (2023) Large language models in law: a survey (2023). [arXiv:abs/2312.03718](https://arxiv.org/abs/2312.03718)
7. Lipton Zachary Chase (2016) The mythos of model interpretability. Commun ACM 61:36–43
8. Katz DM, Hartung D, Gerlach L, Jana A, Bommarito MJ (2023) Natural language processing in the legal domain (2023). [arXiv:abs/2302.12039](https://arxiv.org/abs/2302.12039)
9. Ray PP (2023) ChatGPT: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope. <https://doi.org/10.1016/j.iotcps.2023.04.003>
10. Devlin J, Chang M-W, Lee K, Toutanova K (2019) BERT: pre-training of deep bidirectional transformers for language understanding. North American Chapter of the Association for Computational Linguistics
11. Raffel C, Shazeer NM, Roberts A, Lee K, Narang S, Matena M, Zhou Y, Li W, Liu PJ (2020) Exploring the limits of transfer learning with a unified text-to-text transformer. J Mach Learn Res 21: 140:1–140:67. <https://arxiv.org/abs/1910.10683>
12. Touvron H, Martin L, Stone KR, Albert P (2023) Llama 2: open foundation and fine-tuned chat models. [arXiv: abs/2307.09288](https://arxiv.org/abs/2307.09288)
13. Touvron H, Lavril T, Izacard G, Martinet X, Lachaux M-A, Lacroix T (2023) LLaMA: open and efficient foundation language models. [arXiv:abs/2302.13971](https://arxiv.org/abs/2302.13971)
14. Brown TB, Mann B, Ryder N (2020) Language models are few-shot learners. [arXiv:abs/2005.14165](https://arxiv.org/abs/2005.14165)
15. Beltagy IZ, Lo K, Cohan A (2019) SciBERT: a pretrained language model for scientific text. In: Conference on empirical methods in natural language processing (2019)
16. Hu JE, Shen Y, Wallis P, Allen-Zhu Z, Li Y (2021) LoRA: low-rank adaptation of large language models. [arXiv:abs/2106.09685](https://arxiv.org/abs/2106.09685)
17. Ecourts services: Highcourts of india, District and Taluka courts of India. [https://judgments.ecourts.gov.in/pdfsearch/?p=pdf\\_search/index&escr\\_flag=&app\\_token=](https://judgments.ecourts.gov.in/pdfsearch/?p=pdf_search/index&escr_flag=&app_token=)
18. Indian Kanoon-Search engine for Indian Law. <https://indiankanoon.org/>
19. ILDC (Indian Legal Documents Corpus). <https://github.com/Exploration-Lab/CJPE/tree/main/Data>
20. Semantic Segmentation of Indian Supreme Court Case Documents. <https://github.com/Law-AI/semantic-segmentation>
21. Taori R, Gulrajani I, Zhang T, Dubois Y, Li X, Guestrin C, Liang P, Hashimoto TB (2023) AlpacaFarm: a simulation framework for methods that learn from human feedback. <https://crfm.stanford.edu/blog.html>
22. Dettmers T, Pagnoni A, Holtzman A, Zettlemoyer L (2023) QLoRA: efficient finetuning of quantized LLMs. [arXiv:abs/2305.14314](https://arxiv.org/abs/2305.14314)
23. Llama 2: Efficient Fine-tuning Using Low-Rank Adaptation (LoRA) on Single GPU. <https://infohub.delltechnologies.com/en-us/p/llama-2-efficient-fine-tuning-using-low-rank-dadaptation-lora-on-single-gpu>

# AI-Based Data Analytics & Business Intelligence Chatbot Using Azure Functions and OpenAI



N. Praveen Sundra Kumar, S. Ramakrishnan, and M. Vignesh

**Abstract** A chatbot plays a vital role in enhancing user engagement and satisfaction through real-time interactions, especially in promptly addressing customer queries, revolutionizing business processes. Most chatbots follow rule-based systems, responding based on predefined rules. However, they face difficulties with out-of-scope questions, often requiring human intervention. Relying solely on rules may hinder accurate interpretation of diverse grammatical structures, potentially leading to inaccurate responses. These chatbots support limited languages and lack autonomy, functioning strictly within predefined rules. While efficient with certain queries, they may struggle with those outside their rule set, despite utilizing algorithms like Binary Search Tree (BST) for query processing. The proposed chatbot utilizes advanced AI to interpret natural language queries and extract insights from predefined data models, streamlining data analysis and business intelligence frameworks to enhance decision-making capabilities. This multilingual chatbot streamlines business insights acquisition by allowing users to engage in their preferred language, emphasizing AI's role in optimizing data analysis efficiency, enabling informed decision-making, and reducing man-hours for increased productivity and cost savings.

**Keywords** Artificial intelligence · Data analytics · Business intelligence · Productivity · Cost savings · Binary search tree

## 1 Introduction

An essential feature of this chatbot is its robust multilingual support, ensuring a user-friendly experience for a diverse global user base. The project aims to streamline acquiring business insights, focusing on efficiency and user-friendliness while facilitating the seamless integration of AI into data analysis and reporting for more intuitive business intelligence solutions.

---

N. P. S. Kumar · S. Ramakrishnan · M. Vignesh (✉)  
Dr. Mahalingam College of Engineering and Technology, Pollachi, India  
e-mail: [vignesh.slm03@gmail.com](mailto:vignesh.slm03@gmail.com)

This system automates query writing in SQL databases by generating user data models based on the dataset, eliminating manual intervention. Access is restricted to authorized users, with an auto-session termination feature ensuring database performance. Leveraging OpenAI's natural language processing capabilities [1], users can compose queries in multiple languages, enhancing system accessibility. Incorporating FAQs with default queries streamlines new user onboarding, and the chatbot's versatility extends to data representation, offering table formats or graphical representations based on user preferences through Azure functions. This sophisticated yet user-friendly approach empowers users and organizations in data analytics and business intelligence.

## 2 Literature Review

Natural language applications, especially chatbot systems, have garnered substantial attention in emerging technologies for businesses. Acknowledged for their potential contributions to online businesses, chatbots are recognized as intelligent conversational agents facilitating interactions between human users and computer devices. Recognition of chatbots as revolutionary technologies by MIT and industry leaders like Microsoft CEO Satya Nadella underscores their growing importance [2].

The technical aspects of chatbots, categorized as either independent or web-based, involve intelligent machine-to-human conversation systems achieved through text or speech [3].

Leveraging OpenAI's natural language processing capabilities, users can compose queries in multiple languages, enhancing the system's accessibility [6].

In conclusion, this research paper compares various language models: Text-Embedding-Ada-002 for Embedding Creation, GPT-3.5 Turbo for Document Chatbot, GPT-4 (8 k) for SQL Generation, GPT-4 (32 k) for Transcription, Audio Analysis, and Entity Extraction, Whisper for Audio to Text Transcription, and Reflection on Deprecated Model (Text-Davinci-003).

## 3 Existing System

The current system stores business transactions in a centralized database. When users need access to business data, they request it through various channels like in-person interactions, phone calls, or emails. The IT team manually formulates SQL queries in response, which is time-consuming and hampers overall developer productivity [4, 5].

The current system stores business transactions in a centralized database, with users accessing data through multiple channels. Manual SQL query formulation by the IT team causes inefficiencies, consuming time and impacting productivity. Delays in query formulation affect user satisfaction and system efficiency. Users'

challenges with SQL syntax proficiency lead to suboptimal queries and potential database inefficiencies. Automation is necessary to streamline data retrieval and optimize query processes for improved effectiveness [7, 8].

## 4 Proposed System

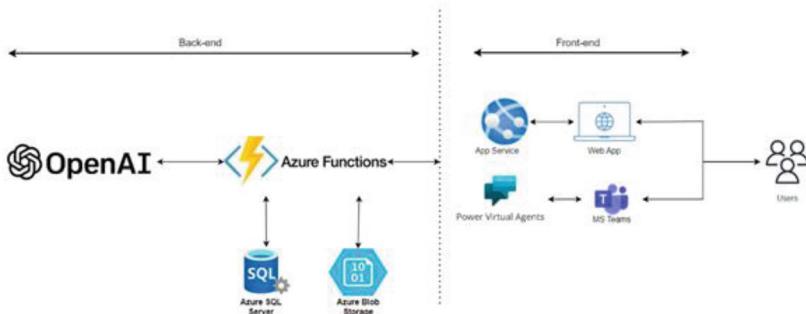
This section describes the schematic design architecture of the proposed methodology.

The proposed system as shown in Fig. 1 revolutionizes data retrieval by eliminating manual query writing or email requests, offering a dedicated application and integrated Microsoft Teams app for easy query submission. Users can effortlessly submit queries in natural language, enabling automatic data retrieval in various formats. Integrated with OpenAI, it supports queries in multiple languages, enhancing accessibility. Developed using Python Flask, the web app seamlessly integrates with Microsoft Teams and functions independently, providing users flexibility. By reducing IT dependencies, it aims to optimize efficiency, enabling swift access to business insights.

### 4.1 Language Model Analysis Used in Open AI

#### Text-Embedding-Ada-002 (Embedding Creation)

**Feedback:** The model demonstrated high efficiency in converting textual information into numerical embeddings, proving pivotal for clustering and content similarity assessments. The accuracy of these embeddings significantly influenced our ability to categorize and retrieve information based on content similarity.



**Fig. 1** DataDialogAI deployment architecture with front-end and back-end connectivity

**Comparison:** Compared to its predecessors, Ada-002 provided more nuanced embeddings, effectively capturing subtleties in textual data. This heightened performance in tasks requiring semantic understanding.

### GPT-3.5 Turbo (Document Chatbot)

**Feedback:** The model excelled in handling conversational queries, delivering accurate and contextually relevant responses promptly. It notably enhanced user experience with swift, coherent, and context-aware interactions.

**Comparison:** Outperforming its predecessors, GPT-3.5 Turbo reduced response times and increased answer relevance. Its efficiency particularly suited real-time customer service applications, demanding quick and accurate responses.

### GPT-4 (8k for SQL Generation)

**Feedback:** GPT-4's proficiency in comprehending complex queries and generating precise SQL statements streamlined data retrieval and analysis tasks. Its extended token limit allowed handling more detailed queries than previous models.

**Comparison:** The transition to GPT-4 for SQL generation marked a substantial improvement, outpacing earlier models in both accuracy and complexity handling, showcasing enhanced capabilities in understanding natural language queries.

### GPT-4 (32k for Transcription, Audio Analysis, and Entity Extraction)

**Feedback:** The 32k version's extended token limit and context improved accuracy in transcription and entity extraction from audio sources. It excelled in detailed content analysis, capturing nuanced information across extended dialogues.

**Comparison:** GPT-4 (32k) surpassed its predecessors and other models, showcasing unparalleled capabilities in handling extensive data inputs for transcription and analysis, significantly enhancing entity extraction and content analysis.

### Whisper (Audio to Text Transcription)

**Feedback:** Whisper exhibited exceptional accuracy in transcribing audio to text, adeptly handling various accents, dialects, and noisy backgrounds. Its performance was crucial for accurate data capture in audio analysis and entity extraction.

**Comparison:** Whisper's transcription capabilities surpassed traditional models, especially in terms of accuracy and robustness across different languages and audio qualities. It significantly reduced manual correction time for transcribed texts.

### Reflection on Deprecated Model (Text-Davinci-003)

**Feedback:** While Text-Davinci-003 was robust in its time, transitioning to newer models like GPT-3.5 Turbo brought noticeable improvements in processing speed, response accuracy, and complex query handling.

**Comparison:** The shift to GPT-4 and GPT-3.5 Turbo highlighted the rapid advancements in language model capabilities, emphasizing the importance of upgrading for enhanced performance and efficiency in natural language tasks.

## Overall Insights

**Adaptation and Evolution:** Migrating to the latest models, such as GPT-4, and utilizing specialized models like Whisper has significantly improved the efficiency, accuracy, and scope of our AI-driven applications.

**Performance Gains:** Each model contributed specific strengths to its application area, enhancing interaction quality, improving complex task accuracy, and streamlining data processing workflows.

**Future Directions:** Continuous evaluation and adoption of newer model versions ensure our solutions stay at the forefront of AI capabilities, meeting evolving demands and expanding the potential of our applications.

## 5 Proposed Algorithm

Various approaches exist for keyword detection, including TF-IDF, frequency-based approaches, text rank algorithm, noun phrase extraction, and rule-based methods. In this project, we chose the TF-IDF algorithm for its simplicity and effectiveness, though its limitations are acknowledged. In the era of big data, novel data processing techniques are essential before analysis. Enhanced versions like adaptive TF-IDF, incorporating hill climbing, and variants applicable across languages using statistical translation methods have been proposed by many researchers.

**Term Frequency (TF):** This measures how frequently a term appears in a document.

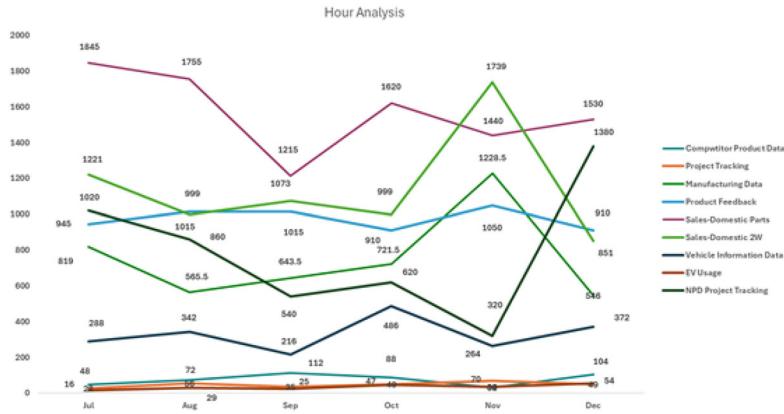
$$\text{freq}(\text{text}, \text{word}) = \text{count of text in given document}/\text{number of words in given document}$$

**Inverse Document Frequency (IDF):** This measures the rarity of a term across the entire document collection. The IDF score increases proportionally to the rarity of the term.

$$\text{document frequency}(\text{text}) = \text{occurrence of text in given documents}$$

$$\text{document frequency}(\text{text}) = \text{Number of documents containing the term text}$$

By combining the TF and IDF scores, the TF-IDF algorithm assigns higher weights to terms that are both frequent within a document and rare across the entire document collection.



**Fig. 2** Hour-based analysis for data received based on data models

## 6 Result Analysis

The experiment results of the project involve a thorough comparison between the current system and the proposed system, aiming to increase productivity while reducing labor costs.

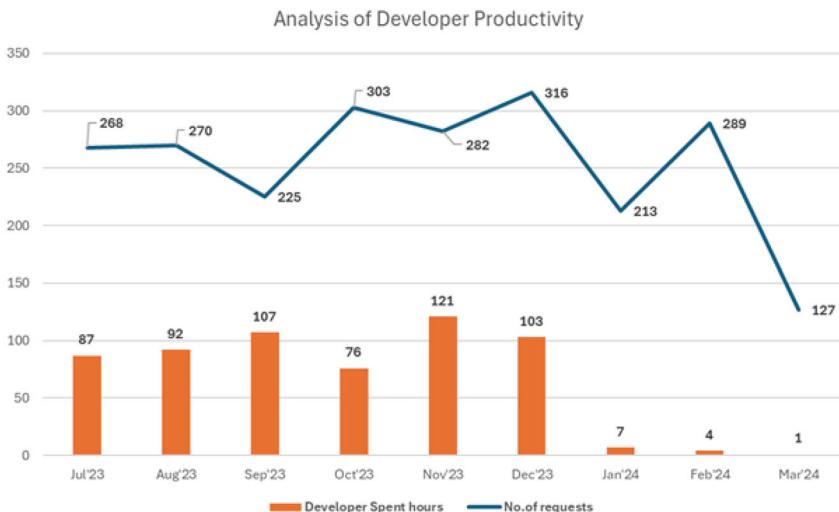
### 6.1 Existing System Process and Man Hours' Analysis

From July 2023 to December 2023, we analyzed data requests to determine the average waiting time for data retrieval. Delays experienced by individuals were extrapolated to represent the collective delay experienced by team members awaiting data access.

The analysis reveals that stakeholders collectively spent approximately 34,314 man-hours from July 2023 to December 2023, as illustrated in Fig. 2. Among these, developers dedicated 586 h to retrieving data from the database server and writing queries.

### 6.2 Experiments of the Proposed System with Productivity Improvement

We introduce a new method enabling users to access data without needing SQL expertise. Users interact via a chatbot interface, supporting over 190 languages through OpenAI's NLP capabilities. The architecture employs Azure Functions to link OpenAI and user interface, using Python Flask web app and Microsoft Teams



**Fig. 3** Developer productivity analysis

for interaction. User input is routed to Azure Functions via App Services, which process requests and formulate queries based on the data model. Results are returned through the chosen interface.

After implementing the new system, developers have experienced increased productivity, allowing them to focus more on their regular tasks and reducing non-value-added work. Analysis indicates that from January 2024 to March 2024, developers spent a total of 12 man-hours, compared to 586 man-hours for the existing system from July 2023 to December 2023. Detailed data is available in Fig. 3.

### 6.3 Cost–Benefit Analysis Through This Proposed System

Based on the analysis of the existing system from July 2023 to December 2023, stakeholders and end users collectively spent 34,314 man-hours, while developers spent 586 man-hours. Upon implementing the new system, developers' non-value-added work decreased by 98%, and waiting time for stakeholders and end users was eliminated. The total time invested in developing the new system was 900 man-hours.

#### 6.3.1 Cost Savings Calculation

**Man-hours saved:** Existing system hours (34,314) – new system development hours (900) = 33,414 man-hours.

**Cost saved:** Saved man-hours (33,414) × Average hourly salary (₹250) = ₹8,353,500.

The available manpower reduced due to the development process is calculated as the difference between the total man-hours spent from July 2023 to December 2023 and the time spent on developing the new system, resulting in 33,414 man-hours. Considering an average man hourly salary of 250 rupees, the total cost of man-hours is computed as 83,53,500 rupees.

The new system implementation offers significant cost savings of approximately ₹ 0.835 Crore by reducing non-essential developer work and eliminating waiting times for stakeholders and end-users. This translates to a direct financial benefit for the company.

## 7 Comparison of Different Algorithms

Considering the comparison, TF-IDF emerges as optimal for general keyword extraction and document summarization, as it considers both a word's frequency within a document and its rarity across the entire corpus. In tasks focused on identifying general keywords and prioritizing simplicity, TF-IDF proves advantageous. Furthermore, combining approaches can enhance results, with TF-IDF serving as a proficient initial keyword extractor followed by rule-based refinement. Moreover, for large datasets, TF-IDF may exhibit computational efficiency based on data characteristics (Table 1).

## 8 Conclusion

This paper introduces an AI-driven chatbot revolutionizing data analysis and business intelligence processes. We present a transformative solution aimed at reshaping the landscape of data analysis and business intelligence. The chatbot leverages advanced AI techniques to interpret natural language queries, extract insights from predefined data models, and provide real-time meaningful business data.

Through the implementation of the proposed system, significant improvements have been observed in terms of developer productivity, reduction of non-value-added work, and elimination of waiting time for stakeholders and end-users. The cost-saving analysis indicates substantial financial benefits, with the new system offering savings of approximately 100 Million US Dollars.

**Table 1** Algorithm comparison table

Technique	Strength	Weakness	Best suited for	Technique
TF-IDF	Keyword extraction, document summarization	Computationally expensive may miss context	General keyword extraction, document summarization	TF-IDF
Frequency-based	Simple, efficient	Ignore words importance across documents	Identifying frequently occurring terms, simple tasks	Frequency-based
Test rank	Capture semantic relationship, key phrase extraction	Sensitive to parameters, computationally expensive for large texts	Understanding connections between words, key phrase extraction	Test rank
Noun phrase extraction	Entity recognition information extraction	Might miss non-phrase concepts requires further processing	Named entity recognition, information extraction	Noun phrase extraction
Rule-based methods	High accuracy specific domains	Limited flexibility, requires domain expertise, time-consuming rule creation	Domain-specific information extraction with well-defined rule	Rule-based methods

## References

1. Dale R (2016) The return of the chatbots. Nat Lang Eng 22(5):811–817
2. Erne R (2011) What is productivity in knowledge work?—A cross-industrial view. J Univ Comput Sci 17(10):1367–1389
3. Jusoh S (2018) A study on NLP applications and ambiguity problems. J Theor Appl Inf Technol 96(6)
4. Hilton J, Nakano R, Balaji S, Schulman J (2021) WebGPT: improving the factual accuracy of language models through web browsing. <https://openai.com/research/webgpt>
5. Agarap Will be the year of conversational commerce. <https://medium.com/chris-messina/2016-will-be-the-year-of-conversational-commerce-1586e85e3991>
6. OpenAI (2023) How should AI systems behave, and who should decide?. <https://openai.com/blog/how-should-ai-systems-behave>
7. Khlaaf H (2023) Toward comprehensive risk assessments and assurance of AI-based systems. Trail of Bits
8. Fan H, Qin Y (2018) Research on text classification based on improved TF-IDF algorithm. In: International conference on network, communication, computer engineering (NCCE 2018), vol 147

# Assistive Live Audio Transcription Glasses for Individuals Suffering Auditory Impairment



Siddharth Menon, Aparna Padma Balaji, Jayant Sasikumar, Thazhai Mugunthan, V. Ravikumar Pandi, Soumya Sathyan, Vipina Valsan, and Kavya Suresh

**Abstract** In this paper, a comprehensive study on the development and implementation of Augmented Reality (AR) glasses designed to provide real-time text conversion of live speech for individuals with hearing impairments is presented. The work aims to break communication barriers, enhance accessibility, and promote inclusivity by merging AR technology with speech recognition and natural language processing. These glasses offer live subtitles, improving communication access and confidence for the deaf and hard-of-hearing community. The technical implementation involves utilizing a speech recognition AI model Whisper to transcribe audio into text and displaying onto the glasses via specular reflection from an OLED display through NodeMCU (ESP32), which is reflected on the mirror in front of the glass. The system effectively converts speech to text, enhancing communication and accessibility. The novelty of this work lies in designing the live audio transcription glasses to show the text clearly on the glass and including offline functionality for transcription.

**Keywords** Accessibility · AR glasses · Augmented reality · Hearing impairments · Speech recognition · Real-time subtitles

## 1 Introduction

People with hearing difficulties can be found in all age groups, races, and ethnic groups, as well as in all socioeconomic and geographic origins. While some people lose their hearing as a result of disease, trauma, aging, or medical conditions, some people are deaf from birth. To get over this, persons afflicted might be able to use a

---

S. Menon · A. P. Balaji · J. Sasikumar · T. Mugunthan

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

V. R. Pandi (✉) · S. Sathyan · V. Valsan · K. Suresh

Department of Electrical and Electronics Engineering, Amrita Vishwa Vidyapeetham, Amritapuri, India

e-mail: [ravikumarpandiv@am.amrita.edu](mailto:ravikumarpandiv@am.amrita.edu)

cochlear implant or hearing device to help them hear better, while others might not be able to hear anything at all. The community as a whole is as diverse as the needs and capacities of people with hearing impairments.

A recent study highlighted the impact of hearing loss on the quality of life of elderly individuals in nursing homes [1]. It revealed that hearing loss can cause distress, anxiety, and depression, leading to feelings of loneliness and isolation. The use of hearing aids showed positive effects on the participants' quality of life, emphasizing the potential of assistive technologies in improving their well-being. However, accessing and maintaining hearing aids can still be challenging for some individuals. This emphasizes the need for ongoing research, development, and awareness to reduce barriers and ensure better accessibility.

## 2 Literature Review

Communication is a fundamental aspect of human interaction, enabling the exchange of ideas, emotions, and information. However, for individuals with hearing impairments, this essential ability can be significantly compromised. According to the World Health Organization (WHO), around 5% of the global population, which is an estimated 432 million adults and 34 million children, are in need of assistance to manage their hearing loss [2]. Some individuals with hearing impairments find hope and improved communication through cochlear implants or hearing aids, which can provide varying degrees of auditory support. For others, hearing loss remains profound, necessitating alternative means of communication, such as sign language or assistive technologies. By merging AR technology and speech recognition, it aims to empower individuals with hearing impairments.

The design procedure for software-based algorithms with sensors and other IoT communications is important in effective development of prototype [3]. There are several applications utilizing Machine learning algorithms along with IoT-based communications such as Smart home with condition monitoring [4], automated fault detection [5], smart traffic light with image processing [6], change detection of forest vegetation [7], and fire detection [8]. Hence, the machine learning algorithms with IoT method will bring the successful design of live audio transcription glasses.

A review of an existing literature explores the evolution of interaction methods for smart glasses, highlighting the key approaches and technologies used to facilitate user interaction mentioned in [9]. This study provides insights into the challenges and advancements in this rapidly evolving field, highlighting its potential impact on various domains. The real-time transcription of spoken language, aligning with the growing interest in leveraging augmented reality to enhance accessibility and assistive technologies for the hearing-impaired community is provided in [10]. The Fresnel lens-based imaging challenges are tackled, and post-capture image processing techniques are introduced to improve image quality [11]. Effective implementation of AR glasses by incorporating OLED displays and Fresnel lenses offers avenues to enhance visual clarity and user comfort [12]. A curved Fresnel lenses which involves dividing

a traditional lens into concentric annular sections, each with a slightly different curvature to create a compact and lightweight optical system for projecting information onto a wearer's field of view [13].

The potential of combining AR and speech technologies can be used to improve communication for the visually challenged, deaf, and hard of hearing [14]. To help visually challenged people in reading names of people, products, objects, and texts is detected using a camera-based system as offered in [15]. An image processing-based object recognition model and its corresponding text reader using Google text to translator [16] is designed to help the visually impaired persons. An IoT-based smart wearable equipment for visually challenged people with [17] facial recognition.

This work focuses on addressing communication challenges faced by the deaf and hard of hearing, contributing to improving accessibility and engaging exchanges while enhancing the overall quality of interpersonal connections. The implementation of AR glasses significantly enriches social interactions for individuals with hearing impairments. Common scenarios, such as group conversations or noisy environments, can now be navigated with greater ease. Users may easily follow and engage in discussions due to the ability of smart glasses to show speech-translated text, facilitating a more fluid and natural exchange of ideas. This improvement makes people feel more involved and like they belong in social circles.

Without the need of translators or specialized equipment, deaf or hard-of-hearing people can converse with hearing people intelligibly. This inclusivity encompasses a broader spectrum of social circumstances, including professional and social settings, where all are welcome to engage equally and foster a more varied and stimulating exchange of ideas.

In summary, the literature review provides insights into the progression of innovations in AR technology, speech-to-text conversion, optical integration, and accessibility. This synthesis serves as the bedrock for the current study, uncovering its significance, breadth, and potential impact in harnessing AR glasses for instant text conversion. By amalgamating insights from existing research, this endeavor aspires to enrich the ongoing dialogue concerning augmented reality's ability to augment communication accessibility and enhance the quality of life for individuals with hearing impairments.

### 3 Smart Glass Prototype

The smart glass prototype is designed to include hardware such as OLED DISPLAY, ESP32, Power source (5–12 V), Host mobile device, Fresnel lens, and Clear mirror as shown in Fig. 1. The speech is recognized and translated into text by the voice to text model running on a mobile device such as a smartphone. This data is sent to Node MCU device to communicate with the display device. The display device aligned with the Fresnel lens produces image reflections which display the text in front of the eye glass.



**Fig. 1** Smart glass prototype with OLED display and Fresnel lens

## 4 Methodology

The novelty aspect of the proposed smart glasses lies in incorporating both affordability and offline functionality. In contrast to high-priced current market alternatives, these smart glasses aim to provide a cost-effective solution without compromising functionality. Moreover, the integration of an offline model helps distinguish proposed glasses further and removes constraints like constant internet connectivity, making it suitable for users with limited internet access. This approach helps make the proposed model unique.

The approach used here involves utilizing a speech recognition AI model to transcribe live audio into text. The system operates on a NodeMCU (ESP32) platform and utilizes an OLED display to display text after adjusting focal length, and magnitude of image. Audio is split into chunks based on speech pauses and then passed through a voice-to-text model. This data is then sent to the NodeMCU via Bluetooth. The AR glasses feature an OLED display for text display which is passed through Fresnel lenses to adjust focal length and magnitude of the image.

The speech recognition model used is a stock Whisper model, an open-source ASR model fine-tuned for high-quality performance on specific distributions. This model was trained by OpenAI on 680,000 h of multilingual and multitask supervised data collected from the web, with approximately one-fifth (117,000 h) comprising non-English audio. It is a weakly supervised deep learning acoustic model built using an encoder-decoder transformer architecture. When fine-tuning for local dialects and languages, a value that is 40x smaller than what has been used for pre-training is an ideal learning rate, and linearly decay it to zero over the course of training. Speech is represented by a one-dimensional array that varies over time, each value representing the signal's amplitude. To process this, the continuous signal must be converted into a discrete form by sampling it at fixed steps (sampling rate). In order to avoid unexpected results, the sampling rate of the audio inputs should match that of the model, i.e., 16 kHz. All inputs must be of uniform length (30 s) by padding shorter samples with zeros at the end of the sequence and truncating longer ones to the required size. Unlike most similar models, an attention mask is not required when forwarding the audio inputs to the Whisper model. The processed audio is then

converted into log-Mel spectrograms which is the input expected by the Whisper model.

## 4.1 System Architecture

The implementation of AR glasses as a solution is built on the NodeMCU platform (ESP32), the approach involves splitting audio into chunks based on speech pauses and then passing through a speech recognition Model. Further details on the model are discussed below.

## 4.2 Speech Recognition Model

The OpenAI's Whisper [18] model is used for speech recognition; Google's speech recognition model may be used instead if running on mobile devices having tighter resource constraints as it only requires API calls to Google. The downside of using an API over an offline model is that it will struggle in areas with poor net coverage.

Whisper is an automatic speech recognition (ASR) model. Whisper models show a strong capacity to generalize across several datasets and domains without requiring fine-tuning, having been trained on 680 k hours of tagged data.

The scaling characteristics of Whisper trained on a range of models with varied sizes are investigated. Using FP16 with dynamic loss scaling and activation checkpointing, it was trained with data parallelism across accelerators.

The models underwent training using gradient norm clipping and AdamW optimization algorithm, with a linear decrease of the learning rate to zero after a warmup period of the first 2048 updates. The models undergo 220 training iterations, which is equivalent to two to three complete passes across the dataset. Each iteration uses a batch size of 256 segments. Gradient clipping involves limiting the size of the gradients during the optimization. This is an effective method to prevent the gradients from growing too large, which is needed to maintain numerical stability and ensure consistent convergence during training.

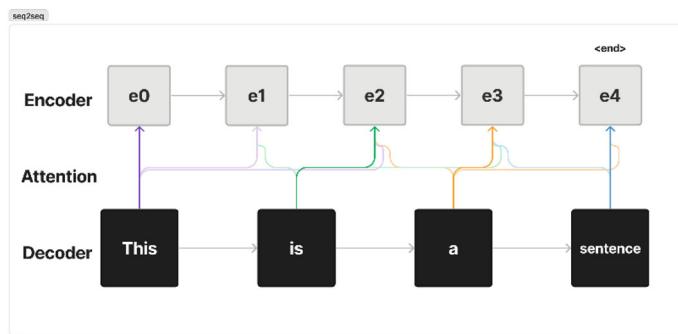
From Table 1, it may be inferred that whisper exhibits less word errors than wav2vec, another state-of-the-art speech recognition model, and achieves an average reduction in relative error of 55.2% when measured against both multilingual and purely English datasets. The Whisper model is inherently intended to be used on audio samples up to thirty seconds in length. Nonetheless, audio samples of any length can be transcribed using a chunking process. Thanks to the Transformers pipeline approach, this is achievable. When creating the pipeline, set `chunk_length_s = 30` for chunking. The pipeline can be used with batch inference if chunking is enabled. By passing `return_timestamps = True`, it may also be expanded to anticipate sequence-level timestamps.

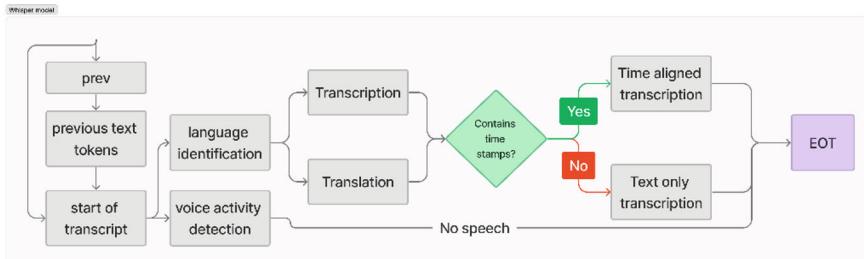
**Table 1** Relative error reduction

Dataset	wav2vec 2.0 (no LM) (%)	Whisper large V2 (%)	RER (relative error reduction)
Common voice	29.9	9	69.9
Tedlium	10.5	4	61.9
Artie	24.5	6.2	74.7
Fleurs En	14.6	4.4	69.9
CHIME6	65.8	25.5	61.2
CORAAL	35.6	16.2	54.5
VoxPopuli En	17.9	7.3	59.2
AMI IHM	37	16.9	54.3
CallHome	34.8	17.6	49.4
WSJ	7.7	3.9	49.4
Switchboard	28.3	13.8	51.2
LibriSpeech other	6.2	5.2	16.1
AMI SDM1	67.6	36.4	46.2
Average (%)	29.30	12.80	55.20

The model is a transformer-based encoder–decoder model, which is sometimes referred to as a sequence-to-sequence model. Sequence-to-sequence models are specifically intended to handle sequential data and are frequently employed in tasks related to natural language processing. The model consists of two primary components: an encoder and a decoder (Fig. 2).

The audio input is divided into segments and transformed into a log-Mel spectrogram, which is subsequently fed into an encoder. An encoder’s function is to receive an input sequence and transform it into a vector with a predetermined length. The decoder utilizes the context vector to generate the matching text, incorporating specific tokens that guide the model in executing various tasks such as identifying

**Fig. 2** Transformer model



**Fig. 3** Whisper model architecture

language, indicating phrase-level timestamps, transcribing speech, and translating speech into English (Fig. 3).

## 5 Results and Discussion

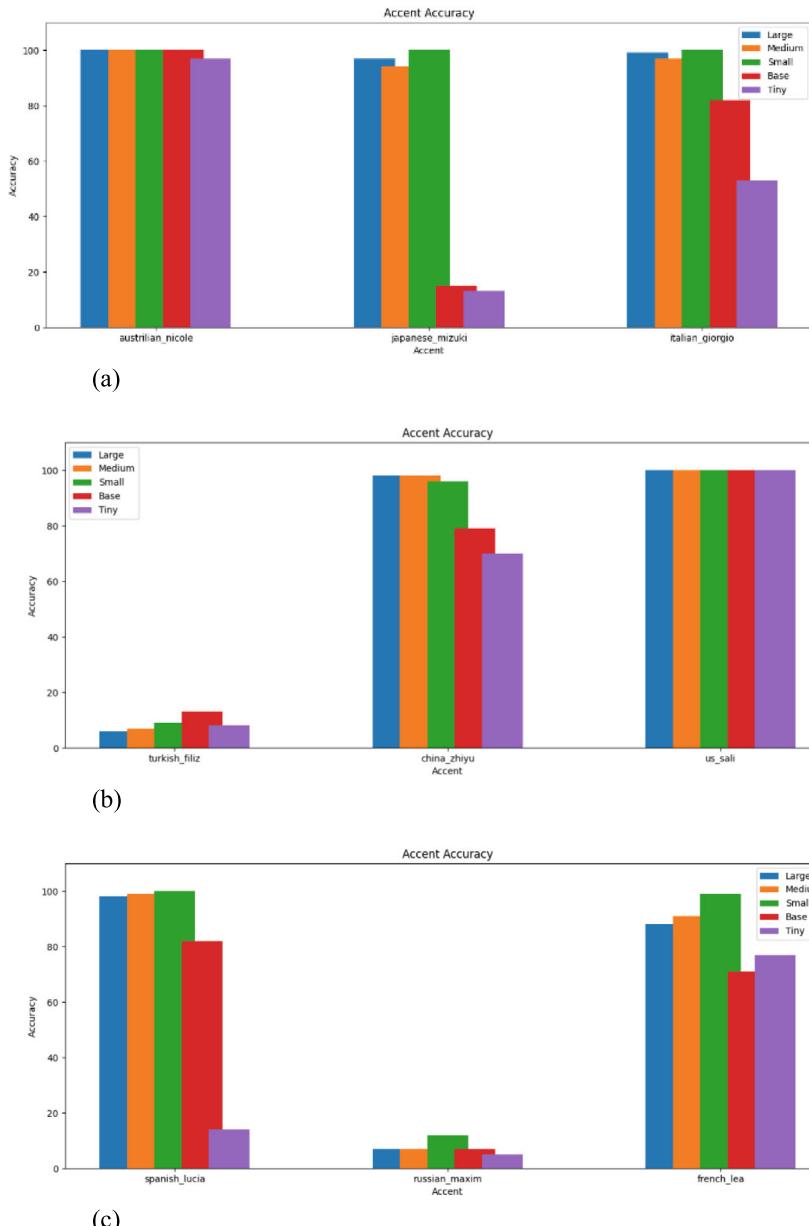
### 5.1 Speech-to-Text Conversion Performance

To test the performance of the Whisper model based on accent, the free text-to-speech converter available at [ttsmp3.com](http://ttsmp3.com) [19] is used to generate audio clips in various accents, these do not include any names in it. These audio samples are then used to test the Whisper model's ability to accurately recognize and process speech of the following accents:

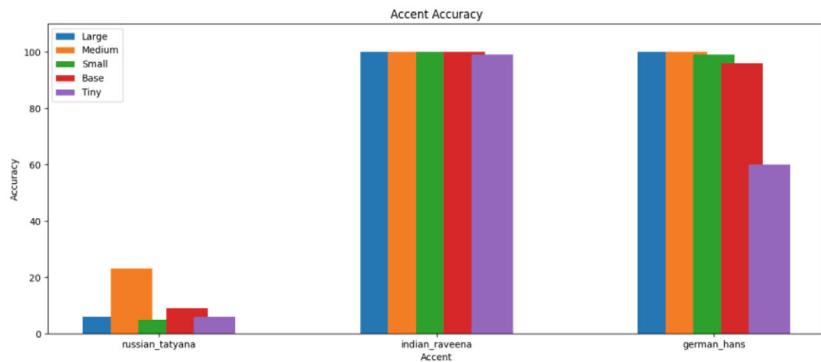
• Australian_Nicole	• Indian_Raveena
• Japanese_Mizuki	• German_Hans
• Italian_Giorgio	• Dutch_Ruben
• Turkish_Filiz	• Polish_Jan
• Chinese_Zhiyu	• British_English_Emma
• American_Sali	• Arabic_Zeina
• Spanish_Lucia	• Brazilian_Portuguese_Victoria
• Russian_Maxim	• Korean_Seoyeon
• French_Lea	• Potugese_Cristoano
• Russian_Tatyana	

The Whisper large, medium, small, base, and tiny models are tested against the above 19 accents and the results of these models are shown in Fig. 4a–g.

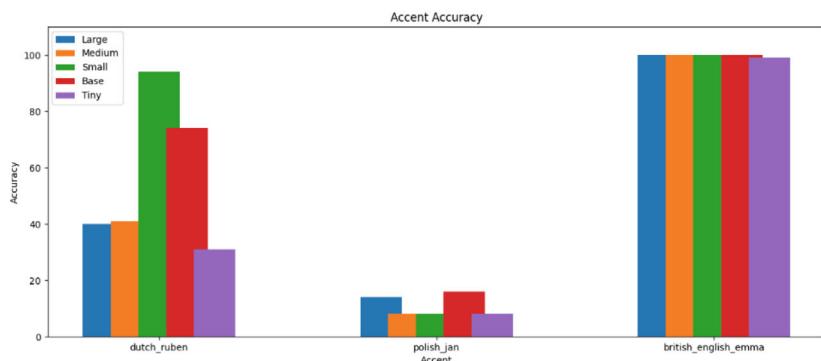
For this study case, Whisper small is the optimal option due to its size and is the model that Whisper recommends using for English. From Fig. 4a–g, the word error rate of the small model is derived (lower is better) (Table 2).



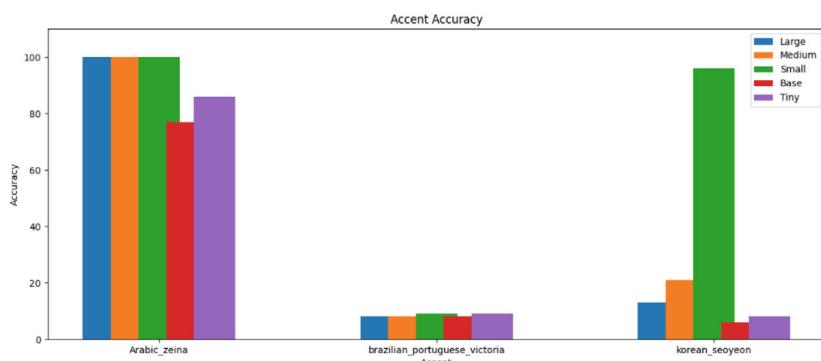
**Fig. 4** **a** Accuracy comparison for Australian\_Nicole, Japanese\_Mizuki, and Italian\_Giorgio. **b** Accuracy comparison for Turkish\_Filiz, Chinese\_Zhiyu, American\_Sali. **c** Accuracy comparison for Spanish\_Lucia, Russian\_Maxim, French\_Lea. **d** Accuracy comparison for Russian\_Tatyana, Indian\_Raveena, German\_Hans. **e** Accuracy comparison for Dutch\_Ruben, Polish\_Jan, British\_English\_Emma. **f** Accuracy comparison for Arabic\_Zeina, Brazilian\_Portuguese\_Victoria, Korean\_Seoyeon. **g** Accuracy comparison for Potugese\_Cristoano



(d)

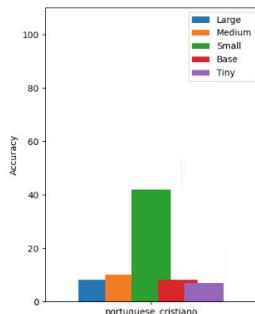


(e)



(f)

**Fig. 4** (continued)

**Fig. 4** (continued)

(g)

**Table 2** Whisper small word error rate

Accent	Word error rate	Accent	Word error rate
American	Low	Turkish	High
Italian	Low	Spanish	Low
Australian	Low	German	Low
British English	Low	Korean	Low
French	Low	Indian	Low
Polish	High	Arabic	Low
Brazilian Portuguese	High	Chinese	Low
Russian	High	Japanese	Low
Dutch	Low	Portuguese	Medium

## 5.2 Real-Time Responsiveness

The implementation demonstrated impressive real-time responsiveness averaging a transcription time of 2.6 s per 15 s of audio, ensuring that the transcribed text appears promptly on the AR glasses' display. The seamless coordination between the speech recognition model, Node MCU platform, and OLED display contributes to maintaining synchronization with ongoing conversations. This responsiveness enhances the user's ability to engage in dynamic interactions, enriching their communication experiences.

The user interactions with others using the above-developed smart glass device are tested for multiple scenarios to know the accuracy and reliability of the prototype. The results of audio translation using AI algorithms are printed in the laptop screen initially and then communicated to display unit through Node MCU automatically for the visibility of disabled persons view and provide their response. Figure 5 shows the examples of translated text and their displays in the computer screen and the main display unit.



**Fig. 5** Examples of audio-translated text displayed in computer and main display

### 5.3 Robustness, Limitations, and Usability in Real-World Environments

Whispers was trained on 680,000 h of diverse multilingual and multitask data leading to improved robustness to accents, background noise and technical language. Whisper excels in zero-shot scenarios indicating a higher degree of adaptability to unseen real-world situations. Although Whisper doesn't outperform models specifically tuned for clean speech benchmarks like LibriSpeech, it exhibits 50% fewer errors than those models in various diverse settings.

In spite of its strengths, extremely loud or disruptive noise might still pose a challenge. Speech that is mumbled, whispered, or heavily accented as well as niche or newly coined terms may not be recognized. Beyond speech recognition capabilities, initial feedback highlighted concerns about the glasses' battery life, weight, fit, and distribution of weight. The prototype had a primitive design and was quite bulky and uncomfortable for extended wear, affecting user experience. Users found the interface easy to learn and navigate and were able to interact with the UI with minimal instruction.

## 6 Conclusion

Augmented Reality (AR) glasses designed in this work suggest a major advancement in the reduction of barriers to communication and the improvement of accessibility in a variety of social circumstances. Although there is room for improvement given

the current accuracy rate of 78% for live speech-to-text conversion, the results highlight how technology has the power to fundamentally alter social interactions and communication dynamics.

The AR glasses developed is a promising solution to the long-standing problems experienced by people with hearing impairments since they integrate voice recognition AI models that are seamlessly synchronized with optical components. In summary, the AR glasses for real-time text conversion represent a step forward in enhancing communication accessibility. While the prototype demonstrated the feasibility of the approach, further research and development is necessary. More specifically, efforts should be made towards improving accuracy in noisy environments, extending battery life, and refining the design for comfort and usability. Despite its current limitations, this work displays the potential of integrating AI and AR technologies to empower individuals with hearing impairments and foster a more inclusive and equitable society.

**Acknowledgements** The authors would like to convey our gratitude to Amrita Vishwa Vidyapeetham and Chancellor Sri Mata Amritananda Mayi Devi, for providing us a chance to showcase our efforts towards these societal applications.

## References

1. Dalton DS, Cruickshanks KJ, Klein BE, Klein R, Wiley TL, Nondahl DM (2003) The impact of hearing loss on quality of life in older adults. *Gerontologist* 43(5):661–668. <https://doi.org/10.1093/geront/43.5.661>. PMID: 14570962
2. <https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss>
3. Rieder B (2020) Engines of order: a mechanology of algorithmic techniques. Amsterdam University Press, Amsterdam, Netherlands
4. Narendran P, Reddy V, Saju S, Suriya LU, Ravi Kumar Pandi V (2022) Smart home with condition monitoring. In: Lecture notes in networks and systems, vol 311, pp 653–667. [https://doi.org/10.1007/978-981-16-5529-6\\_50](https://doi.org/10.1007/978-981-16-5529-6_50)
5. Bhagylekshmi S, Jayasudha MA, Riju G, Mathew N, Ravikumar PV (2019) Automated railway track fault detection using solar powered electric vehicle. In: IEEE international conference on intelligent techniques in control, optimization and signal processing, INCOS 2019
6. Bhardwaj V, Rasamsetti Y, Valsan V (2022) Traffic control system for smart city using image processing. In: Piuri V, Shaw RN, Ghosh A, Islam R (eds) AI and IoT for smart city applications. studies in computational intelligence, vol 1002. Springer, Singapore. [https://doi.org/10.1007/978-981-16-7498-3\\_6](https://doi.org/10.1007/978-981-16-7498-3_6)
7. Balaji SA, Geetha P, Soman KP (2016) Change detection of forest vegetation using remote sensing and GIS techniques in Kalakkad Mundanthurai Tiger Reserve—(a case study). *Indian J Sci Technol* 9(30)
8. Srishilesh PS, Parameswaran L, Sanjay Tharagesh RS, Dr. Thangavel SK, Sridhar P (2019) Dynamic and chromatic analysis for fire detection and alarm raising using real-time video analysis. In: Proceedings of 3rd international conference on computational vision and bio inspired computing, Cham
9. Lee H, Hui P (2018) Interaction methods for smart glasses: a survey. *IEEE Access* 6:28712–28732. <https://doi.org/10.1109/ACCESS.2018.2831081>

10. Dabran I, Avny T, Singher E, Danan HB (2017) Augmented reality speech recognition for the hearing impaired. In: 2017 IEEE international conference on microwaves, antennas, communications and electronic systems (COMCAS), Tel-Aviv, Israel, pp 1–4. <https://doi.org/10.1109/COMCAS.2017.8244731>
11. Nikonorov A, Skidanov R, Fursov V, Petrov M, Bibikov S, Yuzifovich Y (2015) Fresnel lens imaging with post-capture image processing. IN: 2015 IEEE conference on computer vision and pattern recognition workshops (CVPRW), Boston, MA, USA, pp 33–41. <https://doi.org/10.1109/CVPRW.2015.7301373>
12. Lindberg DV, Lee HKH (2015) Optimization under constraints by applying an asymmetric entropy measure. *J Comput Graph Stat* 24(2):379–393. <https://doi.org/10.1080/10618600.2014.901225>
13. Pham T-T, Vu N-H, Shin S (2018) Design of curved Fresnel lens with high performance creating competitive price concentrator photovoltaic. *Energy Procedia* 144:16–32. <https://doi.org/10.1016/j.egypro.2018.06.004>
14. Mirzaei R, Ghorshi S, Mortazavi M (2012) Combining augmented reality and speech technologies to help deaf and hard of hearing people. In: 2012 14th symposium on virtual and augmented reality, Rio de Janeiro, Brazil, pp 174–181
15. Saranya G, Tharun SV, Tamilvizhi T, Surendran R (2023) Intelligent wearable device for visually impaired person to act as a third eye. In: 4th international conference on electronics and sustainable communication systems, ICESC 2023—Proceedings, pp 889–895. <https://doi.org/10.1109/ICESC57686.2023.10193619>
16. Nayana BR, Raju NRY, Velidi AS (2023) Low-cost smart glasses for people with visual impairments. In: IEEE international conference on advances in electronics, communication, computing and intelligent information systems, ICAECIS 2023—Proceedings, pp. 173–176. <https://doi.org/10.1109/ICAECIS58353.2023.10170565>
17. Choudhary S, Dhote N, Deshpande AA, Sambhariya A, Joshi PK (2023) IoT based smart glasses with facial recognition for people with visual impairments. *SSRG Int J Electron Commun Eng* 10(9):154–159. <https://doi.org/10.14445/23488379/IJEEE-V10I9P114>
18. Radford A, Kim JW, Xu T, Brockman G, McLeavey C, Sutskever I (2023) Robust speech recognition via large-scale weak supervision. In: Proceedings of the 40th international conference on machine learning ICML'23, pp 28492–28518
19. Free Text-to-Speech for 28+ languages & MP3 Download | ttsMP3.com (n.d.). <https://ttsmp3.com>

# Graph-Based Predictive Modeling in Drug Response



T. P. Athulya Valsan and Anuraj Mohan

**Abstract** To customize treatment based on an individual patient's unique characteristics, personalized medicine relies on predicting drug response. Such a method maximizes benefits and minimizes side effects. It simplifies choosing which therapy to use by reducing trial-and-error approaches and saves time as well. This also effectively improves patient outcomes while saving costs through needless medical treatment avoidance. This study focuses on the performance in terms of evaluation metrics such as training loss, and F1-score of different graph-based models on drug response prediction when they are combined with different feature extraction methods. In this study, five different graph models are used which are the Graph Convolutional Network, Graph Attention Network, GraphSAGE model, Graph Isomorphism Network, and Graph Transformers, and two feature extraction methods, CNN and LSTM. The investigation ultimately predicts response values for all drug-cell line pairs using a densely connected neural network and compares the performance of different graph models.

**Keywords** SMILES · GCN · GAT · GIN · GraphSAGE · GCN Transformer · GDSC

## 1 Introduction

The fast-growing discipline of precision medicine seeks to tailor medical care to each patient's unique needs. Accurately anticipating drug reactions is essential for optimizing therapeutic efficacy, minimizing side effects, and enhancing overall treatment outcomes. This is particularly crucial in fields like oncology and chronic sickness,

---

T. P. Athulya Valsan (✉) · A. Mohan

Department of Computer Science and Engineering, NSS College of Engineering, Palakkad, Kerala, India

e-mail: [athulyavalsantp@gmail.com](mailto:athulyavalsantp@gmail.com)

A. Mohan

e-mail: [anurajmohan@nssce.ac.in](mailto:anurajmohan@nssce.ac.in)

where different patient populations may not respond well to typical treatments. Precision medicine is being developed by the integration of algorithms that predict response to medication on a graph. Because people with cancer are so different from one another, it is challenging to predict how specific medicines would impact any given patient. Because of the complex interplay between biological components and the dynamic character of cancer, novel techniques to exact predictive modeling are needed. The majority of prediction models in use today rely on traditional methods of data extraction, like chemical fingerprinting and synthetic features, which fall short in capturing the complex molecular structures of pharmaceuticals in the context of evolving drug-cancer interactions. Model and feature extraction strategies need to be improved in order to get over the challenges associated with anticipating the response to cancer medicines. To address uncertainties, patient variability, and molecular complexities, a comprehensive strategy including the latest machine learning algorithms, using larger datasets, and exploring new feature extraction techniques is required. As a result, cancer patients will receive more precise and personalized drug response estimates, which will enable more effective and personalized treatment plans.

This work aims at advancing patient-centered and personalized health care through an improved understanding of drug response mechanisms and the creation of new applications in this field. Five different types of graph convolutional networks were used in this study including Graph Convolutional Network, Graph Attention Network, GraphSAGE model, Graph Isomorphism Network, and Graph Transformers and to describe drug features CNN and LSTM are also used. The performance of different graph models when combined with these feature extraction methods is analyzed with the help of different evaluation matrices to get a better understanding about which model is better for drug response prediction.

## 2 Related Works

There have been several creative methods put forth for forecasting medication reactions in cancer treatments. Using information from drug and cell line similarity networks (DSN and CSN), one such model [1] combines cell line-drug networks. With a Pearson correlation coefficient of 0.6 for most drugs, this model outperforms earlier research in terms of prediction ability. Interestingly, it uses pharmacogenomics information from the Cancer Genome Project (CGP) and the Cancer Cell Line Encyclopedia (CCLE) to predict sensitivity to MEK1/2 inhibitors in BRAF mutant cell lines with remarkable accuracy. Strong links in medication combinations are emphasized, as is the impact of chemical and genetic structures on drug sensitivity. Additionally, the study assesses the resilience of the model using cross-validation, which outperforms individual models and the elastic net model, particularly for medications targeting specific pathways. An alternative method, called HNMDRP [2], predicts medication responses for individual patients using a heterogeneous network-based paradigm. HNMDRP acknowledges the computational complexity of the model and

the crucial importance of target gene nodes and achieves promising results in predicting cell line-drug associations by incorporating multiple sources of data, such as chemical structures of drugs, drug-target interactions, protein-protein interactions, and cell line gene expression profiles.

Moreover, pretreatment baseline tumor gene expression data is used in a strategy focused on chemotherapy-related adverse effects [3] to forecast patient responses. While more accurate data and prospective testing are required, this strategy shows promise for influencing medication development and customizing treatment plans. Similar to this, a preclinical prediction model [4] predicts drug responses in several cell lines at once using support vector machines (SVM) and recursive feature selection. This methodology has the potential to revolutionize medication selection processes by properly predicting drug reactions through the utilization of genetic characteristics.

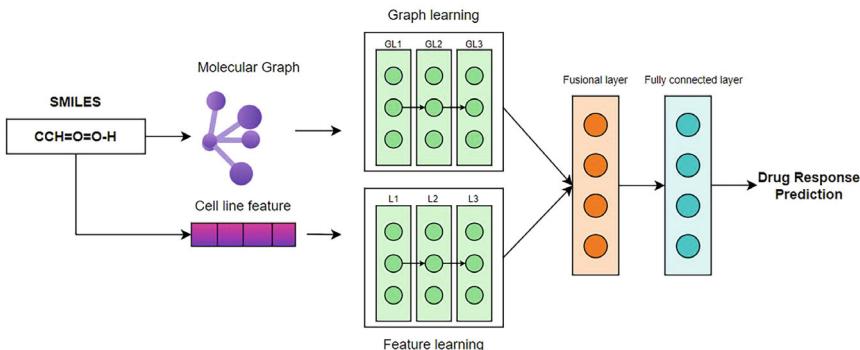
A multimodal attention-based convolutional encoder architecture for interpretable anti-cancer drug sensitivity prediction is presented in a different paper [5]. By taking into account different intracellular interactions and tumor gene expression levels, this model performs better than baseline methods. Additionally, a model called tCNNS [6] addresses problems with disparate representations of drug structures in SMILES format and enables phenotypic screening between cancer cell lines and anti-cancer medications. Furthermore, employing huge drug-induced gene expression datasets and cytotoxicity data, Ensemble Learning for Drug Activity Prediction (ELDAP) [7] uses ensemble learning techniques to predict drug reactions, demonstrating accuracy and versatility in managing a variety of input variables.

Combining drug chemical structures with multi-omics data, DeepCDR [8]—a hybrid graph convolutional network—predicts the response to chemotherapy drugs. It shows great predictive power and translational potential. In comparison to earlier models, DrugGCN [9] exhibits higher prediction accuracy by utilizing biological networks and gene expression data to forecast drug reactions. Additionally, DualGCN [10] solves the difficulties in getting trustworthy single-nucleotide variation (SNV) data and shows strong prediction capacity across a range of medication classes and cancer types. By using graph convolutional networks (GCNs) to predict drug-cell line responses, GraphDRP [11] outperforms previous techniques and enhances interpretability using saliency maps. Finally, by utilizing graph transformer techniques and combining different omics data, GraTransDRP [12] enhances drug representation and prediction accuracy.

Even though multiple works of a similar manner have been conducted in previous years, there is a need of study to compare all the possible graph models and how they perform differently when combined with different feature extraction methods.

### 3 Proposed Method

The model structure (Fig. 1) combines cell line features and SMILES graph branch. Three layers of learning representation in SMILES graph consisted of hierarchical



**Fig. 1** An illustration of the experimental configuration. Every cell line was transformed into a vector with 735 dimensions in a one-hot format. The drug was translated to graph format in the SMILE string. Subsequently, the drug's characteristic was learned using graph-based models. To transform the result to 128 dimensions after the graph neural network the fully connected layer was also utilized. To anticipate the response, these two representations were concatenated and passed through two FC layers

feature extraction from molecular graphs. Non-linearity is added to the model by activation function (rectified linear unit). Every graph uses global max pooling to combine node-level information into one representation across batch dimensions. There are three 1-dimensional convolutional layers or LSTM with cell line feature branch followed by max pooling layers. ReLU activation functions introduce non-linearity after each convolutional layer. Global maximum pooling through the batch dimension merges node-level information into a single representation for each graph. Cell line feature branch comprises three 1-Dimensional Convolutional Layers (Conv1D) or LSTM followed by Max Pooling layers. Non-linearity is introduced using a ReLU activation function after every layer. The output is flattened and transmitted into a fully connected layer, concatenated along the feature dimension. Two fully linked layers with dropout regularization and ReLU activation functions are applied to the concatenated features. The model's predictions are produced by a sigmoid activation function, minimizing overfitting. The model efficiently incorporates information from both graph and sequential data sources to develop representations that capture the underlying relationships between medications and cell lines.

### 3.1 Graph Learning

Mainly five different kinds of graph models are used in this study, and the three layers of learning representation in SMILES graph consisted of hierarchical feature extraction from molecular graphs. The graph models are as follows.

### 3.1.1 Graph Convolutional Network (GCN)

Formally, two matrices were used to hold a graph describing a particular drug,  $G = (V, E)$ : a feature matrix  $X$  and an adjacency matrix  $A$ . The feature matrix  $X$  contains the information on  $N$  nodes in the graph, which has dimensions  $R^{N \times F}$ . An  $F$ -dimensional vector represents each node in the network. The adjacency matrix  $A$ , which has dimensions  $R^{N \times N}$ , shows the relationships between the graph's nodes. The graph convolutional layer uses these two matrices as input and aims to provide a node-level output with  $C$  characteristics for each node, which is represented as

$$AXW \quad (1)$$

where  $W$  is a dimension-wise trainable parameter matrix,  $R^{F \times C}$ . This strategy did, however, have two serious shortcomings. First, it excluded the node itself and combined the feature vectors of every node that was nearby. Second, when matrix  $A$  was multiplied, the feature vector's scale changed since normalization was absent from it. The GCN model [11] was developed to address these problems, and it entailed normalizing and appending an identity matrix to  $A$ . It was also discovered that symmetric normalization produced better results. The equation

$$\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} X W \quad (2)$$

The GCN layer uses these improvements by using  $\tilde{A}$  as the graph adjacency matrix with an extra self-loop and  $\tilde{D}$  as the graph diagonal degree matrix. Three successive GCN layers with a Rectified Linear Unit (ReLU) function after each layer comprised the GCN-based model. After the last GCN layer, a global max pooling layer was added to learn the representation vector of the entire graph. This vector was then merged with the cell line representation to forecast the response value.

### 3.1.2 Graph Attention Network (GAT)

The GAT model's architecture [11] consists in stacking multiple attention layers on top of each other. Each GAT layer receives a node feature vector  $x$  as input and applies a linear transformation to each node using weight matrix  $W$ . The importance of node  $j$  to node  $i$  is then indicated by the attention coefficients, which are then calculated for each pair of nodes connected by an edge and represented as

$$a(Wx_i, Wx_j) \quad (3)$$

After normalizing these coefficients using a soft-max function, each node's output features are calculated using

$$\sigma \left( \sum_{j \in N(i)} \alpha_{ij} W x_j \right) \quad (4)$$

where  $\sigma(\cdot)$  stands for a non-linear activation function and  $\alpha_{ij}$  for the normalized attention coefficients. To generate the graph representation vector, this model combines two GAT layers which are activated by a Rectified Linear Unit (ReLU) function. Next, a global max pooling layer is also added. In particular, multi-head-attentions with ten heads were used for the first GAT layer to make sure that the number of input features and output features are matched. In a similar vein, 128 output features are assigned to the second GAT layer in accordance with the cell line representation vector.

### 3.1.3 Graph Isomorphism Network (GIN)

Among Graph Neural Networks (GNNs), the Graph Isomorphism Network (GIN) [11] is well known for its remarkable discriminative ability. To be more precise, the node feature update procedure uses a multi-layer perceptron (*MLP*) defined as

$$MLP \left( (1 + \mu) x_i + \sum_{j \in N(i)} x_j \right) \quad (5)$$

where  $x$  is the node feature vector,  $N(i)$  is the set of nodes that are neighboring node  $i$ , and  $\mu$  is either a constant scalar or a trainable parameter. The GIN architecture consists in stacking five (32 features each) GIN layers in this case. For each layer, batch normalization layers were added to improve learning capabilities. ReLU activation functions were then used to learn non-linear mapping functions. A full network representation vector was aggregated by adding a global max pooling layer, similar to GAT architectures.

### 3.1.4 GraphSAGE

GraphSAGE presents an approach to inductive representation learning on large-scale graph structures that aims at generating compact vector representations of nodes which is especially useful for networks with many node attribute data. The resulting low-dimensional vector embeddings of nodes are highly useful for a wide range of machine learning applications, such as link prediction, clustering, and node categorization. Conventional embedding frameworks usually work in a transductive fashion, which restricts their ability to produce embeddings for a single fixed graph. As a result, these transductive techniques have difficulty generalizing to different graphs and adjusting to new nodes, including those that arise in dynamic graph settings. GraphSAGE, on the other hand, functions as an inductive framework that

effectively creates representations even for data instances that have never been seen before by using node attribute data.

### 3.1.5 Graph Transformer

The Graph Transformer [12] improves feature extraction from a generalized drug graph by adding missing links. To get beyond the restrictions of Recurrent Neural Networks (RNNs), the Graph Transformer leverages Transformer techniques used in natural language processing to include neighborhood nodes in the process of extracting graph attributes. An adjacency tensor records relationships in a heterogeneous graph  $G = (V, E)$  with node and edge types, while a feature matrix depicts nodes as  $F$ -dimensional vectors:

$$A_P = A_{t1} \dots A_{tP} \quad (6)$$

A meta-path uses adjacency matrices for each type of edge to forecast new connections between nodes. By using  $1 \times 1$  convolution to create soft adjacency matrices, a convex combination of new meta-paths is established. As stated in

$$Z = \prod_{i=1}^C \sigma \left( D_i^{-\frac{1}{2}} A_i^{(l)} X W \right) \quad (7)$$

these together with Graph Convolution networks create node representations. Like natural language processing attention processes, the equation uses neighborhood connections to improve prediction accuracy. To overcome the difficulty in locating nodes in feature extraction because of graph attributes, Graph Transformer precomputes each and every graph in the dataset using Laplacian eigenvectors, which are calculated as

$$\Delta = I - D^{-1/2} A D^{-1/2} = U^T \Lambda U \quad (8)$$

## 3.2 Feature Learning

In terms of the Sequential Feature Extraction Branch, two different types of methods are used for the cell line feature branch: LSTM and three 1-dimensional convolutional layers.

### 3.2.1 LSTM

A type of recurrent neural network (RNN) called Long Short-Term Memory (LSTM) was created to solve the vanishing gradient problem in long-term training. Sequential data applications such as speech recognition, time series prediction, and language modeling benefit greatly from it. Cell state, previous hidden state, and input vectors

are passed into LSTM cells, which then carry out processes including removing extraneous data, adding new data, and generating output. At each time step, the LSTM produces an output tensor containing hidden state activations after the protein input data has been bent to fit its needs. The output gate determines how much of the cell state to reveal, while the input, forget, and output gates regulate the information flow.

### 3.2.2 1D CNN

One-dimensional convolutional neural networks (1D CNNs) in three successive layers are used by the cell line feature branch to find patterns in cell line data. For non-linearity, an activated rectified linear unit (ReLU) function is implemented after every convolutional layer. Max pooling layers are used to downsample the feature maps in an efficient manner while preventing overfitting. Following feature retrieval, fully linked layers are created using the features to create comprehensive representations for drug response analysis and prediction.

## 3.3 *Fusion Layer and FC Layer*

The model captures comprehensive information from many sources by concatenating derived features from SMILES graph branches and cell line data. Fully connected layers receive the concatenated feature vector, enabling complex patterns and non-linear alterations. To lessen overfitting and increase non-linearity, dropout regularization and ReLU activation functions are employed. The output layer generates a final prediction for the drug response prediction task, using a sigmoid activation function to make sure the predicted values are within the range of [0, 1].

## 3.4 *Loss Function and Learning*

Different evaluation metrics are used to analyze the performance of these models.

### 3.4.1 Train and Validation Loss

In order to evaluate model performance, identify optimal parameter values, and minimize training and validation losses for efficient model development, it is important to train machine learning models with low training and validation losses. An equation that is commonly used to represent loss functions is the mean squared error (MSE):

$$L = \frac{1}{N} \sum_{i=1}^N \left( y_{true}^{(i)} - y_{pred}^{(i)} \right)^2 \quad (9)$$

where  $N$  is the number of samples in the dataset.

In the context of machine learning and regression problems, the Pearson score evaluates the linear correlation between the predicted values ( $y_{pred}$ ) generated by a model and the true target values ( $y_{true}$ ). The Pearson correlation coefficient  $r$  can be calculated as follows:

$$r = \frac{\sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^N (x_i - \bar{x})^2 \sum_{i=1}^N (y_i - \bar{y})^2}} \quad (10)$$

Other performance metrics include recall, which measures the model's ability to detect positive instances; accuracy, which evaluates the proportion of correct predictions; precision, which assesses the model's accuracy in predicting positive outcomes; and the F1-score, which balances precision and recall, especially useful for imbalanced datasets.

## 4 Experimental Setup

### 4.1 Dataset

GDSC database ([www.cancerRxgene.org](http://www.cancerRxgene.org)) is a large-scale program that has created extensive datasets for anti-cancer drugs in many cell lines [13]. The -omics dataset gives knowledge about gene expression, which can disclose genetic abnormalities and show how genes work in different biological settings. Drug response measures how well drugs prevent cancer cell growth—it uses IC50 or AUC values. GDSC is the primary source for cell line drug sensitivity data, having evaluated 250 medications across 1,074 cell lines. The benchmark dataset, GDSC version 6.0, identified 948 cell lines and 223 drugs. Of the total number of drug-cell line pairs analyzed, 172,114 (81.4%) had response values available, while 18.6% had response values missing. Response levels for the IC50 ranged from 0 to 1, and drugs were encoded using the SMILES standard format. The GDSC version 6.0 is the primary source for cell line drug sensitivity data.

### 4.2 Experimental Design

In this work, the prediction efficacy of drug-cell line interaction models is assessed using known couples. Of the 211,404 potential drug-cell line combinations, responses

**Table 1** Parameter settings

Parameter	Value(s)
Learning rate	1e-4
Epochs	100
Dropout	0.5
Batch size	Train: 512, Val: 512, Test: 512

for 172,114 pairings are included in the GDSC data. The data was shuffled before division to decrease overfitting and enhance model generalization. Following that, 10% were designated for testing, 10% for validation, and 80% for training out of all known pairs. The model's performance during training was evaluated using the testing set, and its hyperparameters were adjusted using the validation set (Table 1).

Predicting drug-cell line pairings with missing responses is the aim of this effort. Using a pre-trained model from a mixed test trial, predictions were created for absent couples in the GDSC dataset. The primary objective was to assess the expected performance for drugs that had never been observed before. To do this, drugs were simultaneously taken out of the testing and training sets. Out of the 223 medicines, 90% (or 201 medications) had their IC50 values randomly assigned for training, with 80% going toward the training set and 10% going toward validation. The remaining 10%, or 22 medications, made up the testing set. This setting allows one to evaluate the prediction power of the model for drugs that were not employed during training or testing. The models were trained for 100 epochs with a learning rate 1e-4 and a dropout of 0.5.

While considering CNN models, just like between architectures, consistency of learning may be indicated by convergence of training loss too. It is interesting to note that in several metrics (especially F1-score and precision) CNN-based GCN-Transformer model outperforms the one based on LSTM. This means that graph structures may have better prediction ability with more complicated features being extracted by design of CNNs from them. All Pearson scores are good for any CNN model since it measures correlation between expected and actual values which shows that they capture complex patterns within data very well indeed. But there were some intriguing results when comparing those achieved by various LSTM-based ones against those obtained via their corresponding CNN alternatives. While working with the graph data, LSTM models are good for recognizing temporal dependencies; however, those structures like GCN-Transformer which are based on CNN can best be used to discover complicated attributes leading to higher accuracy of prediction. The outcomes must not be taken as universal due to the contextual nature of these findings since different types of datasets and specific tasks performed may have varying effectiveness levels for each architecture (Tables 2 and 3).

In conclusion, it can be stated that the LSTM-based models outperform the competition in terms of temporal linkages inside graph topologies and have strong learning capabilities. The GCN-Transformer and other CNN-based methods, on the other

**Table 2** Results with LSTM

Model	Train loss	Validation loss	Pearson score	F1-score	Recall	Accuracy	Precision
GCN [11]	0.0013	0.0012	0.8508	0.9386	0.9667	0.899	0.9121
GAT [11]	0.0015	0.0013	0.8439	0.9381	0.9504	0.9002	0.9261
GIN [11]	0.0012	0.0012	0.8509	0.9383	0.9635	0.8987	0.9144
GraphSAGE	0.0013	0.0012	0.8519	0.9373	0.9546	0.8986	0.9207
GCN-Transformer [12]	0.0013	0.0013	0.8449	0.9346	0.9607	0.893	0.9099

**Table 3** Results with CNN

Model	Train loss	Validation loss	Pearson score	F1-score	Recall	Accuracy	Precision
GCN [11]	0.001	0.0009	0.8959	0.9482	0.9564	0.9163	0.9401
GAT [11]	0.001	0.001	0.8862	0.9481	0.975	0.9146	0.9228
GIN [11]	0.001	0.001	0.8987	0.9499	0.9756	0.9176	0.9256
GraphSAGE	0.001	0.001	0.8837	0.9453	0.9681	0.9103	0.9236
GCN-Transformer [12]	0.0008	0.0008	0.9113	0.9524	0.9669	0.9228	0.9384

hand, provide higher anticipated accuracy, which makes sense considering their ability to recognize finer details. The job specifics should be considered when choosing between CNN and LSTM architectures. Temporal dependencies and feature extraction capabilities should be carefully evaluated in order to maximize model performance.

## 5 Conclusion

This study investigated the predictive power of graph-based models for drug-cell line reactions. In terms of training loss, CNN models gradually converged, with the GCN-Transformer having the lowest loss. On the other hand, GCN and GraphSAGE showed competitive performance, indicating that they can accurately predict underlying graph structures. CNN-based GCN-Transformer surpassed LSTM models when they were transformed into CNN designs in terms of F1-score and precision among others. This implies that graphs can be made simpler by CNNs as well as improving prediction accuracy. Prediction accuracy is higher for CNN architectures such as GCN-Transformer than LSTM models because it recognizes more complicated features. What this means is that the predictions are better with those designs based

on expert feature extraction skills of a CNN like the GCN Transformer compared to an LSTM which is good at capturing temporal correlations within graph topologies.

## 6 Future Scope

The next steps of the research include interpretability optimization, multimodal information incorporation, transfer learning investigation, and model architecture improvement for graph-based data. Follow-up actions involve clinical setting validation, testing models on different datasets and addressing ethical considerations toward fair predictions. These efforts should lead to drug discovery and personalized medicine leading to more effective treatments against various illnesses.

## References

1. Zhang N et al (2015) Predicting anticancer drug responses using a dual-layer integrated cell line-drug network model. *PLoS Comput Biol* 11(9):e1004498
2. Zhang F et al (2018) A novel heterogeneous network-based method for drug response prediction in cancer cell lines. *Sci Rep* 8(1):3355
3. Geeleher P, Cox NJ, Stephanie Huang R (2014) Clinical drug response can be predicted using baseline gene expression levels and in vitro drug sensitivity in cell lines. *Genome Biol* 15:1–12
4. Dong Z et al (2015) Anticancer drug sensitivity prediction in cell lines from baseline gene expression through recursive feature selection. *BMC Cancer* 15(1):1–12
5. Manica M et al (2019) Toward explainable anticancer compound sensitivity prediction via multimodal attention-based convolutional encoders. *Mol Pharm* 16(12):4797–4806
6. Liu P et al (2019) Improving prediction of phenotypic drug response on cancer cell lines using deep convolutional network. *BMC Bioinform* 20(1):1–14
7. Tan M et al (2019) Drug response prediction by ensemble learning and drug-induced gene expression signatures. *Genomics* 111(5):1078–1088
8. Liu Q et al (2020) DeepCDR: a hybrid graph convolutional network for predicting cancer drug response. *Bioinformatics* 36(Supplement\_2):i911–i918
9. Kim S et al (2021) Graph convolutional network for drug response prediction using gene expression data. *Mathematics* 9(7):772
10. Ma T et al (2022) DualGCN: a dual graph convolutional network model to predict cancer drug response. *BMC Bioinform*. 23(4):1–13
11. Nguyen T et al (2021) Graph convolutional networks for drug response prediction. *IEEE/ACM Trans Comput Biol Bioinform* 19(1):146–154
12. Chu T et al (2022) Graph transformer for drug response prediction. *IEEE/ACM Trans Comput Biol Bioinform* 20(2):1065–1072
13. Yang W et al (2012) Genomics of Drug Sensitivity in Cancer (GDSC): a resource for therapeutic biomarker discovery in cancer cells. *Nucl Acids Res* 41(D1):D955–D961

# Leveraging MaxEnt and TF-IDF Trigrams Against Fake News



S. Siji Rani, Gade Sai Panshul, Tathipamula Harini Sai,  
Lingutla Prem Kumar, and Hareendra Sri Nag Nerusu

**Abstract** In an era dominated by the rampant dissemination of misinformation, the paper's primary objective is to develop a robust system capable of accurately detecting fake news, thereby enhancing the credibility and reliability of information. This paper compares the abilities of SVM classifier, Naive Bayes, and MaxEnt classifiers along with feature engineering techniques such as part-of-speech (POS) tagging, term frequency-inverse document frequency (TF-IDF), and trigram models to achieve higher accuracy levels. Three primary objectives underscore this research: Unlike conventional methods reliant on sentiment analysis and fact-checking this paper aims in the development of a sophisticated text classification model leveraging advanced NLP techniques and neural networks, the systematic evaluation of baseline models such as Margin Maximizing Classifier, Naïve Bayes, and Maximum Entropy (MaxEnt) Classifier, and the implementation of feature engineering techniques to improve model accuracies through refined linguistic analysis. Through meticulous data analysis and comprehensive methodology, this research aims to contribute to the advancement of fake news detection strategies, thereby fostering informed and united communities. The research showcased the effectiveness of the MaxEnt model combined with TF-IDF and trigram features in detecting fake news, achieving a remarkable accuracy of 0.95. This superior performance underscores the model's ability to capture complex linguistic patterns and word associations through a synergistic approach. The results also provide a foundation for future studies to integrate

---

S. Siji Rani (✉) · G. S. Panshul · T. H. Sai · L. P. Kumar · H. S. N. Nerusu  
Department of Computer Science and Engineering, Amrita School of Computing, Amrita Vishwa Vidyapeetham, Amritapuri, India  
e-mail: [sijiranis@am.amrita.edu](mailto:sijiranis@am.amrita.edu)

G. S. Panshul  
e-mail: [amenu4cse20324@am.students.amrita.edu](mailto:amenu4cse20324@am.students.amrita.edu)

T. H. Sai  
e-mail: [amenu4cse20369@am.students.amrita.edu](mailto:amenu4cse20369@am.students.amrita.edu)

L. P. Kumar  
e-mail: [amenu4cse20340@am.students.amrita.edu](mailto:amenu4cse20340@am.students.amrita.edu)

H. S. N. Nerusu  
e-mail: [amenu4cse20330@am.students.amrita.edu](mailto:amenu4cse20330@am.students.amrita.edu)

multimodal data sources and explainable AI, promoting transparency and trust in deployed fake news detection systems.

**Keywords** Fake news detection · POS tagging · Max Ent · Feature engineering · TF-IDF trigrams · Bigram count vectorizer

## 1 Introduction

In the current era of information proliferation, the prevalence of fake news presents a formidable societal challenge, eroding trust in media and disrupting public discourse. This research paper stands out by delineating distinctive research objectives aimed at enhancing the accuracy and effectiveness of fake news detection methodologies. Through a meticulous examination of the evolving landscape of misinformation, this paper sets out to address the critical issue of detecting fake news by leveraging advanced techniques in text vectorization, natural language processing (NLP), and machine learning.

Motivated by the escalating threat of misinformation and its far-reaching consequences on public opinion and decision-making processes, this research paper embarks on a journey to develop innovative approaches to combat fake news. Unlike conventional methods that primarily rely on sentiment analysis, pattern recognition, and fact-checking, this paper adopts a novel approach centered around the development of a detailed text classification model. This model, underpinned by advanced NLP techniques and neural networks, aims to achieve heightened accuracy in discerning fake news from genuine information.

The research paper emphasizes three key unique research objectives, each poised to revolutionize the landscape of fake news detection which includes the Development of a Detailed Text Classification Model, Exploration and Evaluation of Baseline Model, and Feature Engineering for Improved Accuracies.

The paper is organized to methodically display the inquiry about destinations, related work, dataset, proposed framework plan, and calculations utilized, and comes about accomplished. After clearly depicting the special inquiry about objectives within the presentation, an audit of significant earlier works is given to contextualize the study's commitments. The dataset segment diagrams the curation handle and composition of the dataset utilized for show preparation and assessment. The framework plan area illustrates the comprehensive technique, enveloping information preprocessing, vectorization, and pattern modeling, including building, show choice, and training/evaluation stages. Particular calculations like Gullible Bayes, Back Vector Machines, and Greatest Entropy are at that point nitty gritty, shedding light on their individual parts in fake news discovery. At last, the comes about discourse segment presents an examination of the exploratory results, highlighting the execution measurements of different demonstrate setups. This consistent movement adjusts the document's components, directing the peruser through the investigative handle underscoring the study's importance in progressing fake news discovery procedures.

## 2 Related Works

### 2.1 *Detection of Fake News Spreaders*

One approach introduces a graph neural network-based framework in [1] leveraging trust-based strategies and bot filtration, achieving superior performance in identifying fake news spreaders. However, it primarily focuses on the spread of news rather than analyzing content, which limits its comprehensiveness in detecting misinformation [2]. To address this limitation, our research incorporates advanced content analysis techniques into the detection framework, enabling a more holistic approach to identifying fake news spreaders [3]. Specifically, we utilize natural language processing (NLP) algorithms to analyze the linguistic features of news articles, thereby enhancing our ability to identify deceptive content.

### 2.2 *Propagation and Impact of Fake News*

Studies have investigated the propagation of fake news on social media platforms [4] and its detrimental impact on public perception and reputation. Various methodologies, including the development of prediction models using classification methods such as Probabilistic Classifier, Bayesian Network, and J48 with F-score of 69.7%, have been employed. While achieving promising results, limitations in data mining, particularly concerning dataset quality and biases, present challenges [5]. Our research addresses these limitations by implementing robust data collection and preprocessing techniques, ensuring the reliability and representativeness of training data. Specifically, we employ stratified sampling techniques to ensure that our dataset captures the diversity of fake news content circulating online.

### 2.3 *Modeling Techniques for Fake News Detection*

The SA-HyperGAT model [6] introduces an innovative approach to detecting fake news by capturing high-order dependencies between words and sentences. However, its reliance on user comments for sentiment-aware hypergraph construction poses challenges, particularly when comment data is sparse [7]. To enhance its effectiveness, our research explores alternative sources of data and refines the model architecture to reduce reliance on user-generated content, incorporating sentiment analysis from multiple sources. Specifically, we integrate sentiment analysis from news articles, social media posts, and expert opinions to construct a comprehensive understanding of the emotional context surrounding fake news.

## 2.4 Addressing Challenges in Fake News Dissemination

Studies [8] have explored the detrimental effects of widespread fake news dissemination and underscored the necessity for advanced algorithms. However, our proposed system design focuses on enhancing fake news detection through advanced NLP and machine learning techniques. While studies have highlighted the need for advanced algorithms, limitations in examining propagation patterns and integrating user feedback remain; our methodology lays only the groundwork for future advancements in these areas. By combining data preprocessing, vectorization, baseline modeling, feature engineering, and model selection, we aim to provide users with an effective tool to distinguish between real and fake news, contributing to the fight against misinformation.

## 2.5 Innovations in Detection Approaches

Researchers have introduced methods for detecting organized disinformation campaigns and classifying news diffusion graphs [5]. While deep learning techniques like RNNs and CNNs are emphasized, the absence of image deep fake detection is a research gap [1]. Our study integrates graph convolutional networks for image-based fake news detection, providing a comprehensive approach. We utilize advanced image processing algorithms to identify manipulated images in fake news articles.

Another study [7] combated misinformation in low-resource languages, employing machine learning and NLP techniques. It introduced a classifier for distinguishing fake news, acknowledging limitations in adapting to evolving tactics. A different analysis [9] identified thematic clusters in fake news research and mapped them to Sustainable Development Goals. It evaluated generative AI's role in propagation but faced limitations due to biases in bibliometric data and omission of non-indexed literature. The reviewed literature offers insights into fake news detection methodologies, highlighting strengths and limitations. Opportunities for improvement include addressing dataset biases, reliance on user-generated content, and the need for comprehensive modeling techniques. Future research can learn from these drawbacks to develop robust solutions for combating fake news. Each study contributes to our understanding of fake news detection complexities, laying the groundwork for future advancements.

In conclusion, existing research on fake news detection faces limitations such as focusing solely on news propagation, reliance on user-generated content, and insufficient consideration of image deep fake detection. Our study addresses these limitations by incorporating advanced techniques such as content analysis, robust data collection, sentiment analysis from multiple sources, propagation dynamics analysis, and image-based fake news detection. By learning from these drawbacks, our study aims to develop robust solutions for combating fake news, thus advanc-

ing the field. The combination of these methodologies provides a comprehensive approach to detecting and mitigating the spread of misinformation, ultimately contributing to the fight against fake news.

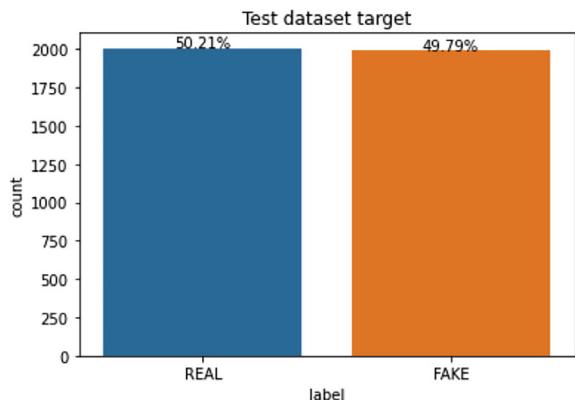
### 3 Dataset

The exploratory setup utilized the integration of three different datasets from Kaggle [10], Reuters [11], and BuzzFeed Political [12] coming about in a comprehensive dataset comprising 6,336 occasions with id, title, content, and name columns. The fastidious curation of this combined dataset included a thinking part for demonstrating, preparing, and assessment, with 4,014 lines committed to preparing and 2,322 lines for consequent evaluation (Fig. 1). Data from the title and content columns was utilized for substance representation, advertising brief rundowns and the most body substance of articles, separately. Progressed Characteristic Dialect Handling (NLP) methods were utilized amid preprocessing, counting lowercasing, disposal of single-letter words, and those containing numbers, tokenization, accentuation expulsion, avoidance of halt words, part-of-speech labeling, and lemmatization for word normalization. These preprocessing steps collectively improved and standardized the printed information, building up a solid establishment for consequent investigation and demonstrating advancement inside the setting of the blended datasets.

### 4 Proposed System

We have developed a sophisticated approach (Fig. 2) to improve the detection of fake news as part of our dedication to this cause. Natural language processing, feature engineering, and machine learning are the three core components that this methodology

**Fig. 1** Dataset composition



combines. We hope to reinvent false news detection by combining these state-of-the-art methods, providing users with a more effective way to distinguish between real and fake news. Now let's dive into each algorithm's specifics and see how their combined efforts enhance the potency of our suggested approach. The methodology provided seven modules: preprocessing of the data, vectorizing the dataset, baseline modeling, feature engineering, final model selection, training, and assessment of the model.

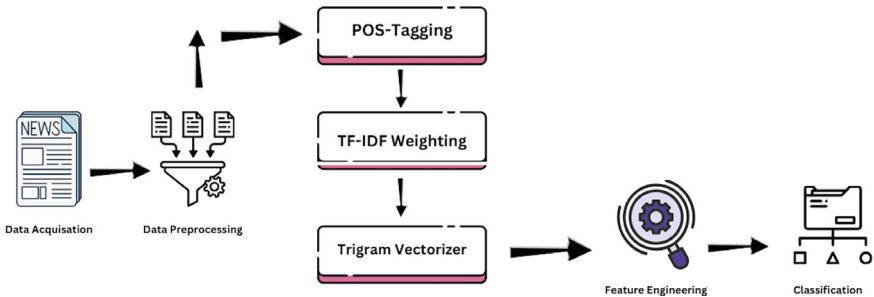
1. Data preprocessing: This module focuses on preparing the dataset for effective modeling. Text cleaning involves converting text to lowercase, eliminating single-letter words, handling numbers, and employing tokenization. Advanced Natural Language Processing (NLP) techniques such as Part-of-Speech (POS) tagging and lemmatization are applied for enhanced text normalization. The goal is to refine the dataset by removing noise and ensuring uniformity, setting the stage for subsequent modeling steps.

2. Vectorizing Dataset: This module involves transforming textual data into a numerical format, a crucial step in machine learning. Techniques like Term Frequency-Inverse Document Frequency (TF-IDF), Bigram Count Vectorizer, and Trigram Vectorizers are employed. These methods convert the text into structured numerical representations, facilitating the understanding and processing of the information by machine learning models [6]. The vectorized dataset serves as the input for subsequent modeling stages, enabling the algorithms to make sense of the textual information and extract meaningful patterns.

3. Baseline Modeling: This module centers around implementing baseline classification models as a foundational step in the model development process. Margin Maximizing Classifier [8], Naïve Bayes, and MaxEnt Classifier are employed to establish initial benchmarks for performance evaluation. The models are trained on the preprocessed dataset to provide an understanding of their effectiveness in distinguishing between fake and real news articles. This step aids in identifying promising models for further refinement in subsequent stages.

4. Feature Engineering: Feature engineering is a critical aspect of enhancing model accuracy. This module explores explicit Part-of-Speech (POS) tagging, TF-IDF weighting, and Bigram Count Vectorizer techniques to extract meaningful features from the dataset. By incorporating linguistic information and identifying significant word associations, these methods contribute to improved model understanding and discrimination capabilities. Feature engineering acts as a bridge between raw text data and model input, elevating the overall predictive power of the classification models.

5. Select Final Model: Building on insights from the baseline models, this module aims to identify the most effective model for deployment. Model selection involves evaluating performance metrics and considering factors such as precision, recall, and F1-score. The chosen model will proceed to the next phase, ensuring that the selected algorithm aligns with the paper's objectives and offers optimal accuracy in classifying news articles.



**Fig. 2** Conceptual illustration of workflow of fake news detection

6. Train Evaluate Model: In this module, the selected model is trained on the preprocessed dataset to learn patterns and associations between features. Evaluation metrics, including accuracy and confusion matrices, are employed to assess the model's performance. Iterative adjustments may be made to fine-tune the model for optimal results, ensuring robustness in distinguishing between fake and real news.

## 5 Algorithms

### 5.1 Max Entropy

The Maximum Entropy (MaxEnt) model for detecting fake news functions as a sophisticated evaluator, scrutinizing textual content to estimate the likelihood of articles belonging to specific categories, such as real or fake news (Algorithm 1). Operating on the principle of maximizing entropy, the model considers prior knowledge encoded in word frequencies for each category, akin to Probabilistic Classifier, but with a focus on capturing complex relationships within the data [13]. Unlike Margin Maximizing Classifier, MaxEnt seeks to achieve a balance between maximizing uncertainty and incorporating available information. The model thrives on flexibility, adapting to intricate patterns in the input space [14]. For this model, the input consists of preprocessed text enriched through feature engineering techniques such as part-of-speech tagging (POS), TF-IDF weighting, and trigram vectorization. These enhancements enable the MaxEnt model to discern detailed linguistic details, contributing to its effectiveness in accurately classifying news articles as either genuine or fabricated.

**Algorithm 1** Maximum Entropy (MaxEnt) Algorithm

---

```

1: Input Training data  $\{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$  where  $x_i$  is a feature vector and  $y_i \in \{1, 2, \dots, C\}$  is the class label
2: Output MaxEnt model parameters: weight vector  $\mathbf{w}$ 
3: Initialize  $\mathbf{w}$  to zeros
4: Choose a convergence threshold  $\epsilon$ 
5: repeat
6:   Set  $\Delta\mathbf{w} \leftarrow 0$ 
7:   for each feature  $f$  and each class  $c$  do
8:     Calculate empirical count:  


$$E(f, c) = \sum_{i=1}^m \mathbf{1}(f(x_i, y_i) = f \text{ and } y_i = c)$$

9:     Calculate model count:  


$$M(f, c) = \sum_{i=1}^m P(y = c|x_i) \cdot \mathbf{1}(f(x_i, y_i) = f)$$

10:    Update feature weight:  $\Delta w_{f,c} = \frac{1}{C} \log \left( \frac{E(f,c)}{M(f,c)} \right)$ 
11:    Update total weight:  $\Delta\mathbf{w} \leftarrow \Delta\mathbf{w} + \Delta w_{f,c}$ 
12:   end for
13:   Update weights:  $\mathbf{w} \leftarrow \mathbf{w} + \Delta\mathbf{w}$ 
14: until  $\|\Delta\mathbf{w}\| < \epsilon$ 
15: return  $\mathbf{w}$ 

```

---

## 5.2 Margin Maximizing Classifier Machine

The Margin Maximizing Classifier Machine (Margin Maximizing Classifier) (Algorithm 2) for fake news detection functions as a discerning agent, analyzing textual cues within news articles to determine their probability of belonging to distinct categories, namely real or fake news [5]. Unlike the Probabilistic Classifier, Margin Maximizing Classifier employs a different methodology, relying on the creation of optimal decision boundaries, or hyperplanes, in a high-dimensional space. It draws upon Bayes' theorem, incorporating prior knowledge derived from word frequencies associated with each category to guide its classification decisions. The model excels at identifying complex patterns and relationships within the text by considering the spatial arrangement of data points. In this context, the input to the Margin Maximizing Classifier model comprises preprocessed text featuring enhanced characteristics through techniques like part-of-speech tagging (POS), term frequency-inverse document frequency (TF-IDF) weighting, and trigram vectorization [15]. These techniques empower the Margin Maximizing Classifier to discern intricate linguistic patterns, ultimately aiding in the accurate categorization of news articles as genuine or fabricated.

## 5.3 Probabilistic Classifier

The Probabilistic Classifiers model (Algorithm 3) is a probabilistic approach used for identifying fake news by analyzing linguistic evidence within news articles. It relies

---

**Algorithm 2** Margin Maximizing Classifier Machine (Margin Maximizing Classifier) Algorithm with Linear Kernel
 

---

- 1: **Input** Training data  $\{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$  where  $x_i$  is a feature vector and  $y_i \in \{-1, +1\}$  is the class label
- 2: **Output** Margin Maximizing Classifier model parameters: weight vector  $\mathbf{w}$ , bias term  $b$
- 3: Initialize  $\mathbf{w}$  and  $b$  to zeros
- 4: Choose learning rate  $\alpha$
- 5: Choose the number of iterations  $T$
- 6: **for**  $t = 1$  to  $T$  **do**
- 7:   **for** each training example  $(x_i, y_i)$  **do**
- 8:     **if**  $y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \leq 1$  **then**
- 9:       Update  $\mathbf{w} \leftarrow \mathbf{w} + \alpha \cdot y_i \cdot \mathbf{x}_i$  {Update weight vector}
- 10:      Update  $b \leftarrow b + \alpha \cdot y_i$  {Update bias term}
- 11:     **end if**
- 12:   **end for**
- 13: **end for**
- 14: **return**  $\mathbf{w}, b$

---

on Bayes' theorem and prior knowledge gained from word frequencies associated with different categories, such as real or fake news [6]. By assessing the relevance of individual words based on accumulated wisdom, the model rates the likelihood of an article belonging to a specific category. This approach assumes that language hints function independently, simplifying the analysis process. The model takes pre-processed text as input and incorporates feature engineering techniques like part-of-speech tagging, TF-IDF weighting, and trigram vectorization [16] to enhance its ability to detect patterns and details in textual data, thereby effectively distinguishing between genuine and fake news stories.

---

**Algorithm 3** Probabilistic Classifier Algorithm for Text Classification
 

---

- 1: **Input** Training data  $\{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$  where  $x_i$  is a document represented as a bag-of-words and  $y_i \in \{1, 2, \dots, C\}$  is the class label
- 2: **Output** Probabilistic Classifier model parameters: class priors  $P(y)$ , and class-conditional probabilities  $P(x_j|y)$  for each word  $x_j$
- 3: **for** each class  $c \in \{1, 2, \dots, C\}$  **do**
- 4:   Calculate class prior:  $P(y = c) = \frac{\text{count}(y=c)}{m}$  {Number of documents with class  $c$ }
- 5:   **for** each word  $x_j$  in the vocabulary **do**
- 6:     Calculate class-conditional probability:  $P(x_j|y = c) = \frac{\text{count}(x_j \text{ in class } c)+1}{\sum_w (\text{count}(w \text{ in class } c)+1)}$  {Laplace smoothing}
- 7:   **end for**
- 8: **end for**
- 9: **return**  $P(y), P(x_j|y)$  for each class and word

---

**Table 1** Comparison of various existing works

Id	Approach	Methodologies	Strengths	Limitations
Ajith et al. [1]	Graph Neural Network (GNN)	Trust-based strategies, Bot filtration	Superior performance in identifying fake news spreaders	Focuses on news spread, not content analysis
Zhao et al. [4]	Propagation study	Prediction models (Probabilistic Classifier, Bayesian Network, J48)	Promising results with F-score of 69.7%	Data mining challenges, dataset quality and biases
Sharma et al. [6]	SA-HyperGAT model	High-order dependencies, Sentiment-aware hypergraph	Innovative approach to word and sentence dependencies	Reliance on user comments, sparse comment data
Rani et al. [8]	Fake news dissemination study	Advanced NLP and machine learning techniques	Highlights need for advanced algorithms	Limited examination of propagation patterns, integration of user feedback
Dong et al. [5]	Disinformation campaign detection	Deep learning (RNNs, CNNs)	Emphasis on news diffusion graphs	Absence of image deep fake detection
Rath et al. [7]	Low-resource language study	Machine learning, NLP techniques	Classifier for fake news detection in low-resource languages	Adaptation to evolving tactics
Nair et al. [9]	Thematic cluster analysis	Generative AI, Mapping to SDGs	Identified thematic clusters, evaluated generative AI	Biases in bibliometric data, omission of non-indexed literature

## 6 Results and Discussion

The efficacy of various models in distinguishing between fake and genuine news articles was evaluated through a series of tests, and the results are presented below. The accuracy of these models, as depicted in Table 1, serves as a crucial metric for assessing their performance.

By integrating Part-of-Speech (POS) labeling along with the Edge Maximizing Classifier (EMC) and Maximum Entropy (MaxEnt) algorithms, an impressive accuracy of 0.91 was achieved. This approach harnesses POS labeling to capture the syntactic structure and linguistic cues within the content, aiding in the discrimination between genuine and fabricated news articles. The Edge Maximizing Classifier and MaxEnt algorithms excel in handling intricate decision boundaries, making them

**Table 2** Comparison of model accuracy with various feature engineering techniques

Models	Accuracy
POS tagging (MaxEnt)	<b>0.91</b>
POS tagging (Probabilistic)	<b>0.89</b>
POS tagging + TF-IDF (MaxEnt)	<b>0.93</b>
POS tagging + TF-IDF (Probabilistic)	<b>0.82</b>
MaxEnt on Trigram + TF-IDF	<b>0.95</b>

particularly well-suited for this task. Their ability to effectively navigate detailed linguistic features alongside POS information enhances the model's capability to discern between real and fake news with a high degree of accuracy.

When Naïve Bayes was utilized in conjunction with POS labeling, the exactness remained tall at 0.89. Naïve Bayes could be a probabilistic classifier that accepts autonomy between highlights, and its adequacy in combination with POS labeling proposes that the syntactic data given by POS labeling is especially valuable in recognizing between genuine and fake news.

Presenting Term Frequency-Inverse Record Recurrence (TF-IDF) near POS labeling, especially in conjunction with the Edge Maximizing Classifier and MaxEnt, encourages progress in the precision to 0.93. TF-IDF could be a commonly utilized procedure in common dialect handling that allots weights to words based on their frequency in a record and their irregularity within the whole corpus. By joining TF-IDF, the models are able to capture the significance of particular words in segregating between veritable and misleading news articles.

In any case, the combination of POS labeling and TF-IDF with Naïve Bayes showed a somewhat lower exactness of 0.82. This result proposes that Naïve Bayes may not be as successful in leveraging the combined data from POS labeling and TF-IDF compared to the Edge Maximizing Classifier and MaxEnt.

An eminent standout among the models was the MaxEnt model, which leveraged Trigram features in conjunction with TF-IDF, achieving an impressive accuracy score of 0.95. Trigram features capture the sequential connections between three adjacent words within the text, providing additional relevant information for distinguishing between genuine and fake news. The integration of TF-IDF further enhanced the effectiveness of this model, highlighting the importance of both contextual information and word significance weights in improving fake news detection. This combination allowed the model to better discern patterns and details within the text, leading to more accurate classification results. These come about, as delineated in Tables 1 and 2, giving a comprehensive diagram of the qualities and weaknesses of different demonstrated arrangements, advertising profitable experiences for optimizing fake news location frameworks.

In conclusion, this research has demonstrated the effectiveness of leveraging advanced natural language processing and machine learning techniques for the critical task of fake news detection. By implementing the Maximum Entropy (MaxEnt)

**Table 3** Classification metrics

Class name	Precision	1-Precision	Recall	1-Recall
Real	0.9381	0.0619	0.9604	0.0396
Fake	0.9603	0.0397	0.9379	0.0621
<i>Overall metrics</i>				
Accuracy	0.9490			
Misclassification rate	0.0510			
Macro-F1	0.9490			
Weighted-F1	0.9490			

classifier in conjunction with feature engineering methods such as TF-IDF weighting and trigram models, we have achieved highly promising results in accurately distinguishing between real and fabricated news articles. The MaxEnt model, combined with TF-IDF and trigram features, exhibited an impressive accuracy of 0.95, outperforming other model configurations explored in this study. This superior performance highlights the model's capability to capture intricate linguistic patterns and word associations, enabled by the synergistic combination of these techniques (Table 3 and Fig. 3).

(1-recall) and (1-precision) are key metrics in binary classification, offering insights into the model's performance. (1-recall) indicates missed positive instances among all actual positives, reflecting the model's tendency to overlook positives. (1-precision) reveals incorrect positive predictions among all instances classified as positive, highlighting misclassification issues. Both metrics complement precision and recall, enriching the understanding of the model's capabilities and limitations.

**Fig. 3** Confusion matrix

		Training Set		
		Real	Fake	SUM
TARGET	OUTPUT			
	Real	606 47.53%	40 3.14%	646 <b>93.81% 6.19%</b>
Fake	25 1.96%	604 47.37%	629 <b>96.03% 3.97%</b>	
	SUM	631 <b>96.04% 3.96%</b>	644 <b>93.79% 6.21%</b>	1210 / 1275 <b>94.90% 5.10%</b>

## References

1. Ajith TT, CV AK, Nandakishore J, Subramanian MSA, S SR (2021) Enhanced movie recommendation using knowledge graph and particle filtering. In: 2021 2nd International Conference on Smart Electronics and Communication (ICOSEC). IEEE, Trichy, India, pp 1139–1144. <https://doi.org/10.1109/ICOSEC51865.2021.9591834>
2. Kumar Santhosh NC, Sailaja M, Ali Hussain M, Rahman SZ (2022) Applications of machine learning for fake news detection in social networks. *Int J Recent Innov Trends Comput Commun* 10(2s):146–150
3. Singh J, Gupta A (2023) Fake news detection using BERT and Elmo: a comparative study. *Int J Inf Technol* 16(2):567–577
4. Zhao Z, Zheng P, Xu S, Wu X, Huang J (2019) A multi-modal deep learning approach for fake news detection. *IEEE Trans Knowl Data Eng* 31(12):2631–2646. <https://doi.org/10.1109/TKDE.2018.2880571>
5. Dong D, Fuqiang L, Guowei L, Bo L (2022) Sentiment-aware fake news detection on social media with hypergraph attention networks. In: Proceedings of the 2022 conference on neural information processing systems, pp 2174–2180
6. Sharma U, Sajeet GP, Rani SS (2022) Personalized fashion recommendation using nearest neighbor pagerank algorithm. In: 2022 International conference on connected systems & intelligence (CSI). IEEE, pp 1–6
7. Rath B, Salecha A, Srivastava J (2022) Fake news spreader detection using trust-based strategies in social networks with bot filtration. *Soc Netw Anal Min.* <https://doi.org/10.1007/s13278-022-00890-z>
8. Rani SS, Baby SS (2022) Real-time influencer detection in twitter using a hybrid approach. *Proc Comput Sci* 215:461–470
9. Nair A, Harikumar G, Vissutha MP, Ajanalakshmi D, Deepthi LR (2022) Classification of trust in social networks using machine learning algorithms. In: 2022 third international conference on intelligent computing instrumentation and control technologies (ICICICT). IEEE, pp 501–505
10. Kaggle: WELFake\_Dataset. [https://www.kaggle.com/datasets/saurabhshahane/fake-news-classification?select=WELFake\\_Dataset.csv](https://www.kaggle.com/datasets/saurabhshahane/fake-news-classification?select=WELFake_Dataset.csv)
11. Reuters-21578. <https://paperswithcode.com/dataset/reuters-21578>
12. BuzzFeed-Webis Fake News Corpus 2016. <https://paperswithcode.com/dataset/buzzfeed-webis-fake-news-corpus-2016>
13. Alquran H, Banitaan S (2022) Fake news detection in social networks using data mining techniques. In: Proceedings of the 2022 international conference on artificial intelligence and internet of things (AIIoT). IEEE, pp 1–6. <https://doi.org/10.1109/AIIoT54504.2022.9817287>
14. Sa D, Chitturi B (2019) Deep neural approach to Fake-News identification. In: Proceedings of the international conference on computational intelligence and data science (ICCIDIS 2019), pp 276–281. <https://doi.org/10.1016/j.procs.2020.03.276>
15. Wang W, He K, Liu T, Bao Y (2019) Attention-based LSTM for early detection of fake news. In: 2019 International joint conference on neural networks (IJCNN). IEEE, pp 1–8. <https://doi.org/10.1109/IJCNN.2019.8811091>
16. Lal NM, Krishnanunni S, Vijayakumar V, Vaishnavi N, Rani SS, Raj KD (2021) A novel approach to text summarisation using topic modelling and noun phrase extraction. In: Advances in computing and network communications: proceedings of CoCoNet 2020, vol 2. Springer, Singapore, pp 285–298

# VR Phantom Haven: Phantom Limb Pain Management Using Virtual Reality



Aditya Shah, Siddhi Muni, Gautam Mehendale, and Chetashri Bhadane

**Abstract** Phantom limb pain (PLP) remains a persistent challenge for amputees, with current treatments often falling short. This research delves into the potential of virtual reality (VR) as a groundbreaking approach for PLP management. We designed a VR-based system incorporating features specifically tailored to address PLP complexities, including interactive interfaces, customizable environments, and real-time pain detection using machine learning models achieving high accuracy (92%) and precision (94%). A pilot test with three patients yielded promising results: all participants reported a significant decrease in pain levels, from level 3 to 0, within an average of 30–45 min of VR exposure. These preliminary findings suggest VR's potential to offer a safe, engaging, and potentially effective approach to PLP management. Further research with larger and more diverse participant groups is necessary to solidify these results and explore VR's broader impact on amputee well-being, paving the way for VR as a transformative tool in chronic pain management.

**Keywords** PLP · Phantom pain management · Virtual reality · Explainable AI · XAI · SHAP

## 1 Introduction

Phantom limb pain (PLP) is a cruel deception of the human body. Amputees continue to experience vivid sensations of pain, burning, tingling, or throbbing in the limb that is no longer present. It's a testament to the profound interconnectedness of the brain and body, where the brain struggles to reconcile the loss of sensory input with its established body map. The reported prevalence of PLP varies, but estimates suggest it affects up to 80% of amputees, significantly impacting their daily lives and overall

---

A. Shah (✉) · S. Muni · G. Mehendale · C. Bhadane  
Dwarkadas J. Sanghvi College of Engineering, Bhaktivedanta Swami Marg, Mumbai 400056,  
Maharashtra, India  
e-mail: [adityashah841@gmail.com](mailto:adityashah841@gmail.com)

C. Bhadane  
e-mail: [chetashri.bhadane@djsce.ac.in](mailto:chetashri.bhadane@djsce.ac.in)

well-being. Despite its prevalence, the exact mechanisms underlying PLP remain a puzzle. Current theories suggest that the brain, deprived of sensory input from the missing limb, attempts to reorganize, leading to misinterpretations and the perception of pain in a non-existent body part.

Traditional treatment strategies for PLP often fall short of providing lasting relief. Medications, nerve blocks, and physical therapy can offer some benefit, but their effectiveness can be limited and vary considerably between individuals. Moreover, these approaches often focus solely on pain reduction, neglecting the psychological toll of PLP, which can include social isolation, anxiety, and depression. This highlights the urgent need for novel and comprehensive interventions that address both the physical and psychological aspects of PLP.

Virtual reality (VR) technology presents a promising avenue for PLP management. VR allows for the creation of immersive and interactive environments that can potentially address PLP through multiple mechanisms [1]. By engaging the user in visually stimulating and interactive experiences, VR can provide a powerful distraction from the sensations of phantom pain. Additionally, VR offers the potential to promote neuroplasticity, the brain's ability to reorganize itself. By incorporating activities that stimulate the somatosensory cortex, the region of the brain responsible for processing sensory information, VR may help the brain remap the body and reduce phantom limb sensations. Furthermore, VR environments can be tailored to individual needs and preferences, allowing for a more personalized and potentially more effective therapeutic approach.

This research paper delves into the potential of VR as a tool for managing PLP. We aim to develop a sophisticated VR application specifically designed for this purpose. The application will incorporate features such as interactive interfaces, customizable environments, real-time physiological monitoring, and therapist-controlled usage protocols. By creating a safe, engaging, and adaptable VR experience, we hope to offer individuals with PLP a novel and effective approach to pain management, improved quality of life, and a renewed sense of embodiment.

## 2 Literature Review

Cheung et al. [2] examine the efficacy of X-reality technologies (virtual, augmented, and mixed reality) in managing phantom limb pain (PLP) among amputees, highlighting various interventions and their impact on pain reduction. Data collection included search strategies, screening, data extraction, and meta-analysis. The average pain reduction post-intervention was 2.30 (95% CI: -3.38 to -1.22), with virtual reality showing the most significant effect, reducing pain by 2.83 (95% CI: -4.43 to -1.22). The study calls for broader applicability and diverse study designs.

Another study evaluated the effectiveness of virtual reality (VR) in treating pain across different age groups, finding VR significantly reduces anxiety, pain unpleasantness, and pain intensity, though it showed moderate efficacy for chronic pain conditions like low back pain and cancer-related pain [3]. A meta-analysis of 31

randomized controlled trials up to October 2020 was conducted, reviewing sources such as the Cochrane Library, PubMed, EMBASE, and Web of Science. Results indicated VR was particularly effective in younger patients, also reducing anxiety, pain unpleasantness, heart rate, and facilitating quicker dressing changes. Further research is needed to explore VR's potential in chronic pain management.

The potential of VR as a non-pharmacological treatment for myofascial pain syndrome (MPS) was discussed, noting VR's success in neuroplasticity and acute pain management but its untested application for MPS [4]. The study proposes routine VR therapy sessions for MPS and compares outcomes with simulated VR, referencing a study on chronic low back pain. Methodological limitations, including patient selection and reliance on self-reported data, were acknowledged. Further trials are recommended to assess VR's effectiveness as an adjunct to traditional therapies like trigger point injections (TPIs).

In the context of brachial plexus injuries (BPIs), a single case study showed immersive VR with haptic feedback significantly reduced pain by 50% and improved range of motion [5]. This suggests VR's potential in BPI rehabilitation. However, the study emphasizes the need for larger sample sizes and further research to confirm VR's efficacy for BPIs.

A study developed and evaluated VR therapy for veterans with phantom sensations and limb pain (PLP), finding high user satisfaction and significant reductions in PLP intensity and unpleasant phantom sensations [6]. Fourteen participants underwent VR therapy similar to mirror therapy, with assessments conducted before and after the intervention. Following VR therapy, phantom sensations and PLP intensity decreased markedly, though the study's small sample size suggests further research with larger, more diverse populations is needed.

Another study investigates phantom limb pain (PLP) by focusing on abnormal sensorimotor cortical representations [7]. It demonstrates that brain-computer interface (BCI) training effectively reduces PLP by inducing cortical reorganization and weakening the phantom hand representation. Unlike traditional methods, BCI training reduces pain without explicit phantom hand movements or visual feedback, highlighting the crucial role of cortical plasticity in managing PLP.

Lastly, a comprehensive review assessed the challenges in treating PLP and evaluated various therapies, noting the lack of high-quality evidence for establishing a first-line treatment [8]. The review of 38 treatments, using databases like MEDLINE, EMBASE, and Cochrane, identified one notable trial with repetitive transcutaneous magnetic stimulation showing short-term pain reduction. Moderate quality trials indicated some efficacy for gabapentin, ketamine, and morphine, but raised bias concerns. The study highlights the necessity for more robust research to identify definitive PLP treatments.

Table 1 shows the tabular summary of 5 research works along with their research gaps which are addressed in this paper.

**Table 1** Prominent research gaps

Name of the paper	Research gap
Virtual and augmented reality-based treatments for phantom limb pain: A systematic review	Broader applicability and diverse study designs are needed to validate the findings across various populations and settings
A virtual reality intervention for treating phantom limb pain: Development and feasibility results	Small sample size; further research with larger, more diverse populations must confirm efficacy
Using virtual reality exposure therapy in pain management: A systematic review and meta-analysis of randomized controlled trials	Moderate efficacy for chronic pain conditions; further investigation is required to explore VR's potential for long-term pain management
Virtual reality combined with robotic facilitated movements for pain management and sensory stimulation of the upper limb following a Brachial Plexus injury: A case study	Single case study; larger sample sizes are needed to confirm VR's efficacy for Brachial Plexus injuries
A review of the management of phantom limb pain: Challenges and solutions	Existing treatments focus primarily on pain reduction, often neglecting the psychological aspects of PLP; comprehensive interventions are needed

### 3 Methodology

The methodology used in this research can be broadly divided into two sections as follows:

1. User Interface
2. Pain Level Detection

#### 3.1 User Interface

##### 3.1.1 Patient Interface

Patients are given permission to use the proposed system via a secure login portal, and they should use their virtual reality (VR) environment nickname that is designated to them along with the time period stated by the doctor. This method ensures that VR sessions end when the time allocated runs out. Upon authentication, amputee patients are seamlessly integrated into a VR setting tailored specifically to their therapeutic needs. The interface within this environment functions dually as a navigator and an interactive companion, facilitating a guided and engaging experience throughout the session. The user wears an Apple Watch during the session, which serves for both data collection and safety monitoring. At the end of each VR session, the system generates a report based on medical telemetry from the session and sends it to the

doctor. The user's future therapy is modified using information obtained from VR session reports, followed by scheduling for another one as per this new data.

### **3.1.2 Virtual Reality World**

The virtual reality (VR) environment features a user-controlled timer displayed prominently at the top of the interface, reflecting the session duration as inputted by the user. This VR world is designed with an adventurous pirate theme to enhance the immersive experience and engage the user's interest. Within this thematic setting, users can interact with avatars that bear a resemblance to familiar figures, facilitating communication and further enriching the interactive experience. Additionally, users may also freely roam throughout the vast virtual reality environment, which acts as a distraction to lessen their phantom pain by drawing them in and encouraging interaction with the virtual environment. The user data is collected through an Apple Watch worn by the user during the session, from which 14 principal components are obtained for effectively analyzing the data. If the medical graphs exceed certain thresholds, the VR world automatically shuts down to ensure the user's safety.

### **3.1.3 Doctor Interface**

Following each VR session, the dedicated doctor is provided with a comprehensive report that includes medical graphs showcasing the patient's heart rate, skin conductance, and pain levels. This data allows the doctor to assess the therapeutic impact of the VR experience on the patient. Through the analysis of these parameters, the doctor may more effectively customize the length and intensity of subsequent virtual reality sessions, so promoting a more successful recovery and guaranteeing the patient's comfort and safety throughout the therapy.

## ***3.2 Pain Level Identification***

### **3.2.1 Dataset Description**

Current clinical pain assessment methods lack objectivity and robustness, relying solely on patient self-reported pain levels. Verbal scales, visual analog scales (VAS), and numeric rating scales (NRS) are prevalent tools, but are limited to patients with unimpaired cognitive function. To address this challenge, we present a dataset of bio-potentials for the development of a surrogate pain intensity measure via machine learning [9]. The dataset encompasses data from 85 participants who underwent controlled, painful heat stimuli. Electrodes recorded electromyography (EMG) [10],

skin conductance level (SCL), and electrocardiography (ECG) signals [11]. Feature extraction resulted in a total of 159 features encompassing various mathematical domains: amplitude, frequency, stationarity, entropy, linearity, variability, and similarity.

### 3.2.2 Data Pre-processing

The raw data underwent several preprocessing steps to ensure optimal model performance and interpretability.

Given the minimal presence of missing values (less than 0.1%), a complete-case deletion approach was employed. This strategy minimizes the impact of potential biases introduced by imputation techniques for such a small proportion of missing data.

The data exhibited a balanced distribution across the five pain levels (approximately 1700 data points per level). This characteristic was maintained throughout the preprocessing pipeline to avoid skewing the model towards specific pain levels.

Feature selection focused on identifying features most relevant to pain intensity prediction. This was achieved by calculating the correlation between each feature and the target variable (pain levels). Features with low correlation were removed, resulting in a reduced set of 57 features. This step enhances model efficiency by eliminating redundant or irrelevant information.

Z-score normalization was applied to the remaining features. This technique standardizes the data by subtracting the mean and dividing by the standard deviation, ensuring all features contribute equally during model training, regardless of their original scale.

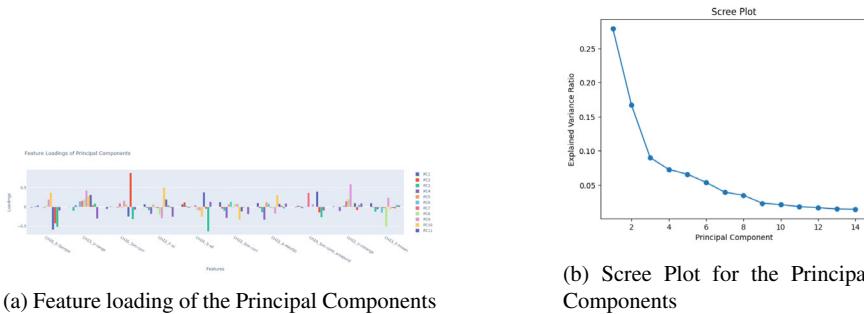
Principal Component Analysis (PCA) [12] was employed to reduce the dimensionality of the data while preserving most of the variance. We selected the top 14 principal components, accounting for over 90% of the total variance, for further analysis as seen in Fig. 1a and b.

The distribution of the principal components exhibited skewness and outliers. To address this, Winsorization [13] was implemented to cap extreme values within a specified number of standard deviations from the median. Finally, Yeo-Johnson transformation [14] was applied to achieve a near-normal distribution for the principal components, further improving model performance and interpretability.

### 3.2.3 Architecture

This research evaluates six distinct architectures, each utilizing different deep learning techniques to predict the pain level data from the processed data.

The first model employs a sequential architecture with rectified linear unit (ReLU) activation functions and L2 regularization. ReLU functions introduce non-linearity within the network, crucial for modeling complex relationships between features and



**Fig. 1** Data processing

pain intensity. L2 regularization helps prevent overfitting by penalizing models with excessively large weights.

The second model incorporated dropout layers and batch normalization for enhanced performance. Dropout layers stochastically drop out neurons during training, thereby mitigating overfitting by reducing co-adaptation between features. Batch normalization addresses internal covariate shift, improving training stability and convergence.

The remaining models (Models 3–6) represent variations on these core deep learning principles. They explore different network configurations, activation functions (employing Exponential Linear Unit (ELU) activations), and optimization techniques to achieve a balance between model complexity, efficiency, and robustness.

To leverage the strengths of these individual models, a soft voting ensemble technique was employed. This approach involves obtaining the output probabilities for each pain level from all six models. Subsequently, weighted averages are calculated for each pain level based on the individual model's precision on the validation set. This weighted summation serves as the final predicted pain intensity for a given data point. By combining the predictions from multiple models, the ensemble approach aims to achieve superior performance and generalizability compared to any single model.

### 3.2.4 Explainable AI (xAI)

To understand the importance and relevance of the 14 PCs extracted, an xAI technique, SHAP (SHapley Additive exPlanations) [15] is used. It is a powerful method used to interpret the classification models for classifying among 5 different categories (pain levels). It is used to identify which PC, out of the 14 PCs has the highest impact on the decision-making process. Firstly, a representative subset of about 2500 data points (500 data points for each pain level category) from the training set is sampled for the background data. Next, features are scaled and selected using MinMaxScaler and SelectKBest methods, respectively. After the KernelExplainer is

initialized, SHAP values are calculated for further analysis and visualization. The following are some SHAP visualization techniques used in the research to analyze the importance of principal components in the model's decision-making:

1. Summary Plots: These visualizations depict the significance of each trait throughout the dataset. They illustrate the Shapley values for each feature, indicating how they influence the model's predictions, shifting them away from the average or base estimate as seen in Fig. 2a.
2. Waterfall Plots: Waterfall plots visualize how each feature's Shapley value contributes to the final prediction of a single instance. They facilitate a clear understanding of how these contributions add up to influence the overall prediction as seen in Fig. 2b.

## 4 Results

The developed machine learning models demonstrated high accuracy and precision in pain recognition. Individual models achieved an average accuracy of 86% and a precision of 89% in classifying pain levels as seen in Table 2. The implementation of an ensemble voting system, weighted by the individual model's precision on the validation set, further improved these metrics. The ensemble system achieved an overall accuracy of 92% and a precision of 94%, indicating a significant improvement in pain level classification compared to individual models.

A pilot test with three patients was conducted to evaluate the efficacy of the proposed VR-based PLP management system. In all three cases, the system successfully detected and tracked a gradual decrease in pain levels, ranging from level 3 (moderate) to level 0 (no pain) within an average timeframe of 30 to 45 minutes of VR exposure. These initial findings suggest the potential of the VR system as a promising tool for PLP management.

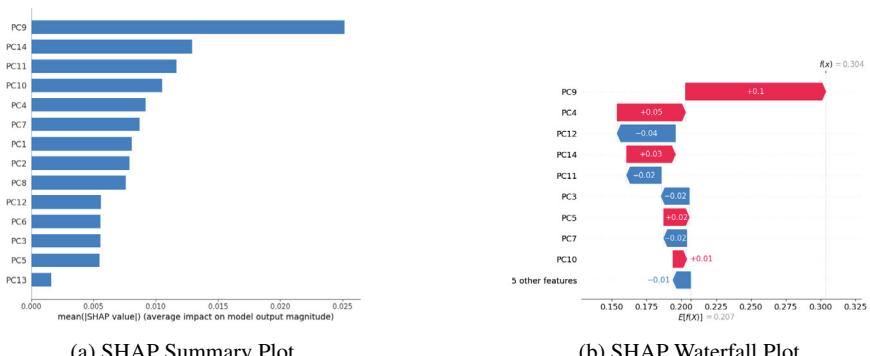


Fig. 2 Explainable AI (XAI): SHAP

**Table 2** Results of individual deep learning architectures

Models	Accuracy (%)	Precision (%)
Model 1	83.4	82.6
Model 2	81.8	83.8
Model 3	80.9	<b>89</b>
Model 4	<b>86</b>	80.6
Model 5	85.5	82.5
Model 6	81.4	84.4

The high accuracy and precision achieved by the machine learning models demonstrate their effectiveness in classifying pain levels based on bio-potential signals. Furthermore, the ensemble voting approach further enhances performance, highlighting the benefits of leveraging multiple models for improved generalizability. The successful pilot testing of the VR-based PLP management system provides preliminary evidence for its potential to alleviate PLP through immersive VR experiences. While the sample size was limited, these results warrant further investigation in a larger clinical trial to confirm the efficacy and long-term benefits of this novel approach.

## 5 Conclusion

Our pioneering VR-based system for managing Phantom Limb Pain (PLP) leverages advanced real-time pain detection powered by machine learning, achieving impressive rates of 92% accuracy and 94% precision. In a breakthrough pilot test, the system was remarkably effective, with each of the three participants experiencing a complete elimination of pain—from level 3 to 0—within a mere 30–45 minutes of VR engagement. Although the study was limited by its small sample size, the results powerfully suggest that VR not only offers a safe and immersive treatment alternative but could also redefine effective strategies for PLP management. To solidify these promising findings and to explore the potential for VR to improve psychological well-being, further research with larger cohorts is imperative. This work represents a significant step forward in the application of VR technology for pain relief, potentially transforming the quality of life for amputees worldwide.

## References

1. Vassantachart AY, Yeo E, Chau B (2022) Virtual and augmented reality-based treatments for phantom limb pain: a systematic review. *Innov Clin Neurosci* 19(10–12):48–57. PMID: 36591552; PMCID: PMC9776775. (Oct–Dec 2022)
2. Cheung JC-W, Cheung DSK, Ni M, Chen K-W, Mao Y-J, Feng L, Lam W-K, Wong DW-C, Leung AK-L (2023) X-reality for phantom limb management for amputees: a systematic review and meta-analysis. *Eng Regen* 4(2):134–151. <https://doi.org/10.1016/j.engreg.2023.02.002>
3. Huang Q, Lin J, Han R, Peng C, Huang A (2022) Using virtual reality exposure therapy in pain management: a systematic review and meta-analysis of randomized controlled trials. *Value Health* 25(2):288–301. <https://doi.org/10.1016/j.jval.2021.04.1285>
4. Wang EF, Jotwani R (2023) Virtual reality therapy for myofascial pain: evolving towards an evidence-based non-pharmacologic adjuvant intervention. *Interv Pain Med* 2(1):100181. <https://doi.org/10.1016/j.inpm.2023.100181>
5. Snow PW, Dimante D, Sinisi M, Loureiro RCV (2022) Virtual Reality combined with Robotic facilitated movements for pain management and sensory stimulation of the upper limb following a Brachial Plexus injury: a case study. *IEEE Int Conf Rehabil Robot* 2022:1–6. <https://doi.org/10.1109/ICORR55369.2022.9896552>. (July, 2022)
6. Rutledge T, Velez D, Depp C, McQuaid JR, Wong G, Jones RCW, Atkinson JH, Giap B, Quan A, Giap H (2019) A virtual reality intervention for the treatment of phantom limb pain: development and feasibility results. *Pain Med* 20(10):2051–2059. <https://doi.org/10.1093/pmt/pnz121>. (1 Oct 2019)
7. Yang H, Yanagisawa T (2024) Is phantom limb awareness necessary for the treatment of phantom limb pain? *Neurol Med Chir (Tokyo)*. 64(3):101–107. <https://doi.org/10.2176/jnmc.2023-0206>. Epub 2024 Jan 24. PMID: 38267056; PMCID: PMC10992984. (15 Mar 2024)
8. Richardson C, Kulkarni J (2017) A review of the management of phantom limb pain: challenges and solutions. *J Pain Res* 10:1861–1870. <https://doi.org/10.2147/JPR.S124664>. (7 Aug 2017)
9. Gruss S, Walter S, Traue HC, Werner P, Andrade A (2016) Data from: pain intensity recognition rates via biopattern feature patterns with support vector machines. Zenodo, 01 Oct 2016. <https://doi.org/10.5061/dryad.2b09s>
10. Mills KR (2005) The basics of electromyography. *J Neurol Neurosurg Psychiatry* 76(Suppl 2):ii32–ii35. BMJ Publishing Group Ltd
11. Sörnmo L, Laguna P (2006) Electrocardiogram (ECG) signal processing. In: Wiley encyclopedia of biomedical engineering. John Wiley & Sons, Inc., Hoboken, NJ, USA
12. Maćkiewicz A, Ratajczak W (1993) Principal components analysis (PCA). *Comput Geosci* 19(3):303–342. [https://doi.org/10.1016/0098-3004\(93\)90090-R](https://doi.org/10.1016/0098-3004(93)90090-R)
13. Brownen-Trinh R (2019) Effects of winsorization: the cases of forecasting non-GAAP and GAAP earnings. *J Bus Financ Account*. 46(1–2):105–135. <https://doi.org/10.1111/jbfa.12365>
14. Weisberg S (2001) Yeo-Johnson power transformations. Department of Applied Statistics, University of Minnesota. Accessed 1 June 2003
15. Frye C, de Mijolla D, Begley T, Cowton L, Stanley M, Feige I (2021) Shapley explainability on the data manifold. [arXiv:2006.01272](https://arxiv.org/abs/2006.01272)

# Beyond the Surface: Exploring Segmentation Techniques in DL for Early Brain Tumor Detection



Soni Singh, Pratyush Mishra, Md. Kaish, Jordan-Kény Gnansounou Dansi, Sunaina Singh, Johnstone Joel Ngorma, and Sahla Ambrein

**Abstract** The initial testing and treatment planning of brain tumors using MRI scans depend on brain tumor segmentation, which has a direct effect on patient outcomes. Because manual segmentation takes a lot of time and arduous, automated techniques are required for efficiency. Our paper focuses on reviewing MRI-based brain tumor segmentation techniques, particularly emphasizing recent advancements in deep learning. These deep learning techniques are becoming more and more well-liked because they can process complex image data efficiently and produce cutting-edge outcomes. We begin by introducing the significance of brain tumor segmentation and conventional methodologies before delving into a detailed discussion of cutting-edge deep learning algorithms. By critically evaluating these methods, we aim to elucidate their strengths and limitations, paving the way for their integration into routine clinical practice. Through our comprehensive review, we strive to contribute to the advancement of brain tumor diagnosis, ultimately enhancing patient care in neuro-oncology.

---

S. Singh · P. Mishra (✉) · Md. Kaish · J.-K. G. Dansi · J. J. Ngorma · S. Ambrein

Department of Computer Science and Engineering, Lovely Professional University, Phagwara, Punjab, India

e-mail: [pratyushmishra287@gmail.com](mailto:pratyushmishra287@gmail.com)

S. Singh

e-mail: [sonisingh0107@gmail.com](mailto:sonisingh0107@gmail.com)

Md. Kaish

e-mail: [mdkaish1999@gmail.com](mailto:mdkaish1999@gmail.com)

J.-K. G. Dansi

e-mail: [Jordandansi05@gmail.com](mailto:Jordandansi05@gmail.com)

J. J. Ngorma

e-mail: [ngorimajohnstone@gmail.com](mailto:ngorimajohnstone@gmail.com)

S. Ambrein

e-mail: [ambreinsahla25@gmail.com](mailto:ambreinsahla25@gmail.com)

S. Singh

Department of Electrical & Electronics School of Engineering, University of Petroleum and Energy Studies (UPES), Bidholi, Dehradun, India

e-mail: [sunisingh0306@gmail.com](mailto:sunisingh0306@gmail.com)

**Keywords** Segmentation · Deep learning · Tumor · Healthcare · Analysis and classification

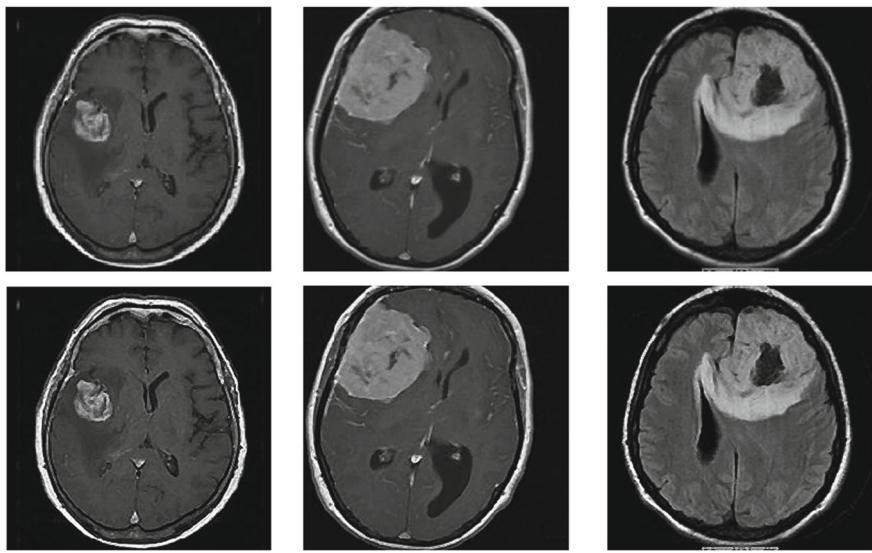
## 1 Introduction

Cancer is the term used to refer to the irregular development and division of cells inside the body, leading to the formation of tumors [1]. In the case of brain tumors, these abnormal cell growths occur within the brain tissue itself. While brain tumors are not as common as some other types of cancer, they are among the most lethal. Based on where they start, brain tumors can be broadly divided into two types [2]: The first one is primary brain tumors, which emerge from cells in brain tissue, and the second one is metastatic brain tumors. Glial cells, which are proliferating brain cells, give rise to gliomas, a kind of initial tumor of the brain. They represent the main focus of current research in brain tumor segmentation. Gliomas are a class of tumors that include low-grade gliomas such as oligodendrogiomas and astrocytomas, as well as the highly aggressive grade IV glioblastoma multiforme (GBM). Treatment of gliomas typically involves a combination of surgical removal, chemotherapy, and radiotherapy, tailored to the specific characteristics and grade of the tumor [3].

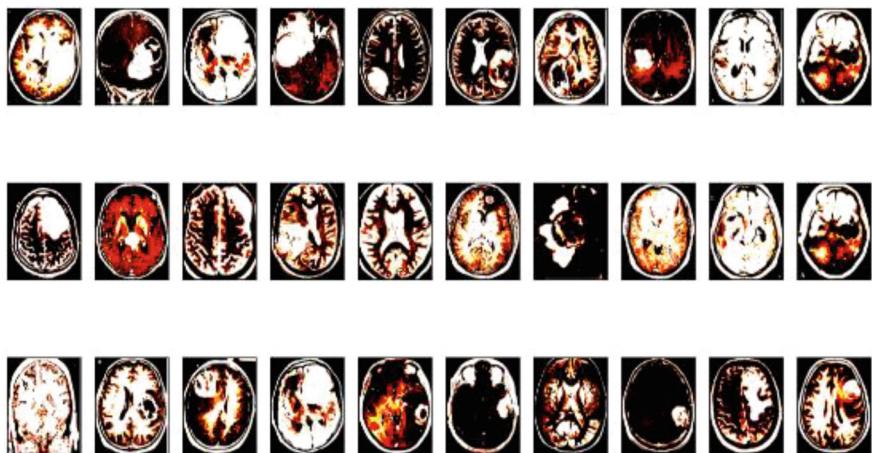
Enhancing treatment results and patient survival rates requires a timely identification of gliomas. Several diagnostic imaging techniques, including magnetic resonance spectroscopy (MRS), can be used to identify brain tumors. Segmentation of brain tumors is a critical step in the treatment planning process, as it involves distinguishing and separating tumor tissues from normal brain tissues as shown in Fig. 1. This segmentation process includes identifying active tumor cells, necrotic core regions, and edema (fluid accumulation) surrounding the tumor as shown in Fig. 2. Currently, categorization typically occurs manually by qualified radiologists in medical centers. This is a laborious and hard operation, particularly when dealing with large numbers of mixed MRI images [2, 4].

Nonetheless, tremendous progress has been made in the last few years in the creation of automatic segmentation systems, especially when utilizing deep learning approaches. These techniques enable the accurate and efficient segmentation of brain tumors by using massive datasets to identify patterns and features in the Fig. 3.

The remainder of the study is structured to provide an in-depth analysis of brain tumor segmentation methods. It starts with a quick overview of the segmentation techniques now in use before going into great detail about deep learning algorithms, which have recently become the cutting edge. The ultimate objective is to increase the precision and efficacy of tumor identification and therapy, which will ultimately help patients suffering from brain tumors [2].



**Fig. 1** MRI images of brain tumor [3]

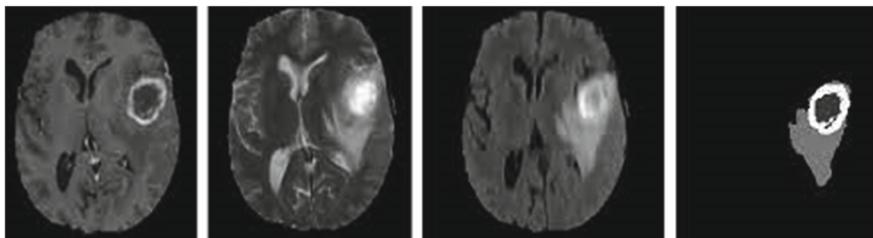


**Fig. 2** Image segmentation [2]

## 2 Various Techniques for Brain Tumor Segmentation

These approaches are classified in three categories: fully, semi, and manual automation based on the degree of the user action is needed. These are mentioned below:

- Manual Segmentation



**Fig. 3** Segmented tumors from left to right [8]

- Semi-automatic Segmentation
- Fully Automatic segmentation.

#### • **Manual Segmentation**

The process entails meticulously examining numerous image slices, identifying tumors, and meticulously delineating their boundaries manually. Despite its effectiveness, manual segmentation suffers from drawbacks: it consumes considerable time, relies heavily on the individual radiologist's skills, and is prone to significant variability both within and between raters. However, it continues to be a widely used method for confirming the results of partially and completely automated segmentation algorithms [7].

#### • **Semi-automatic Segmentation**

Three primary functions of user input are supported by semi-automatic methods in MRI analysis: initialization (beginning the process), intervention or feedback, and result assessment [11]. Additionally, users can tweak the settings of pre-processing techniques to better suit the input images. Apart from starting the process, users can guide the algorithm along the way by giving feedback and making adjustments as needed. They can also evaluate the outcomes and make changes or redo the process if necessary [5].

Hamamci and associates devised the “Tumor Cut” procedure. The maximal tumor diameter on MRI scans can be calculated by the user in this semi-automated segmentation procedure. After that, a two-step, cellular automata (CA)-based seeding tumor segmentation procedure is initiated [13]. This approach obtains the final tumor volume by integrating the data after the algorithm is applied independently to each MRI modality.

Recently, a semi-automatic method was conducted using a new categorization approach system, where a brain tumor is trained and classed inside the same brain as shown in Fig. 2. Therefore, it is necessary to address noise and intensity bias correction. These subsets of voxels are used in a method to extract characteristics such as intensity values and spatial coordinates [5].

- **Segmentation by Fully Automatic method**

Segmentation using fully automatic methods in image processing divides an image into meaningful segments without human intervention. Techniques include thresholding, clustering, region growing, edge-based methods, watershed transformation, graph-based methods, deep learning, feature-based methods, MRFs, and mean-shift clustering [6]. Thresholding sets a value to classify pixels, while clustering groups pixels by similarity. Region growing iteratively adds neighboring pixels meeting criteria.

Edge-based methods detect boundaries, and watershed treats images as topographic surfaces [9]. Graph-based methods use graph cuts, deep learning employs CNNs, and feature-based methods analyze texture, color, or shape. MRFs consider spatial relationships, and mean-shift clustering clusters by density. These methods cater to various image characteristics and application needs [10].

- **Challenges in Fully Automatic method**

Automatic glial tumor segmentation is an extremely challenging subject. Tumor location, size, and form vary from patient to patient in a 3D data collection consisting of brain MRI data. Moreover, a major challenge is that tumor borders are often unclear, irregular, and discontinuous, especially when contrasted with traditional edge-based methods [11].

- **Segmentation with BRATS Dataset**

Analyzing the output of the state-of-the-art brain tumor picture segmentation methods realistically is a challenging endeavor. The BRATS benchmark—with this shared dataset, an objective comparison of several glioma segmentation methods is possible [12]. There are 110 scans available for testing with unknown grades and unknown background information. The Dice Score, which gauges the degree of overlap between the segmented tumor region of the ground truth and its actual tumor area [14]. This score aids in evaluating how well the segmentation technique captured the tumor regions [15].

- **Segmentation using Oslo Data Set**

The Oslo University Hospital provided multi-parametric MRI data, which included T2-weighted, as well as T1-weighted, FLAIR, and T1 post-contrast pictures. 52 patients, all over the age of 18, who had scans between 2003 and 2012 made up this dataset.

The majority of the images featured anisotropic voxels, which are frequently observed in 2D clinical imaging. The Oslo dataset was manually segmented into three labels, edema (ED), non-enhancing tumor core (NEN), and enhancing tumor core (ET), by an in-house neuroradiologist. This procedure was correlated with the BRATS dataset [16].

- **Segmentation method categories**

There are two types of brain tumor segmentation techniques—First is generative and the other one is discriminative technique. In-depth evaluations of these methods were already supplied. Discriminative techniques seek to identify the connection between the input image and the physical reality. Their primary methods are extraction of features and identification.

They utilize supervised learning methods, which are usually called for significant amounts of trustworthy authentic data [17], whereas generative methods construct statistical models using past data on the precise position and dimension of wholesome tissues. Using previously obtained healthy tissue atlases, the unknown tumor compartments are retrieved [19]. Table 1 shows the review of the existing DL model for brain tumor segmentation.

### 3 Network Architecture of Segmentation Method

Gliomas are difficult to segment automatically because of their complicated histology. A unique triple network architecture was created in order to overcome this difficulty [21]. With this architecture, the segmentation issue is addressed by training three different models, each of which is treated as a binary classification problem: one for TC (TC-net), one for ET (EN-net), and one for WT prediction (WT-net) [22].

Figure 4 shows the images analyzed including T1, T2, T2-FLAIR, and post-contrast T1 (T1C). The TC-net segments the tumor’s core (TC), the EN-net segments the enhancing tumor (ET), and the WT-net segments the entire tumor (WT).

The networks employed a method based on 3D patches. Images with multiple parameters were run through the Dense UNet. Dense blocks were made using the 64 feature maps produced by the first convolution [23]. Every dense block had five layers. Four progressively connected sublayers were incorporated in each layer3D convolution, and 3D spatial dropout [18]. For each subsequent layer within the dense block, the input was utilized to produce a set of feature maps, denoted as “k,” which were then combined with the input of the next layer. This process is iterated to generate additional feature maps. Each dense block’s output was connected to the next decoder portion via a skip connection. All feature maps were efficiently used because of this networked system, which also provided direct supervisory signals to every layer in the design.

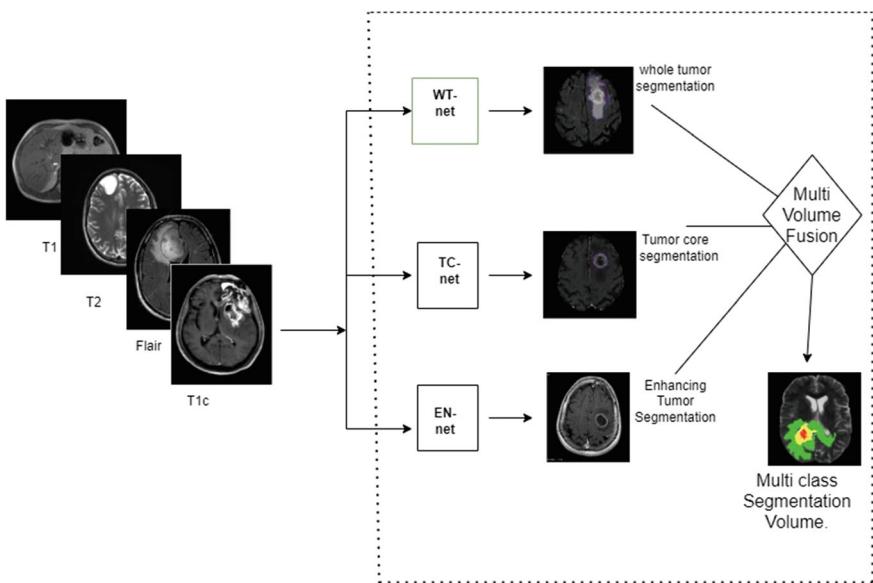
**Table 1** Review of various DL models in brain tumor segmentation

Authors	Method	Level of user interaction	Accuracy for whole tumor	Accuracy for core tumor	Accuracy for active tumor
[1]	Medical training and experience	Manual	0.88	0.93	0.74
[2]	CNN with small (3 * 3) filters for deeper architecture	Fully automatic	0.88	0.83	0.77
[3]	Generative model that performs joint segmentation and registration	Semi-automatic	0.88	0.83	0.72
[4]	Cascaded two-pathway CNNs for simultaneous local and global processing	Fully automatic	0.88	0.79	0.73
[2]	Concatenated RFs, trained using asymmetry and first-order statistical features	Fully automatic	0.87	0.78	0.74
[5]	3D CNN architecture using 3D convolutional filters	Fully automatic	0.87	0.77	0.73
[6]	Uses SVM; training and segmentation implemented within the same brain	Semi-automatic	0.86	0.77	0.73
[7]	Local structured prediction with CNN and k-means	Fully automatic	0.83	0.75	0.77
[8]	Two-pathway CNN for simultaneous local and global processing	Fully automatic	0.85	0.74	0.68

(continued)

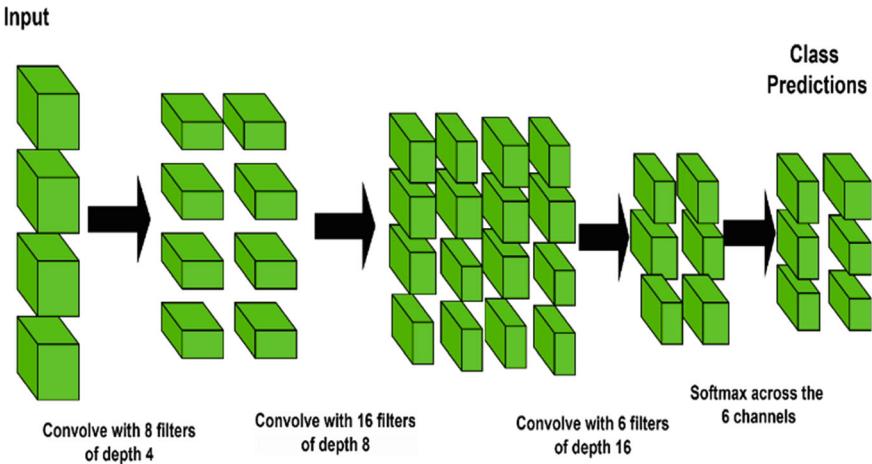
**Table 1** (continued)

Authors	Method	Level of user interaction	Accuracy for whole tumor	Accuracy for core tumor	Accuracy for active tumor
[9]	Generative model, uses cellular automata to obtain tumor probability map	Semi-automatic	0.72	0.57	0.59
[10]	Four CNNs, one for each modality, with their outputs concatenated as an input into a RF	Fully automatic	Not reported	Not reported	Not reported

**Fig. 4** Tumor segmentation on different FLAIR [9]

## 4 Deep Learning Methods

Researchers' interest in deep learning techniques has grown as a result of recent successes using Convolutional Neural Networks (CNNs) in tasks including biological image segmentation and object recognition. Unlike conventional techniques that depend on hand-crafted features, CNNs learn complex features by themselves from the data as shown in Fig. 5. Because of this property, research on brain tumor segmentation is now more concerned with creating efficient network topologies than with



**Fig. 5** Architecture for brain tumor segmentation in using CNN

extracting features through conventional image processing methods [9]. CNNs work by using patches that have been taken out of the images as inputs [19, 20].

The approach involves using multi-modality 3D patches. This input data includes spatial intensity information in three dimensions and it also adds to the computational burden of the network [21]. By using the CNN method to process both bigger and smaller patches at the same time, this technique allows brain MRI scans to be analyzed for both finer features and greater context. To address class imbalances, a two-phase training approach is implemented [22].

A method utilizing Convolutional Neural Networks (CNN) for local structured prediction is proposed in one approach. Following that, these labeling areas are arranged using the technique of k-means to generate a label patch glossary of dimension N [23, 24]. This method simplifies the classification process by first pre-grouping the label patches according to similarity, and then assigning input picture patches to one of these pre-defined clusters using a CNN [25] by extracting multi-planar patches surrounding each pixel. After concatenating the final hidden layer outputs from those CNNs, feature maps are utilized to train an RF classifier.

## 5 Summary of the Research

In this, various segmentation methods are reviewed for early brain tumor detection, with a focus on recent advancements in deep learning (DL). Brain tumor segmentation plays a pivotal role in the diagnosis and treatment planning process, directly impacting patient outcomes. Manual segmentation, although accurate, is time-consuming and reliant on individual expertise. Therefore, automated techniques,

particularly those employing deep learning algorithms, have gained significant attention due to their efficiency and accuracy [26].

We began by highlighting the significance of brain tumor segmentation and the conventional methodologies used. Manual segmentation, despite its drawbacks, remains a widely used method, especially for validating the results of automated techniques. Semi-automatic segmentation methods have emerged, allowing user interaction at different stages of the process, from initialization to result assessment [27, 28].

Fully automatic segmentation methods, on the other hand, eliminate the need for human intervention, utilizing various techniques such as thresholding, clustering, and deep learning [29]. However, automatic segmentation poses challenges, including the variability in tumor characteristics and the ambiguity of tumor borders, particularly in gliomas.

To evaluate segmentation techniques, datasets like BRATS and Oslo have been utilized, offering standardized benchmarks for comparison. These datasets enable objective assessment using metrics like the Dice Score, which measures the overlap between segmented and ground truth tumor regions.

## 6 Conclusion

Brain tumor segmentation is tough, but thanks to public datasets like BRATS, researchers can compare methods fairly. Deep learning has shown promise in accurately segmenting gliomas from MRI scans. Unlike traditional methods, CNNs can learn complex features directly from the images, making them more effective. By incorporating data from other imaging techniques like PET, MRS, and DTI, we can make these methods even better, hopefully leading to more reliable tumor segmentation for cancer diagnosis. This is the ideal region for deep learning, particularly when using convolutional neural networks (CNNs). CNNs have the ability to autonomously remove complex features from the multi-modal MRI pictures.

## References

1. Hamamci A et al (2012) Tumor-cut: segmentation of brain tumors on contrast enhanced MR images for radiosurgery applications. *IEEE Trans Med Imaging* 31(3):790–804
2. Menze B et al (2015) The multimodal brain tumor image segmentation benchmark (brats). *IEEE Trans Med Imaging* 34(10):1993–2024
3. Prastawa M, Bullitt E, Gerig G (2009) Simulation of brain tumors in MR images for evaluation of segmentation efficacy. *Med Image Anal* 13(2):297–311
4. Havaei M, Larochelle H, Poulin P, Jadoin PM (2016) Within-brain classification for brain tumor segmentation. *Int J Cars* 11:777–788
5. Emblem KE, Pinho MC, Zollner FG, Due-Tonnessen P, Hald JK, Schad LR, Meling TR, Rapalino O, Bjornerud A (2015) A generic support vector machine model for preoperative glioma survival associations. *Radiology* 275:228–234

6. Emblem KE, Due-Tonnessen P, Hald JK, Bjournerud A, Pinho MC, Scheie D, Schad LR, Meling TR, Zoellner FG (2014) Machine learning in preoperative glioma MRI: survival associations by perfusion-based support vector machine outperforms traditional MRI. *J Magn Reson Imaging* 40:47–54
7. Bakas S, Akbari H, Sotiras A, Bilello M, Rozycki M, Kirby JS, Freymann JB, Farahani K, Davatzikos C (2017) Advancing the cancer genome atlas glioma MRI collections with expert segmentation labels and radiomic features. *Sci Data* 4:170117
8. Bauer S, Wiest R, Nolte L, Reyes M (2013) A survey of MRI-based medical image analysis for brain tumor studies. *Phys Med Biol* 58:97–129
9. Liu J, Wang J, Wu F, Liu T, Pan Y (2014) A survey of MRI-based brain tumor segmentation methods. *Tsinghua Sci Technol* 19(6):578–595
10. Angelini ED, Clatz O, Mandonnet E, Konukoglu E, Capelle L, Duffau H (2007) Glioma dynamics and computational models: a review of segmentation, registration, and in silico growth algorithms and their clinical applications. *Curr Med Imaging* 3:262–276
11. Gordillo N, Montseny E, Sobrevilla P (2013) State of the art survey on MRI brain tumor segmentation. *Magn Reson Imaging* 31(8):1426–1438
12. Kwon D et al (2014) Combining generative models for multifocal glioma segmentation and registration. In: Medical image computing and computer-assisted intervention—MICCAI 2014. Springer, pp 763–770
13. Singh S, Ramkumar KR, Kukkar A (2024) Deep adaptive CHIONet: designing novel herd immunity prediction of COVID-19 pandemic using hybrid RNN with LSTM. *Multimedia Tools Appl* 83(10):29583–29615
14. Singh S, Ramkumar KR, Kukkar A (2023) Analysis and implementation of microsoft azure machine learning studio services with respect to machine learning algorithms. In: Modern electronics devices and communication systems: select proceedings of MEDCOM 2021. Springer Nature, Singapore, pp 91–106
15. Ifing T, Zahr NM, Sullivan EV, Pfefferbaum A (2009) The SRI24 multichannel atlas of normal adult human brain structure. *Hum Brain Mapp* 31:798–819
16. Singh S, Ramkumar KR (2022) Significance of machine learning algorithms to predict the growth and trend of COVID-19 pandemic. *ECS Trans* 107(1):5449
17. Tustison NJ, Cook PA, Klein A, Song G, Das SR, Duda JT, Kandel BM, van Strien N, Stone JR, Gee JC, Avants BB (2014) Large-scale evaluation of ANTs and FreeSurfer cortical thickness measurements. *Neuroimage* 99:166–179
18. Jégou S, Drozdzal M, Vazquez D, Romero A, Bengio Y (2017) The one hundred layers tiramisu: fully convolutional densenets for semantic segmentation. In: 2017 IEEE conference on computer vision and pattern recognition workshops (CVPRW). IEEE, Honolulu, HI, pp 1175–1183
19. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105
20. Singh S, Ramkumar KR, Kukkar A (2023) Pandemic outbreak prediction with an enhanced parameter optimisation algorithm using machine learning models. *Int J Electron Secur Digit Forensics* 15(4):359–386
21. Singh S, Ramkumar KR, Kukkar A (2021, October) Machine learning techniques and implementation of different ML algorithms. In: 2021 2nd global conference for advancement in technology (GCAT). IEEE, pp 1–6
22. Ciresan D et al (2012) Deep neural networks segment neuronal membranes in electron microscopy images. In: Advances in neural information processing systems, pp 2843–2851
23. Urban G et al (2014) Multi-modal brain tumor segmentation using deep convolutional neural networks. In: MICCAI multimodal brain tumor segmentation challenge (BraTS), pp 31–35
24. Havaei M, Davy A, Farley WD, Biard A, Courville A, Bengio Y, Pal C, Jadoin PM, Larochelle H (2016) Brain tumor segmentation with deep neural networks. *Med Image Anal.* <https://doi.org/10.1016/j.media.2016.05.004>
25. Davy A et al (2014) Brain tumor segmentation with deep neural networks. In: MICCAI multimodal brain tumor segmentation challenge (BraTS), pp 1–5

26. Pereira S, Pinto A, Alves V, Silva CA (2016) Brain tumor segmentation using convolutional neural networks in MRI images. *IEEE Trans Med Imaging* 35(5):1240–1251
27. Dvorak P, Menze B (2015) Structured prediction with convolutional neural networks for multi-modal brain tumor segmentation. In: MICCAI multimodal brain tumor segmentation challenge (BraTS), pp 13–24
28. Singh S, Ramkumar KR, Kukkar A (2024) Machine learning approach for data analysis and predicting coronavirus using COVID-19 India dataset. *Int J Bus Intell Data Mining* 24(1):47–73
29. Rao V, Sarabi MS, Jaiswal A (2015) Brain tumor segmentation with deep learning. In: MICCAI multimodal brain tumor segmentation challenge (BraTS), pp 56–59

# Salary Prediction Using Machine Learning Techniques



Pijush Ghorai and Rupashri Barik

**Abstract** In the world of professionals, figuring out how much someone should get paid can be tricky. This work goes into the complex world of salary prediction. This study presents a comprehensive approach to salary prediction using machine learning techniques, incorporating extensive data preprocessing and advanced model optimization. The dataset, derived from the Stack Overflow Developer Survey, includes variables such as Country, Age, Remote Work status, Education Level, Years of Professional Coding Experience, and Salary. Initial preprocessing involved filtering the dataset to include only full-time employed individuals, mapping countries based on frequency, randomizing age within specified ranges, and standardizing experience and education levels. Outlier salaries are adjusted to mitigate the impact of extreme values. Linear Regression, Decision Tree, and Random Forest are employed here to predict salaries. The Random Forest model demonstrated the best performance, with a Mean Absolute Error (MAE) of 17,166, a Root Mean Squared Error (RMSE) of 22,031, and an Accuracy of 81%. To further enhance the model performance, a unique optimization approach is applied, involving extensive hyperparameter tuning and cross-validation. This study underscores the importance of thorough data preprocessing and hyperparameter optimization in improving the accuracy of salary prediction models, providing valuable insights for both researchers and practitioners in the field of machine learning. This research bridges the gap between traditional salary models and the evolving landscape of employment, providing a valuable tool for organizations seeking precise and data-driven salary estimations.

**Keywords** Linear regression · Decision tree · Random forest · Hyperparameter tuning · Salary prediction

---

P. Ghorai · R. Barik (✉)  
JIS College of Engineering, Kalyani, WB, India  
e-mail: [rupashri.barik@jiscollege.ac.in](mailto:rupashri.barik@jiscollege.ac.in)

P. Ghorai  
e-mail: [pghorai098@gmail.com](mailto:pghorai098@gmail.com)

## 1 Introduction

In today's job market, understanding deserving salary worth is crucial for job seekers. Salaries vary across industries and employees often switch companies for better pay. Most previous salary prediction methods were too simple, focusing on just one or two factors like education or experience. To address this, a solution has been proposed that considers important features like experiences, qualifications, skills, location, and job demand. This comprehensive approach helps to negotiate fair salaries. Despite advancements, accurately predicting salaries remains challenging. Traditional models may not capture the intricate patterns that affect salaries. This study focuses on evaluating and optimizing three machine learning algorithms: Linear Regression, Decision Trees, and Random Forests. By tuning hyperparameters, it aims to enhance predictive capabilities. Using diverse datasets, this work aims for models that can generalize well across employment scenarios, providing more accurate salary estimates. Here the proposed approach includes a wide array of features that significantly impact salary outcomes, allowing for more nuanced predictions. Here, a comparative analysis of Linear Regression, Decision Trees, and Random Forests has been made highlighting their strengths and weaknesses in salary prediction. Here, by optimizing hyperparameters, improved performance tailored to salary datasets.

This work is motivated by the need for organizations to have precise salary estimation tools. As professionals contribute to employment data, the focus is to bridge the gap between traditional salary models and workforce demands. By incorporating advanced algorithms and diverse datasets, a comprehensive salary prediction approach has been done. Accurate salary estimations are crucial for talent acquisition and compensation strategies. This research aims to empower organizations with insights for informed decision-making in workforce management and contributes to refining salary prediction mechanisms, aiding organizations in talent management complexities.

## 2 Background Study

The field of salary prediction using machine learning algorithms has garnered considerable focus at the current time. Different approaches and methodologies are explored to improve the accuracy of salary predictions. This section illustrates some related work in this area, highlighting key findings and methodologies for salary estimation in today's dynamic job market.

## ***2.1 Traditional Salary Prediction Models***

Early efforts in salary prediction focused on simplistic models that primarily relied on factors such as education level, years of experience, and job title. Despite their simplicity, these models frequently proved inadequate in encompassing the intricate determinants that impact salary projections, consequently resulting in flawed estimations.

## ***2.2 Machine Learning-Based Approaches***

The latest developments in machine learning have enabled researchers to develop more sophisticated models for salary prediction. Algorithms such as Decision Trees, Linear Regression, and Random Forest have been widely used due to their ability to capture complex relationships in the data. Unlike traditional salary prediction models that rely on simplistic assumptions, these algorithms can analyze large amounts of data and identify subtle patterns that influence salary outcomes.

## ***2.3 Mutual Information Regression***

Mutual information regression is a method used for feature selection, particularly in regression tasks. It measures the dependency between two variables by figuring out how much the knowledge of one variable reduces the uncertainty of the other. In the context of feature selection for salary prediction, mutual information regression can be used to identify the most informative features that have a significant impact on predicting salaries.

## ***2.4 Multiple Linear Regression***

Multiple linear regression [1] is a supervised machine learning algorithm for predicting a continuing outcome, making it highly applicable in the context of salary prediction. In this presented work, it refers to an extension of the linear regression model to multiple independent variables.

## 2.5 Decision Trees

Decision trees are another popular algorithm for salary prediction. Recursively dividing the data into subsets according to the values of the input features is how decision trees operate. Decision trees are capable of capturing non-linear relationships in the data, making them more flexible than linear regression models.

## 2.6 Random Forest

Random forest algorithm is for ensemble learning that enhances prediction accuracy by integrating numerous decision trees. It works by training a large number of decision trees on random subsets and then averaging their predictions. Random forest is much effective for salary prediction by capturing complex relationships in the data.

## 2.7 Hyperparameter Tuning

Hyperparameter tuning is major for optimizing ML models' performance. These settings, external to the model, control learning and significantly affect performance. Grid Search exhaustively explores a hyperparameter grid to find the best combination, but it can be computationally expensive. Randomized Search randomly samples hyperparameter combinations are more efficient for large search spaces. For accurate tuning, combining grid search and randomized search is effective. Grid Search initially explores a broad hyperparameter grid and randomized search refines it for optimal values. This approach balances thoroughness and efficiency in hyperparameter tuning.

Lothe et al. [2] explored salary prediction using second-order polynomial transformation with linear regression. It achieved an MSE of 357 and 76% accuracy, meeting the target of an MSE below 360. However, it suggests further exploration of advanced ML techniques to improve accuracy, noting the challenge of adding parameters while maintaining accuracy. Ayua et al. [3] described the Salary Prediction Model for Non-Academic Staff using Polynomial Regression yielded promising results. The model achieved an R2 score of 97.2%, significantly higher than the previous 76%. However, a small dataset limits the understanding of model performance on larger datasets, especially where it causes frequent job switching among employees. Mukherjee et al. [4] predicted employee salaries based on their years of experience and hard work. The study highlights Linear Regression as an effective algorithm with 97% accuracy. However, it follows a traditional approach, using fewer features and only Linear Regression, which limits its capability for complex scenarios. Dutta et al. [5] proposed a Prediction Engine that uses decision tree and ensemble models to predict salaries based on key features, yielding encouraging and

precise results. Decision tree and random forest classifier provided the best accuracy scores. Challenges include dealing with noisy data and balancing social and financial aspects in salary prediction. Navyashree et al. [6] compared the performance of random forest, support vector regression, and decision tree for predicting employee salaries. Random forest achieved 97% accuracy, decision tree 85%, and support vector regression 90%. The system evaluates employees' years of experience against their annual salary, identifying the most effective data mining techniques for this task. Chen et al. [7] described a salary prediction model based on candidate resumes that utilizes multiple regression models and a stacking ensemble method. Random forest regression, decision tree and ridge regression are applied, comparing results based on RMSE and MAE. The limited dataset affects accuracy, indicating a need for further research to improve the model. Saeed et al. [8] analyzed biodata and employment outcomes of job seekers in India, building a salary prediction model using Naïve Bayes, random forest, and SVM. Naïve Bayes achieved the lowest RMS Error (0.305) and the highest accuracy among the algorithms used. Quan et al. [9] emphasized human resource analytics for data-driven decisions. Using CRISP-DM, exploratory data analysis, and decision tree algorithms, the Optimized HP Tree model achieved 57% accuracy, highlighting the personal skill development. Feng et al. [10] compared CNN and Random Forest for salary prediction using a dataset with age, working hours, and education. CNN outperformed Random Forest with an error rate of 0.0732 and variance of 0.1899 versus 0.2437 and 0.8285. Here, the challenges include data limitations and sample size issues, requiring better-quality datasets for improved accuracy. Das et al. [11] introduced a novel approach to predict salary using graphical representations and Linear Regression with Polynomial features. It focuses on potential salary growth but lack of complexity in data relationships and performance score discussion. Kuo et al. [12] focused on salary prediction using job title, work experience, and education level. It uses a stacked de-noising auto-encoder for pre-training weights and a fully connected neural network for prediction. The model, compared with SVM, SVR, logistic regression, random forest, ordinal regression, shows improved accuracy. Huang et al. [13] studied a dataset with age, work class, education, occupation, and more. It applies the Pearson Correlation Coefficient method to analyze variable relationships. Results indicate random forest outperforms decision tree and logistic regression, achieving 84.74% accuracy. Bansal et al. [14] made a comparison between Simple Linear Regression (SLR) and Multiple Linear Regression (MLR) for predicting employee salaries and house prices using Kaggle datasets. MLR outperforms SLR, with R-squared values of 0.67 versus 0.49 for house prices and 0.92 vs. 0.75 for salaries. Matbouli et al. [15] cover salary prediction using machine learning, highlighting artificial neural networks' superior performance over least squares method, support vector machine, and Gaussian process regression. Neural networks achieved an  $R^2$  improvement from 0.62 to 0.94, reducing errors by around 60%.

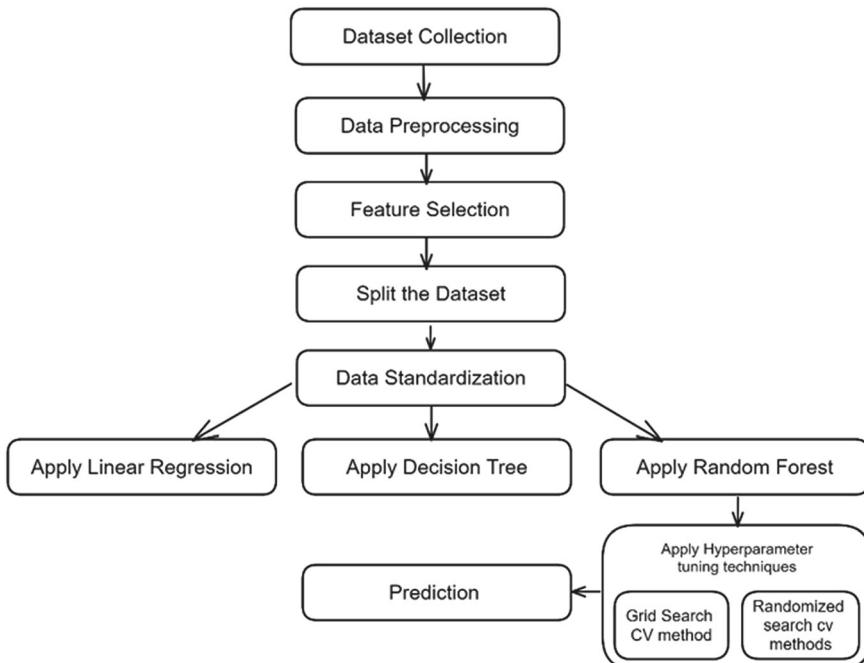
### 3 Proposed Methodology

The proposed methodology utilizes three machine learning algorithms for salary prediction: Linear Regression, Decision Trees, and Random Forests. These algorithms are chosen for their ability to capture complex relationships in the data and provide accurate predictions. Figure 1 illustrates the system flow of the proposed algorithm for salary prediction.

#### 3.1 Dataset Collection

Employee Salary Prediction Dataset [16] from Kaggle comprises 375 rows and 7 columns, containing information about employees, including their qualifications, experiences, job title, and salaries. It serves as a smaller, focused dataset for initial model training and evaluation.

Stack Overflow Developer Survey 2023 Dataset [17] is a larger dataset consisting of 48,019 rows and 84 columns, including a wide range of information about developers, such as their education, programming languages used, years of professional



**Fig. 1** System flow of the proposed working model

coding experience, job satisfaction, and salary. This dataset provides a more comprehensive and diverse set of features for training and evaluating the salary prediction models.

### ***3.2 Data Preprocessing***

The next step involves cleaning the data. This includes removing duplicate entries, standardizing the format of the data, and correcting any errors. Unused columns can be deleted at this stage. Missing values are filled using the pandas library that uses the last valid observation to fill gaps. Outliers are also addressed by grouping the Salary Dataset by “Education Level” and calculating the 25 and 75%. Salaries outside these thresholds are adjusted. Finally, the intermediate percentiles are dropped, leaving standardized salary data.

### ***3.3 Feature Selection***

Feature selection is important for an effective salary prediction model. The methodology starts with mutual information regression to find the most informative features. This method measures the dependency between variables to identify which ones are most relevant for predicting salary. After selecting the most informative features, they become the input variables (X), with salary as the output variable (y).

### ***3.4 Split the Dataset***

The dataset is split into training and testing set with 70% of the data used for training and 30% for testing. This allows for the evaluation of the model’s performance on unseen data and ensures that the model generalizes well to new instances.

### ***3.5 Data Standardization***

This step involves standardizing the numerical features in the dataset using the StandardScaler. Standardization ensures that each feature has a mean of 0 and a standard deviation of 1, which helps the machine learning models converge faster and makes them less sensitive to the scale of the features. Ensuring that the models can effectively learn from the data and make accurate predictions.

### ***3.6 Apply Machine Learning Algorithms***

After preprocessing and feature selection, different machine learning algorithms are applied. Multiple linear regression applied to predict a dependent variable (“Salary”) based on multiple independent variables (“Education Level”, “Years of Experience”, and “Total Skill”). It seeks a linear relationship between the inputs (x) and the output (y). Using sklearn’s linear regression class, fit the model with X and Y and visualize it. Decision tree model used to predict salary based on independent variables. DecisionTreeRegressor imported from sklearn.tree, create an instance, and fit it to the training data. Random forest algorithm predicts using an ensemble of decision trees. RandomForestRegressor imported from sklearn.ensemble, create an instance with dataset-specific parameters, and fit it to the data.

### ***3.7 Hyperparameter Tuning Techniques***

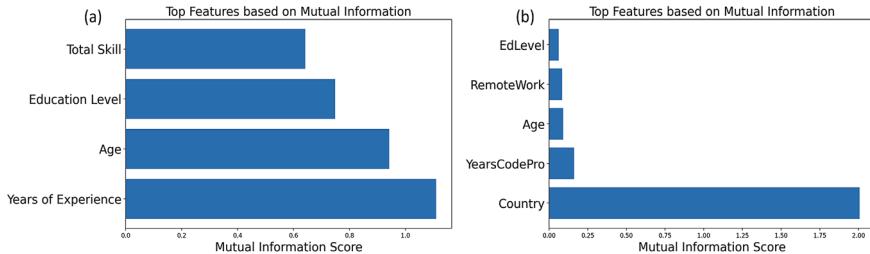
Then, hyperparameter tuning is applied to the Random Forest model using GridSearchCV [18] and RandomizedSearchCV [19] to find the best hyperparameters for the model. The tuned Random Forest model is then evaluated on the testing set using metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), and R-squared.

### ***3.8 Prediction Scores***

Once the model has been trained, it is used to make predictions on the testing set. The predictions are then compared to the actual values in the testing set to evaluate the model’s performance. The detailed result is discussed in the next section.

## **4 Results and Discussion**

This section explores salary prediction for professionals, aiming to provide a unified approach by leveraging diverse datasets. The model integrates Linear Regression and Random Forest algorithms for enhanced accuracy. This study contributes to refining salary prediction mechanisms and provides insights for organizations in talent acquisition and compensation strategies. The use of mutual information regression for feature selection improved the performance of the models by selecting the most informative features. This helped in reducing noise and focusing on the features that have a significant impact on salary prediction.

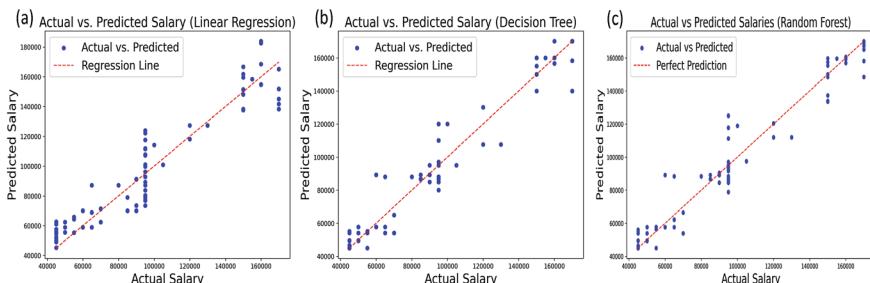


**Fig. 2** Mutual information of selected features dataset. **a** Employee salary prediction dataset and **b** Stack overflow developer survey 2023 dataset

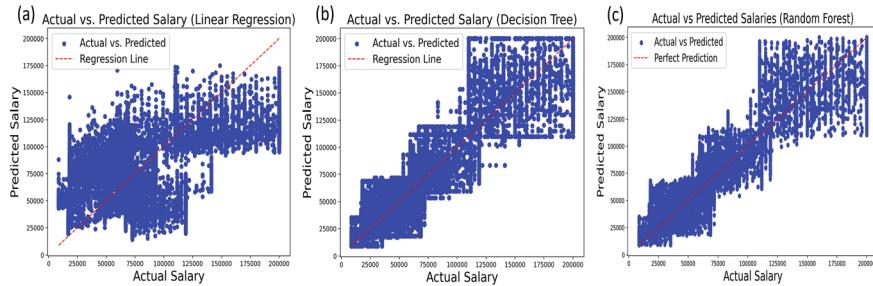
In Figure 2, the process ensures that the model is trained on the most relevant features, which can lead to improved prediction accuracy and generalization to unseen data.

Results from the Employee Salary Dataset show that Linear Regression achieved a high accuracy of 89.91%, with a MAE of 1535.90 and RMSE of 13954.13. The Decision Tree model improved on this with an accuracy of 95.45%, though it had higher errors. Random Forest, a more advanced ensemble method, further improved accuracy to 95.59%, with slightly higher errors MAE of 5768.54 and RMSE of (9225.50) compared to the Decision Tree. Grid search CV provided a marginal accuracy improvement (95.01%) over Random Forest, while Randomized search CV yielded the lowest accuracy (93.53%) among the grid search CV methods. Overall, Decision Tree and Random Forest outperformed Linear Regression, with Random Forest achieving the highest accuracy. Despite their computational complexity, these ensemble methods proved valuable in improving prediction accuracy. However, the gains from hyperparameter tuning via grid search CV were marginal.

Figure 3 visually confirms the superior performance of the Random Forest model in predicting employee salaries. The lower error metrics and higher Accuracy Score demonstrate its ability to capture complex relationships within the data, resulting in more accurate and reliable predictions.



**Fig. 3** Visualize of employee salary dataset. **a** Multiple linear regression, **b** decision tree and **c** random forest



**Fig. 4** Visualize results of stack overflow survey dataset. **a** Multiple linear regression, **b** decision tree and **c** random forest

Results from the Stack Overflow Survey Dataset show a similar pattern to the Employee Salary Dataset. Linear Regression performed poorly with an accuracy of 33.81%, MAE of 34632.64, and RMSE of 41975.21, indicating significant errors in prediction. The Decision Tree model improved accuracy to 74.32%, with an MAE of 18550.30 and RMSE of 26145.50, suggesting it captured some non-linear relationships better than linear regression. Random Forest further improved accuracy to 79.85%, with an MAE of 17239.12 and RMSE of 23160.45, likely due to its ability to handle non-linear relationships and reduce overfitting. Grid search CV yielded the highest accuracy at 81.15%, demonstrating the effectiveness of hyperparameter tuning. Randomized search CV performed similarly to grid search CV but was slightly less accurate. Overall, Decision Tree, Random Forest, and hyperparameter-tuned models performed significantly better than Linear Regression, emphasizing the importance of advanced algorithms in predicting salaries from complex datasets.

Figure 4 visually confirms the numeric findings, showing the enhanced predictive performance of the Random Forest model for the Stack Overflow Survey Dataset.

Tables 1 and 2 indicate that the decision tree and random forest algorithms consistently outperform linear regression in both datasets. The performance of the Random Forest model, especially after hyperparameter tuning with GridSearchCV, appears to be better compared to RandomizedSearchCV in both datasets. The choice between GridSearchCV and RandomizedSearchCV depends on the dataset size, the complexity of the hyperparameter space and the available computational resources. Overall, the Random Forest model, especially after hyperparameter tuning, showed the best performance among the algorithms tested. It achieved the lowest errors and highest Accuracy scores, demonstrating the importance of ensemble methods like Random Forest and hyperparameter tuning in improving prediction accuracy. These findings meet the objectives of developing an accurate salary prediction model.

**Table 1** The results of employee salary dataset

Algorithms	MAE	RMSE	Accuracy (%)
Multiple linear regression	1535.90	13954.13	89.91
Decision tree	5503.84	9371.47	95.45
Random forest	5768.54	9225.50	95.59
Grid search CV	6411.84	9812.19	95.01
Randomized search CV	8735.98	1178.58	93.53

**Table 2** The results of the stack overflow survey dataset

Algorithms	MAE	RMSE	Accuracy (%)
Multiple linear regression	34632.64	41975.21	33.81
Decision tree	18550.30	26145.50	74.32
Random forest	17239.12	23160.45	79.85
Grid search CV	17177.92	22400.49	81.15
Randomized search CV	18540.40	23092.59	79.97

## 4.1 Comparative Analysis

A comparative analysis of the proposed work with some current works has been presented in Table 3.

In Table 3, separately two data sets have been considered for the proposed model. The employee salary dataset and Stack Overflow Survey Dataset have been taken here.

**Table 3** Comparative analysis of a few existing models

Authors and year	Used model	Accuracy (%)
Lothe et al. [2]	Linear regression	96
	Polynomial transformation	76
Ayua et al. [3]	Polynomial regression	97.2
Mukherjee et al. [4]	Linear regression	97
Dutta et al. [5]	Decision tree classifier	84
	Random forest classifier	87
Navyashree et al. [6]	Decision tree regression	84
	Support vector regression	87
Saeed et al. [8]	Support vector machine	40
	Naïve Bayes	41
	Random forest	37
Huang et al. [13]	Random forest	84.74
	Decision tree	81.18
	Logistic regression	82.557

## 5 Conclusion

This work explored the complex world of employee salary prediction using machine learning algorithms. Through extensive experimentation and analysis, several key findings emerged: the algorithm performance decision trees and random forests consistently outperformed linear regression in predicting salaries. Their ability to capture complex relationships in the data and handle non-linearities makes them more suitable for this task. The use of mutual information regression for feature selection significantly improved the performance of the models. By selecting the most informative features, reduce noise and focus on the most relevant factors affecting salary prediction. Hyperparameter tuning, particularly using GridSearchCV, led to a significant improvement in the performance of the Random Forest model. Proposed works showed good generalization on unseen data, indicating their potential for real-world applications by leveraging advanced techniques in feature selection and hyperparameter tuning, accurate and reliable models that can assist organizations in making informed decisions related to compensation and talent acquisition strategies.

## References

1. <https://guhanesvar.medium.com/feature-selection-based-on-mutual-information-gain-for-classification-and-regression-d0f86ea5262a>
2. Lothe DM, Tiwari P, Patil N, Patil S, Patil V (2022) Salary prediction using machine learning. Int J Adv Sci Res Eng Trends. <https://doi.org/10.51319/2456-0774.2021.5.0047>
3. Ayua SL, Malgwi YM, Afrifa J (2023) Salary prediction model for non-academic staff using polynomial regression technique. In: Artificial intelligence and applications 2023, vol 00, no 00, pp 1–8. <https://doi.org/10.47852/bonviewAIA3202795>
4. Mukherjee T, Satyasaivanti SB (2022) Employee's salary prediction. Int J Adv Res Ideas Innov Technol 8(3):1357. <https://www.ijarit.com/manuscripts/v8i3/V8I3-1357.pdf>
5. Dutta S, Halder A, Dasgupta K (2018) Design of a novel prediction engine for predicting suitable salary for a job. In: 2018 fourth international conference on research in computational intelligence and communication networks (ICRCICN) 2018. <https://doi.org/10.1109/ICRCICN.2018.8718711>
6. Navyashree M, Navyashree MK, Neetu M, Pooja GR, Arun B (2019) Salary prediction in it job market. Int J Comput Sci Eng. E-ISSN: 2347-2693. <https://doi.org/10.26438/ijcse/v7si15.7884>
7. Chen Y, Li X (2023) Salary prediction based on the resumes of the candidates. CDEMS. <https://doi.org/10.1051/shsconf/202317003013>
8. Saeed AK, Abdullah PY, Tahir AT (2023) Salary prediction for computer engineering positions in India. J Appl Sci Technol Trends 04. <https://doi.org/10.38094/jastt401140>
9. Quan TZ, Raheem M (2023) Human resource analytics on data science employment based on specialized skill sets with salary prediction. Int J Data Sci 40–59. ISSN 2722-2039
10. Feng Z, Liu Z, Yin Y (2023) Comparison of deep-learning and conventional machine learning algorithms for salary. In: 3rd international conference on signal processing and machine learning. <https://doi.org/10.54254/2755-2721/6/20230910>
11. Das S, Barik R, Mukherjee A (2020) Salary prediction using regression techniques. In: Proceedings of industry interactive innovations in science, engineering & technology (I3SET2K19). <https://ssrn.com/abstract=3526707>

12. Kuo JY, Lin HC, Liu CH (2021) Building graduate salary grading prediction model based on deep learning. In: IASC2021. <https://doi.org/10.32604/iasc.2021.014437>
13. Huang Z (2023) Salary prediction with analyzing affected elements by using Pearson correlation. BCP Bus Manag 44
14. Bansal U, Narang A, Sachdeva A, Kashyap I, Panda SP (2021) Empirical analysis of regression techniques by house price and salary prediction. In: ICCRDA 2020, IOP conference on series: materials science and engineering. <https://doi.org/10.1088/1757-899X/1022/1/012110>
15. Matbouli YT, Alghamdi SM (2022) Statistical machine learning regression models for salary prediction featuring economy wide activities and occupations. Information 13:495. <https://doi.org/10.3390/info13100495>
16. Dataset Availability: “Employee Salary Prediction Data Set”. <https://www.kaggle.com/datasets/ahmadscientist/salary-prediction-data-set>
17. Stack Overflow Annual Developer Survey 2023. <https://survey.stackoverflow.co/>
18. Tune Hyperparameters with GridSearchCV. <https://www.analyticsvidhya.com/blog/2021/06/tune-hyperparameters-with-gridsearchcv/>
19. Hyperparameter Tuning Using Randomized Search. <https://www.analyticsvidhya.com/blog/2022/11/hyperparameter-tuning-using-randomized-search/>

# Tax Technology as a Catalyst for Globalization of Companies and Digital Transformation



Zornitsa Yordanova 

**Abstract** This paper explores the transformative role of tax technology in enabling the globalization of companies through digital transformation. By employing a qualitative research methodology, the study investigates how advanced tax technologies facilitate multinational organizations in navigating complex international tax landscapes, thereby enhancing their global operational capabilities. The findings underscore the pivotal role of tax technology in streamlining compliance processes, providing real-time data analytics, and supporting strategic tax planning across jurisdictions. Challenges such as organizational resistance, data security concerns, and regulatory complexities are also discussed, highlighting the need for integrated solutions that align with corporate cultures and structures. The paper further examines the implications of tax technology on corporate strategy, compliance, and risk management, proposing that effective utilization of digital tools in tax functions can significantly contribute to the competitive positioning of companies in the global marketplace. This study aims at providing valuable insights for policymakers, tax professionals, and business leaders seeking to leverage technology for enhanced global tax management and corporate expansion.

**Keywords** Tax technology · Digital transformation · Emerging technologies · Technology management · Globalization

## 1 Introduction

The rapid advancement of digital technologies has revolutionized various aspects of business operations, including tax management in corporations [1]. This article aims at exploring the role of tax technology as a facilitator for the globalization of companies, focusing on digital transformation from an organizational perspective. By leveraging a qualitative research approach, this study delves into the benefits, challenges, and implications associated with adopting tax technology in multinational

---

Z. Yordanova (✉)

University of National and World Economy, Sofia, Bulgaria

e-mail: [zornitsayordanova@unwe.bg](mailto:zornitsayordanova@unwe.bg)

organizations. It addresses a clearly identified problem in the scientific literature that points to international tax compliance and tax complexity as major barriers to internationalization and globalization [2]. The research findings highlight the significant impact of tax technology on enhancing the global reach of companies. Firstly, tax technology enables streamlined processes, increased accuracy, and reduced compliance risks, contributing to improved efficiency in tax management across borders [3]. Secondly, the adoption of tax technology facilitates real-time data access and analysis, empowering organizations to make informed tax decisions in an increasingly complex and dynamic global business environment [4].

However, the implementation of tax technology and digital transformation in general also present several challenges. These include organizational resistance to change, resource constraints, data security concerns, and regulatory complexities across jurisdictions [5]. Moreover, the paper discusses the importance of aligning digital transformation initiatives with the organizational culture and structure to foster the successful adoption and integration of tax technology into existing systems [6]. The implications of tax technology adoption extend beyond operational efficiency and compliance. It influences strategic decision-making, tax planning, and risk management, enabling companies to adapt to evolving global tax landscapes. Additionally, tax technology integration requires collaboration between tax and IT departments, highlighting the need for interdisciplinary teamwork within organizations [7].

Overall, this paper contributes to the growing body of knowledge on tax technology and digital transformation of corporate tax functions and their role in the globalization of companies. That topic has been recently also discussed by Chen, Xiao, and Jiang who found that digital transformation strengthens firms' competitive performance, thereby possibly further enhancing the willingness of enterprises to pursue digital transformation [8]. It provides insights into the benefits, challenges, and organizational implications associated with digital transformation in tax management. Policymakers, tax professionals, and organizational leaders can utilize these findings to develop effective strategies for leveraging tax technology as a means to enhance global competitiveness and navigate the complexities of international taxation in the digital era.

## 2 Theoretical Background: Digital Transformation Facilitates Globalization in Tax Management

### 2.1 *The Role of Digital Technologies in Tax Management*

Digital transformation, driven by advancements in technology, has revolutionized business practices across industries. The field of tax management is no exception, as organizations embrace digital solutions to streamline processes, enhance compliance, and optimize tax strategies. This article explores the implications of digital

transformation in tax management, shedding light on the benefits and challenges faced by organizations in this dynamic landscape.

## ***2.2 Automation and Efficiency Gains***

Digital technologies, such as robotic process automation (RPA) and artificial intelligence (AI), have automated repetitive tax processes, leading to increased efficiency, reduced errors, and enhanced productivity. These technologies enable organizations to process large volumes of data, perform complex calculations, and generate accurate tax reports in a fraction of the time [9].

## ***2.3 Real-Time Data and Analysis***

The adoption of digital technologies facilitates real-time access to financial and operational data, enabling organizations to make informed tax decisions. With cloud computing, organizations can store and analyze vast amounts of data, allowing for improved tax planning, risk assessment, and compliance monitoring. Real-time insights empower tax professionals to respond promptly to changing regulatory requirements and market dynamics [10].

## ***2.4 Improved Compliance and Risk Management***

Digital solutions provide organizations with tools to ensure compliance with complex tax regulations. Integrated tax management systems, coupled with advanced data analytics, enable organizations to identify potential risks and flag non-compliance issues promptly. Automated compliance checks and real-time monitoring reduce the likelihood of penalties and reputational damage [11].

## ***2.5 Globalization and Tax Management***

Globalization has fueled the expansion of multinational corporations (MNCs), cross-border trade, and investment flows, resulting in new challenges for taxation [12]. As businesses operate across national boundaries, tax systems need to adapt to address the complexities arising from globalization. The increasing mobility of capital, labor, and intellectual property has posed challenges for tax policy. Governments strive to maintain an attractive business environment while ensuring a fair and efficient tax system. They face the challenge of balancing the need for tax revenues with the

need to remain competitive in attracting investments. This has led to policy changes, including tax incentives, to attract MNCs and stimulate economic growth [13]. MNCs engage in sophisticated tax planning strategies to minimize their global tax burden. Transfer pricing, profit shifting, and tax havens are commonly employed tactics [14].

The complex nature of cross-border taxation has necessitated international cooperation to address issues such as double taxation, tax base erosion, and profit shifting. Efforts have been made to establish international tax standards and promote cooperation among countries [15]. Initiatives like the Base Erosion and Profit Shifting (BEPS) project led by the OECD aim to combat tax avoidance by enhancing transparency, aligning tax rules, and improving information sharing [16].

Globalization has both positive and negative impacts on developing countries' tax systems. On the one hand, increased foreign investment can stimulate economic growth and generate tax revenues. On the other hand, developing countries often face challenges in effectively taxing MNCs, leading to potential revenue losses and increased inequality [17]. Efforts are being made to enhance capacity building and promote fair tax practices to ensure developing countries can benefit from globalization [18].

Globalization and cross-border taxation present various challenges and controversies. One major challenge is the difficulty of accurately assessing and taxing the profits of MNCs operating in multiple jurisdictions [16]. Disputes over transfer pricing and allocating taxing rights between countries often arise. The digital economy further complicates matters as it enables businesses to operate remotely, leading to debates on the taxation of digital services and the establishment of a fair international tax framework [19].

### 3 Research Design

Given the comprehensive scope of our investigation, we performed two distinct analyses. The initial investigation primarily attempted to identify the primary obstacles and factors to consider, while the subsequent analysis sought to extrapolate potential future paths and prospects in the realm of digital transformation and tax administration within organizations. We performed a bibliometric analysis and subsequent literature review to further our understanding, focusing on the work of Ghasemzadeh et al. [20], who also explored a broad research domain with similar objectives.

#### 3.1 Data Sample

We retrieved the data source for addressing our study objectives from the Scopus database using the specified prompt:

(TITLE-ABS-KEY ("tax" OR "value added tax") AND TITLE-ABS-KEY ("internationalization" OR "globalization") AND TITLE-ABS-KEY ("technology" OR

"automation" OR "digital\*") AND (LIMIT-TO (DOCTYPE, "ar")) AND (LIMIT-TO (SRCTYPE, "j")) AND (LIMIT-TO (LANGUAGE, "English")).

Consequently, we acquired a total of 115 research articles that were utilized for both the bibliometric analysis and the literature evaluation in order to ensure consistency and the potential to reproduce the findings.

### ***3.2 Bibliometric Analysis***

This study utilizes bibliometric analysis to investigate the scope of digital transformation in tax systems regarding globalization and internationalization. We employ R Studio and the Biblioshiny package to conduct a systematic analysis of 115 articles obtained from the Scopus database. The articles are chosen based on particular keywords associated with taxation, digital technology, and globalization, utilizing the specified search string. The methodology entails the process of creating visual representations of the connections between writers who have collaborated on articles and the citations of those papers. This allows for the identification of authors who play a significant role, influential papers, and patterns of collaboration. In addition, we perform a content analysis to identify dominant themes and patterns across time, which aids in comprehending the development of research within the chosen field. This methodology is influenced by previous studies, such as the research conducted by Aria and Cuccurullo [21] on bibliometric practices and the study by Khaqqi et al. [22], which used bibliometric approaches to examine the incorporation of renewable energy sources into existing technology frameworks.

### ***3.3 Systematic Literature Review***

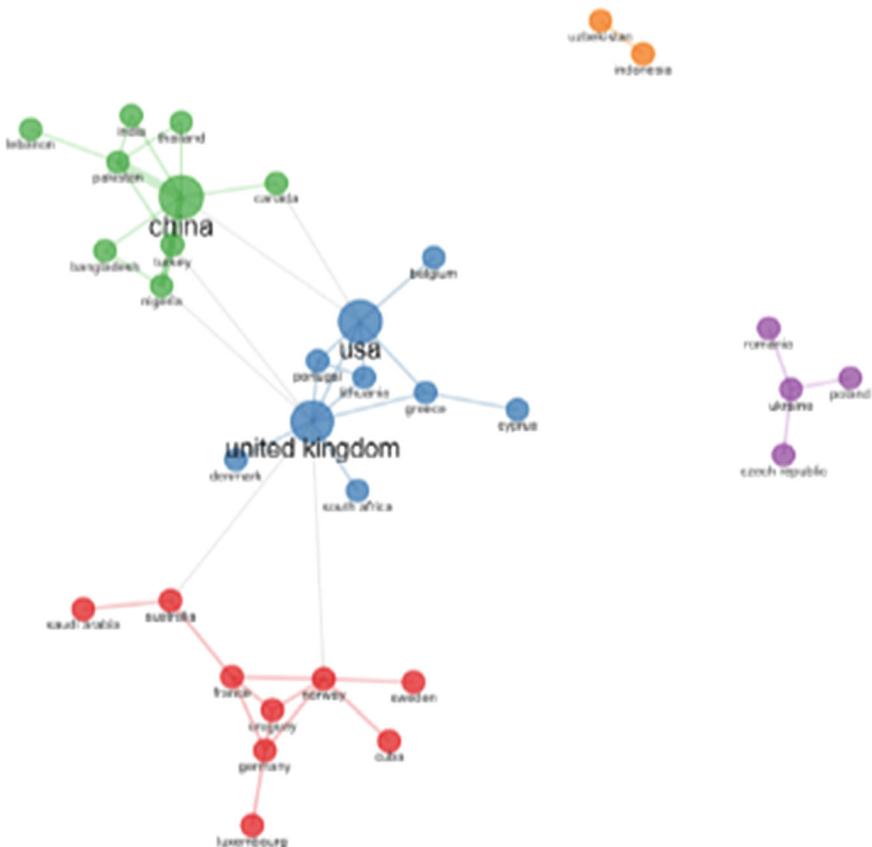
The literature review is guided by the findings from the bibliometric analysis. Using the same dataset, we focus on synthesizing the identified research themes to construct a comprehensive review of the field. The review methodology involves thematic analysis, where data extracted from the bibliometric study are categorized into themes such as challenges, advancements, and strategic implications of digital transformation in tax systems.

We draw on methodologies similar to those used in the studies by Roland Berger [23], who explored digital transformation strategies in financial services, and a recent survey by Bauer and Légaré [24] on digital innovation in tax administration. These references provide a framework for handling complex and voluminous data, enabling a detailed exploration of how digital technologies are reshaping tax practices globally.

## 4 Results and Discussion

### 4.1 Bibliometric Analysis

Figure 1 illustrates a coupling clustering based on the principal keywords of the research papers analyzed. The diagram reveals that although globalization, global trade, tax, and technologies are central themes within the dataset, the focus predominately gravitates toward specific countries. Notably, the majority of the research concentrates on tax-related and technology-related complexities in the United States, the United Kingdom, and China. A more nuanced examination suggests that the emphasis on these countries stems not merely from their tax or technological complexities but rather from their high volume of transactions, as they are pivotal hubs in the global trade network.



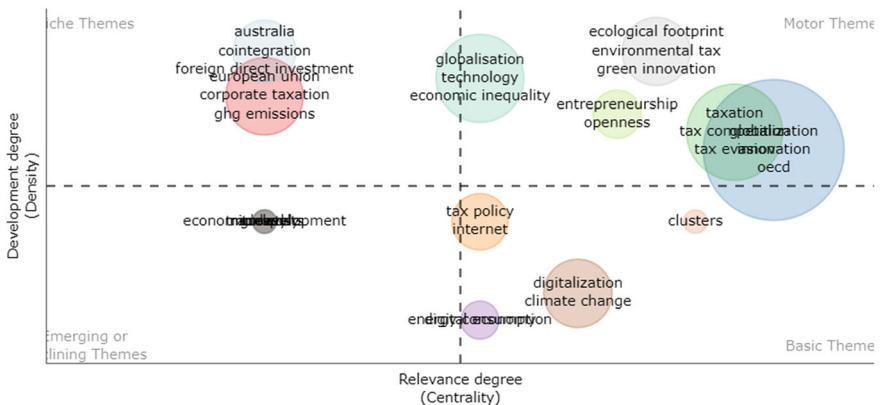
**Fig. 1** Coupling clustering

Globalization remains the primary focus of the research papers in this dataset. However, additional smaller clusters of research also emerge, encompassing themes such as innovation, economic development, environmental taxation, green innovation, and international political economy. These topics demonstrate interconnections at the metadata level with the overarching themes of globalization and tax technology initiatives (Fig. 2).

The subsequent figure highlights emerging scholarly interest in themes related to ecological issues, which are notably multidisciplinary and closely linked with taxation. While technology remains a relatively novel aspect within this discourse, digitalization has increasingly become central, particularly in discussions about climate change. However, corporate taxation emerges as an under-researched topic within the broader context. Consequently, there is a pressing need for focused research, especially on indirect taxes (Fig. 3).



**Fig. 2** Auto keywords clusters



**Fig. 3** Development degree of topics versus relevance

## 4.2 *Revealing the Potential for Tax Technology as a Catalyst for the Globalization of Companies and Digital Transformation*

### 4.2.1 Challenges and Considerations

**Data Security and Privacy:** The increased reliance on digital technologies raises concerns about data security and privacy. Organizations must implement robust cybersecurity measures to protect sensitive tax-related information from cyber threats. Compliance with data protection regulations, such as the General Data Protection Regulation (GDPR), is crucial to maintaining trust and avoiding legal repercussions [25].

**Skills and Talent Gap:** The successful adoption of digital tax solutions requires a skilled workforce. Organizations need tax professionals who possess a combination of tax expertise and technological proficiency. Bridging the skills and talent gap through training programs and recruitment strategies is essential for organizations to leverage the full potential of digital transformation in tax management [26].

**Regulatory and Legal Complexity:** Digital transformation in tax management introduces new challenges in navigating complex and evolving regulatory environments. Tax authorities are adapting their practices to monitor digital transactions, cross-border activities, and emerging business models. Organizations must stay abreast of regulatory changes to ensure compliance and avoid potential penalties or reputational risks.

### ***4.3 Future Directions and Opportunities***

**Advanced Analytics and Predictive Modeling:** The integration of advanced analytics and predictive modeling capabilities holds significant promise for tax management. By leveraging big data analytics, organizations can identify patterns, predict tax implications, and optimize tax strategies. Machine learning algorithms can assist in tax forecasting, scenario analysis, and identifying tax-saving opportunities.

**Blockchain Technology:** Blockchain technology has the potential to revolutionize tax processes by providing a secure and transparent platform for recording transactions. Smart contracts on blockchain can automate tax calculations, ensuring accurate and real-time reporting. Blockchain's immutability and auditability enhance tax compliance and minimize fraud risks.

**Collaboration and Integration:** Digital transformation in tax management requires collaboration between tax professionals, IT departments, and other relevant stakeholders. Effective integration of tax technology with existing systems and processes is crucial for seamless data flow and streamlined operations. Collaboration between different departments and external service providers enhances the effectiveness of tax technology implementation.

Tax technology can indeed act as a catalyst for the globalization of companies and digital transformation. Here are 7–9 critical ways and factors in which tax technology facilitates this process:

- **Streamlined Compliance:** Tax technology enables companies to navigate complex tax regulations across multiple jurisdictions efficiently. Automated compliance tools and digital tax management systems ensure accurate reporting, reducing the risk of non-compliance. This streamlined compliance process allows companies to expand their operations globally with confidence.
- **Real-time Reporting and Analysis:** Tax technology provides real-time access to financial data, facilitating timely reporting and analysis. Companies can monitor their tax positions across different countries, identify potential risks, and make informed decisions based on up-to-date information. Real-time reporting also enables companies to respond promptly to tax authorities' inquiries, reducing the risk of penalties and disputes.
- **Global Tax Planning:** Tax technology enhances global tax planning capabilities by providing tools for scenario analysis, tax forecasting, and optimization of tax strategies. Companies can model the tax implications of various business scenarios, assess the impact of international transactions, and identify tax-efficient structures. This proactive approach to tax planning helps companies optimize their global tax position and support their expansion plans.
- **Centralized Tax Data Management:** Tax technology allows companies to centralize and manage tax data efficiently. A unified tax data management system eliminates the need for manual data consolidation, reduces errors, and enhances data integrity. Companies can easily access and share tax-related information across different departments and jurisdictions, promoting collaboration and enabling effective global tax management.

- Automation of Routine Processes: Tax technology automates routine tax processes, freeing up valuable time for tax professionals to focus on more strategic initiatives. Automated data gathering, calculations, and document generation streamline tax compliance processes, minimizing manual errors and reducing administrative burden. This automation increases operational efficiency, especially when dealing with large volumes of tax-related data.
- Integration with Enterprise Systems: Integration between tax technology solutions and other enterprise systems, such as ERP (Enterprise Resource Planning) and financial systems, enables seamless data flow and enhances data accuracy. This integration eliminates manual data entry and reconciliation efforts, ensuring consistency between tax data and financial records. It also provides a holistic view of the company's financial and tax positions, supporting better decision-making.
- Enhanced Risk Management: Tax technology helps companies manage tax risks associated with global operations. Advanced analytics and risk assessment tools identify potential tax exposures, enabling companies to address issues proactively. By monitoring changes in tax laws and regulations, companies can adapt their tax strategies accordingly and minimize the risk of non-compliance or unexpected tax liabilities.
- Scalability and Flexibility: Tax technology solutions are designed to be scalable and adaptable to accommodate the changing needs of a global company. As companies expand their operations or enter new markets, tax technology can easily scale to handle increased tax complexities. Additionally, tax technology offers flexibility in accommodating local tax requirements, ensuring compliance with specific regulations in each jurisdiction.
- Collaboration and Communication: Tax technology facilitates collaboration and communication between tax teams, finance departments, and other stakeholders involved in global operations. Through digital platforms and centralized repositories, teams can share information, work on tax planning strategies, and collaborate on cross-border projects. Efficient communication and collaboration ensure alignment and coordination, supporting the globalization of companies.

These critical ways and factors demonstrate how tax technology acts as a catalyst for the globalization of companies and enables their digital transformation. By leveraging the power of tax technology, companies can navigate the complexities of global tax compliance, enhance tax planning strategies, and support their expansion plans effectively.

## 5 Conclusion

Digital transformation has brought significant advancements to the field of tax management. Organizations that embrace digital technologies and leverage them strategically can reap benefits such as increased efficiency, improved compliance, and enhanced decision-making capabilities. However, challenges related to data

security, skills development, and regulatory complexity must be addressed to maximize the potential of digital transformation in tax management. Looking ahead, continued innovation and collaboration will shape the future of tax technology, enabling organizations to navigate the evolving tax landscape with confidence.

**Acknowledgements** The paper is supported by the UNWE Research Program, project NID NI 4/2023

## References

1. Parviainen P, Tihinen M, Kääriäinen J, Teppola S (2017) Tackling the digitalization challenge: how to benefit from digitalization in practice. *Int J Inf Syst Proj Manag* 5(1):63–77
2. Shirokova G, Tsukanova T (2013) Impact of the domestic institutional environment on the degree of internationalization of SMEs in transition economies. *Int J Entrep Innov* 14(3):193–204
3. Alexander G (2022) Blocking the gap: the potential for blockchain technology to secure VAT compliance. *EC Tax Rev* 31(3)
4. Dobell J (2017) The future of tax technology is now. *Int'l Tax Rev* 28:41
5. Hinings B, Gegenhuber T, Greenwood R (2018) Digital innovation and transformation: an institutional perspective. *Inf Organ* 28(1):52–61
6. Fischer M, Imgrund F, Janiesch C, Winkelmann A (2020) Strategy archetypes for digital transformation: defining meta objectives using business process management. *Inf Manag* 57(5):103262
7. George G, Schillebeeckx SJ (2022) Digital transformation, sustainability, and purpose in the multinational enterprise. *J World Bus* 57(3):101326
8. Chen Z, Xiao Y, Jiang K (2023) The impact of tax reform on firms' digitalization in China. *Technol Forecast Soc Chang* 187:122196
9. Cooper LA, Holderness DK Jr, Sorensen TL, Wood DA (2019) Robotic process automation in public accounting. *Account Horiz* 33(4):15–35
10. Finch G, Goehring B, Marshall A (2017) The enticing promise of cognitive computing: high-value functional efficiencies and innovative enterprise capabilities. *Strategy & Leadership* 45(6):26–33
11. Von Solms J (2021) Integrating Regulatory Technology (RegTech) into the digital transformation of a bank Treasury. *J Bank Regul* 22:152–168
12. Dunning JH, Lundan SM (2008) Multinational enterprises and the global economy. Edward Elgar Publishing
13. Haudi H, Wijoyo H, Cahyono Y (2020) Analysis of most influential factors to attract foreign direct investment. *J Critical Rev* 7(13)
14. Yoo JS (2022) The effects of transfer pricing regulations on multinational income shifting. *Asia-Pac J Account Econ* 29(3):692–714
15. Sidik M (2022) Digital services tax: challenge of international cooperation for harmonization. *Jurnal Pajak Dan Bisnis (Journal of Tax and Business)* 3(1):56–64
16. Popescu CRG (2020) Sustainability assessment: does the OECD/G20 inclusive framework for BEPS (base erosion and profit shifting project) put an end to disputes over the recognition and measurement of intellectual capital? *Sustainability* 12(23):10004
17. Mpofu FY (2022) Taxation of the digital economy and direct digital service taxes: Opportunities, challenges, and implications for African countries. *Economies* 10(9):219
18. Harvie C (2019). Micro-, small-and medium-sized enterprises (MSMEs): challenges, opportunities and sustainability in East Asia. Trade logistics in landlocked and resource cursed Asian countries, pp155–174

19. Usman IMA, Saha TR (2022) An overview of tax challenges of digital economy. *Asia-Pacific J Manag Technol (AJMT)* 3(2):56–63
20. Ghasemzadeh K, Escobar O, Yordanova Z, Villasalero M (2022) User innovation rings the bell for new horizons in e-health: a bibliometric analysis. *Eur J Innov Manag* 25(6):656–686
21. Aria M, Cuccurullo C (2017) Bibliometrix: An R-tool for comprehensive science mapping analysis. *J Informet* 11(4):959–975. <https://doi.org/10.1016/j.joi.2017.08.007>
22. Khaqqi KA, Sikarwar VS, Lucquiaud M, Kelsall GH, Hellgardt K, Li J (2018) Incorporating negative externalities in the analysis of the energy system: a review of the integration of renewable energy in the UK electricity system. *Renew Energy* 120:399–410. <https://doi.org/10.1016/j.renene.2017.12.058>
23. Berger R (2016) Digital transformation of industries: digital transformation in financial services. Roland Berger Report. <https://www.rolandberger.com/en/Insights/Publications/Digital-transformation-of-industries.html>
24. Bauer K, Légaré F (2020) Recommendations for increasing the uptake of digital innovations in healthcare: a systematic review of the literature. *J Innov Health Inf* 27(2):1–10. <https://doi.org/10.14236/jhi.v27i2.1079>
25. Politou E, Alepis E, Patsakis C (2019) Profiling tax and financial behaviour with big data under the GDPR. *Comput Law Secur Rev* 35(3):306–329
26. Saeed M (2023) Digital services tax: impacts on multinational enterprises and transfer pricing adjustments. *Innovative Soc Sci J* 2(1)

# Sightless Fashion: Deep Learning Shopping Solutions



Clara Joseph and Sruthy Manmadhan

**Abstract** Fashion shopping can be particularly challenging for those with visual impairments, as the ability to visually assess clothing is typically crucial. The absence of accessible shopping experiences can result in dependence on others, limiting independence. To address this issue, “Sightless Fashion” introduces an innovative use of deep learning to improve the shopping journey for visually impaired individuals. Deep learning, a branch of artificial intelligence, has made significant strides in fields like computer vision and natural language processing. By harnessing these technologies, we can create systems that interpret visual data and comprehend text, bridging the gap between users and fashion items. This paper proposes a fresh approach that integrates cutting-edge deep learning models to provide visually impaired users with personalized and accessible fashion shopping experiences. The system outlined combines computer vision and natural language processing to enable users to interact with fashion items through non-visual cues like voice commands. By analyzing product descriptions, user preferences, and image characteristics, the system generates customized recommendations, empowering users to explore and select clothing items that suit their unique tastes and needs. This method not only promotes inclusivity in the fashion realm but also enhances independence and empowerment for individuals with visual impairments. The study’s method demonstrates an overall prediction accuracy of 89% for categorizing items and 78% for recognizing patterns. These findings underscore the system’s effectiveness in accurately identifying and suggesting fashion items.

**Keywords** FashionNet feature Extraction · Convolutional neural networks (CNN) · Long Short-Term Memory (LSTM)

---

C. Joseph (✉) · S. Manmadhan

Department of Computer Science, NSS College of Engineering, Palakkad, Kerala, India  
e-mail: [clarajoseph0307@gmail.com](mailto:clarajoseph0307@gmail.com)

## 1 Introduction

In an era dominated by visual content and online shopping experiences, individuals with visual impairments encounter substantial challenges in accessing information about clothing items, limiting their ability to engage fully in the dynamic realm of fashion. Choosing clothing with intricate patterns and colors can be particularly difficult for those who are blind or visually challenged. While these individuals often use their hands to identify items by texture, visual data such as patterns and colors remain inaccessible. Consequently, coordinating clothing without the assistance of a sighted person is challenging.

Recognizing the gap in accessibility, our research introduces a method aimed at revolutionizing accessibility for visually impaired individuals in the fashion domain. By leveraging advanced technologies such as image recognition and natural language processing, our approach seeks to generate detailed and meaningful captions for clothing images. This innovative solution addresses the limitations of conventional text-based descriptions, empowering individuals with visual impairments to make informed and independent decisions about their clothing choices. Through this research, we aim to bridge the accessibility divide, ensuring that everyone, regardless of visual ability, can partake in the diverse and expressive world of fashion.

The manuscript is structured as follows: Introduction outlines the relevance and significance of the study, including the problem statement and objectives. Literature Review presents a comparison of existing methods and approaches, highlighting the contributions and limitations of previous studies. Proposed Work describes the proposed system, including a detailed flowchart of the methodology, the algorithms used, and the overall workflow. Dataset and Methodology provides details about the dataset, including its source, characteristics, and preprocessing steps, along with the methodology used for training and evaluation. Results and Discussion discusses the experimental results, including accuracy metrics and performance evaluation, and compares these with existing methods. Conclusion summarizes the findings, emphasizes the contributions of the study, and suggests directions for future research.

## 2 Related Works

Convolutional Neural Networks (CNNs), an artificial intelligence method [1] were utilized in the approach to effectively classify and identify stains on clothing items. The effectiveness of the technique was demonstrated by achieving an impressive F1 score of 91% in stain identification through the fine-tuning of a deep learning object identifier, namely the region-based CNN. This work highlights the potential of AI-based frameworks to aid visually impaired individuals in identifying clothing and managing stains. However, limitations were encountered in accurately identifying outfit types due to over-classification and biases in the models' attire. To enhance the stain detection dataset, these restrictions need to be addressed by adding additional

data and reducing the number of outfit type identification categories. Despite these obstacles, the study presents a viable way to enable visually impaired individuals to choose clothing and manage stains, potentially improving their quality of life on a daily basis.

The method introduces the Vision4All concept, [2] leveraging the ResNet34 architecture to enhance deep learning models such as FashionNet for recognizing garment attributes. Additionally, it incorporates a text-based interaction module, enabling users to inquire about clothing characteristics, with these queries being processed through similarity analysis and word connections. Among its contributions is a proposal for a comprehensive framework that utilizes both textual and visual cues to assist visually impaired individuals in identifying various clothing qualities, including categories and styles. Furthermore, it demonstrates how voice-based interaction can be combined with deep learning models to simplify information acquisition about apparel for consumers. However, the method's reliance on pre-existing datasets and word embeddings for text similarity analysis may hinder the system's ability to understand and respond effectively to unique or less common queries. It is concluded that additional validation and improvement are necessary to ensure the system's accuracy and resilience in real-world scenarios. Despite these challenges, the suggested method offers a new avenue for visually impaired individuals to access clothing information through voice-activated communication, potentially enhancing their independence and confidence in daily activities (Table 1).

The research focuses on utilizing deep neural networks to provide customized outfit recommendations based on each user's specific fashion tastes. In order to do this [3], the study makes use of deep networks. Specifically, it uses a feature network to extract pertinent data from clothing items and a matching network to determine how similar the products are to the user's tastes. Although the intended audience consists mainly of people looking for customized outfit recommendations which may include people who are visually impaired, the evaluation of the paper focuses on how well these recommendations work as well as how well the deep neural networks perform in identifying user preferences. One significant drawback, as the article points out, is that it could be difficult to extrapolate the findings to a wide range of user tastes and fashion sense. The study also emphasizes the need for more testing and improvement across a wider range of user demographics and fashion preferences in order to improve the model's resilience and practicality.

**Table 1** Summary table

Study	Results
Stain identification using CNNs	F1 score of 91% for stain identification
Vision4All concept	Proposed framework for garment attribute recognition and voice-based interaction
Customized outfit recommendations	Effective in identifying user preferences and providing recommendations

### 3 Methodology

The system model encompasses a complete pipeline for categorizing fashion images. Figs. 1 and 2 shown below detail the proposed method. Initially, fashion photos are sourced from a dataset stored in a CSV file. Utilizing the Pandas library, relevant data, including picture URLs and classifications, is extracted from the dataset. Subsequently, these images are organized into a structured dataset folder based on their respective categories, facilitating data management and accessibility. At the core of the system lies a Convolutional Neural Network (CNN) architecture, constructed with the assistance of the Keras toolkit. This CNN design comprises convolutional layers for feature extraction, max-pooling layers for spatial downsampling, and dropout layers to mitigate overfitting. Furthermore, fully connected dense layers are incorporated in the classification process, aiding the model in recognizing patterns and predicting outcomes. Subsequently, the model is trained on augmented data by minimizing the loss on the training set, adjusting its weights and biases using the binary cross-entropy loss function and the Adam optimizer. Following training, a separate test dataset is utilized to evaluate the model's performance in unfamiliar scenarios, yielding metrics like test loss and accuracy. Lastly, to ensure the preservation and future deployment of the trained model, the architecture and weights are stored to files. In essence, by integrating data management, model development, training, evaluation, and storage functionalities, this system model provides a comprehensive framework for fashion image classification. Convolutional neural networks (CNNs) would be used in the model to extract features from images, and then LSTM layers would be used to analyze the sequential data. Based on the characteristics they had retrieved, the LSTM layers would learn during training to provide captions for the input images.

#### 3.1 *Data Collection and Preprocessing*

Image of apparel are collected from a dataset called “dress patterns.csv,” which contains information about different types of clothing like tops, dresses, and trousers. The dataset undergoes preprocessing to extract relevant data such as image URLs and classifications. These images are then organized into subfolders for each clothing category within a structured dataset folder named “dataset\_category.” This folder structure simplifies the preparation and loading of data for training models. Data preparation steps include resizing images to a uniform size, converting them to a standard format, and splitting them into training and testing sets.

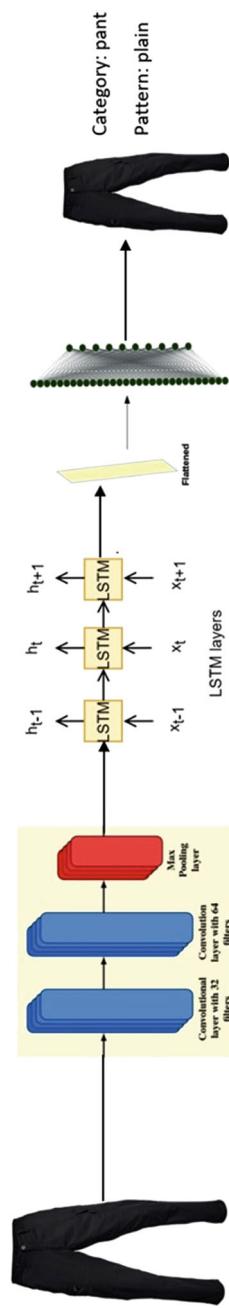
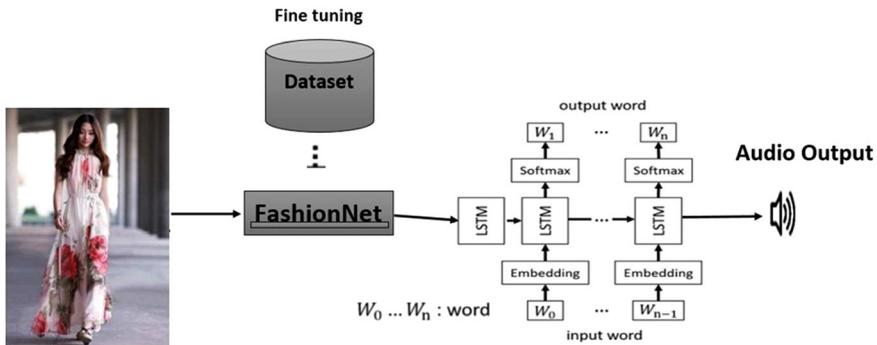


Fig. 1 Fashionnet, LSTM to generate description



**Fig. 2** Architecture

### 3.2 *FashionNet Feature Extraction*

A convolutional neural network (CNN) that has been specially built and trained for fashion-related image identification tasks is used in FashionNet Feature Extraction.

For feature extraction, FashionNet, like many other convolutional neural networks (CNNs), usually uses a sequence of convolutional layers followed by maxpooling layers. The network is able to recognize several low-level visual patterns, such as edges, textures, and colors, to these convolutional layers, which apply a set of learnable filters to the input fashion photos. FashionNet is a resource designed to assist individuals with vision impairments in generating subtitles by categorizing clothing through advanced feature extraction algorithms. It accurately identifies clothing items by analyzing a range of visual attributes such as color, pattern, and texture. Each succeeding convolutional layer builds on the representations that the preceding layers have learnt, extracting ever-more-abstract and sophisticated characteristics from the input data as it moves through the network. The feature maps are downsampled, lowering their spatial dimensions while keeping the most crucial information, via the max-pooling layers inserted in between the convolutional layers. High-level representations of the input photos are learned by FashionNet through this hierarchical feature extraction method, which is crucial for tasks like fashion categorization and attribute prediction.

### 3.3 *Sequence Processing*

The Sequence Processing Module plays a pivotal role in generating captions for fashion images by leveraging Long Short-Term Memory (LSTM) networks. The utilization of sequence processing aids in identifying textile groups, simplifying the task of generating captions for individuals with visual impairments. The system employs sophisticated algorithms and feature extraction techniques to thoroughly

analyze a range of visual characteristics for effective categorization of clothing items. Subsequently, this information is harnessed to craft descriptive captions, enhancing the ability of visually impaired individuals to comprehend and appreciate different clothing articles. Specifically designed for captioning fashion images, this module utilizes the visual features extracted by the feature extraction module and analyzes them sequentially to generate contextually appropriate captions. The embedding layers within the Sequence Processing Module convert discrete words in the captions into continuous vector representations, enabling the LSTM network to process them effectively. Additionally, various attention mechanisms may be incorporated to selectively focus on different regions of the image when generating each word in the caption, thereby enhancing the model's ability to align textual descriptions with visual attributes. In summary, the Sequence Processing Module utilizes LSTM networks to decode visual features extracted from fashion photographs, producing detailed and contextually accurate captions that capture the essence and content of the images.

### ***3.4 Training***

The Training Module is integral to the development of the fashion image captioning model, guiding the learning process to generate precise and contextually relevant captions. It utilizes either random or pre-trained weights for both the feature extraction (CNN) and sequence processing (LSTM) components. Leveraging the CNN, it extracts visual features from images, while the LSTM network is employed to craft captions. Through an optimization approach akin to Adam, the module adjusts model parameters to minimize loss and enhance caption production accuracy. Continuous performance tracking via periodic evaluations on the validation set ensures model refinement. Once trained, the model is stored on disk for future use in captioning new fashion photos. Ultimately, the Training Module plays a pivotal role in enhancing the model's ability to produce meaningful and coherent descriptions for fashion photos through its iterative optimization process. The further processing to detect patterns and category are dealt in Sect. 4.

## **4 Modules**

### ***4.1 Cloth Category Determination***

In the context of the provided labels dictionary, class category prediction involves determining the class or category to which an input belongs, specifically identifying the type of clothing item depicted in an input image. The model generates a numerical index representing its prediction, as each numerical index in the labels dictionary in Fig. 3 corresponds to a specific apparel category. During the prediction phase, the

```

labels = {
    0: 'dress',
    1: 'hat',
    2: 'longsleeve',
    3: 'outwear',
    4: 'pants',
    5: 'shirt',
    6: 'shoes',
    7: 'shorts',
    8: 'skirt',
    9: 't-shirt'
}

```



The given image is a t-shirt

**Fig. 3** Category labels

trained model analyzes the input image, producing a probability distribution across various apparel categories. Each category is assigned a probability by the model, indicating the likelihood that the input image belongs to that category. Subsequently, the predicted category for the input image is determined based on the category with the highest probability. Leveraging the labels dictionary, the predicted index can be mapped back to the label associated with the corresponding apparel category. This mapping provides valuable insights into the type of clothing item depicted in the input image, enabling the interpretation of the model's prediction in terms of easily understandable clothing categories.

## 4.2 Clothing Pattern Prediction

Predicting patterns involves identifying recurring visual patterns within an image, particularly in the context of clothing. Typically, the initial step in this process is preprocessing the image to extract relevant characteristics corresponding to specific patterns. Each cluster formed by grouping similar pixels based on color represents a dominant color in the image. This method determines the dominant colors by assigning each pixel to the nearest cluster centroid after specifying the desired number of clusters (or colors).

Subsequently, the next step shown as Fig. 4 is to associate the dominant colors with named colors that closely match them. This association is achieved by comparing the RGB values of the dominant colors with those in a predefined dictionary of named colors. The closest\_color function selects the named color closest to each dominant color by calculating the Euclidean distance between the RGB values of each dominant color and those of the named colors. Through this approach, each dominant color is efficiently labeled based on its nearest color.



**Fig. 4** Color prediction

The resulting named colors, obtained from mapping the prominent colors to their corresponding hues, describe the visual patterns present in the image. This provides a practical and understandable method for comprehending and explaining the primary visual elements inherent in the clothing items depicted in the image.

### 4.3 *Speech Conversion*

The Text-to-Speech module is essential for ensuring accessibility for visually impaired users as it generates spoken captions. When fashion product descriptions are created, Text-to-Speech technology steps in to convert these written descriptions into spoken words. This functionality enables individuals with visual impairments to access and understand details about fashion items without primarily relying on visual cues. Using sophisticated algorithms, Text-to-Speech technology replicates human speech with natural intonation and rhythm. This allows visually impaired individuals to engage in an immersive experience where they can perceive and comprehend descriptions of fashion items [4–6]. Ultimately, Text-to-Speech functionality enhances the browsing or shopping experience for people with visual impairments by enabling them to interact meaningfully with visual content.



The given image is a red t-shirt with plain pattern



The given image is a black pant with plain pattern

**Fig. 5** Result

## 5 Results and Discussion

### 5.1 Dataset Generalization

To ensure compatibility and optimal performance, several crucial steps are involved in preparing the dataset for FashionNet and LSTM models. This process begins with assembling a diverse collection of preprocessed fashion photos accompanied by descriptions. These photos are then enhanced for robustness and diversity through supplementation, normalization, and scaling. The textual captions are tokenized, and each token is assigned a unique integer index to prepare them for input into the LSTM model [7–10].

FashionNet, a specialized convolutional neural network, extracts rich visual features from the preprocessed images. These features, along with tokenized caption sequences, are used to train the LSTM model for caption generation. Model performance is evaluated using various metrics, and adjustments are made to enhance caption coherence and accuracy.

In the experiment, 15,000 photos were given to trained model and corresponding outputs were analyzed and produced. As seen in Fig. 5, the model shows an accuracy of 89% in category prediction and 78% in pattern prediction. Some outputs show descriptions about the features of the clothes and patterns. This result indicates good learning as it implies a strong relationship between the produced words and the visual attributes.

## 6 Conclusion

In summary, the utilization of FashionNet and LSTM for generating captions shows significant promise in enhancing accessibility for individuals with visual impairments. By combining FashionNet's robust visual feature extraction capabilities from fashion images with LSTM's expertise in natural language processing, the model can effortlessly produce descriptive captions for fashion items. The generated captions

are converted to speech using Text-to-Speech technology, allowing visually impaired users to listen to the descriptions of fashion items. This technology enables individuals who are blind or visually impaired to access and comprehend fashion-related content, thereby enhancing their engagement with fashion media and shopping experiences. However, continuous improvement and optimization of the model are necessary to ensure accurate and coherent caption generation. Additionally, refining and customizing the system to meet the specific needs of visually impaired users through usability testing and user feedback is essential. Ultimately, the integration of FashionNet and LSTM represents a significant advancement in leveraging AI technology to enhance accessibility and inclusion in the fashion industry.

## References

1. Rocha D, Soares F, Oliveira E, Carvalho V (2023) Blind people: clothing category classification and stain detection using transfer learning. *Appl Sci* 13(3)
2. Khalid L, Gong W (2022) Vision4All—a deep learning fashion assistance solution for blinds. In: 2022 5th international conference on artificial intelligence and big data (ICAIBD). IEEE, pp 156–161
3. He T, Hu Y (2020) FashionNet: personalized outfit recommendation with deepneural network. [arXiv:1810.02443](https://arxiv.org/abs/1810.02443)
4. Guravaiah K, Bhavadeesh YS, Shwejan P, Vardhan AH, Lavanya S (2023) Third eye: object recognition and speech generation for visually impaired. *Procedia Comput Sci* 218:1144–1155
5. Chang YH, Zhang YY (2022) Deep learning for clothing style recognitionusing YOLOv5. *Micromachines* 13(10)
6. Yang X, Yuan S, Tian Y (2021) Assistive clothing pattern recognition for visually impaired people. *IEEE Trans Hum-Mach Syst* 44(2):234–243
7. Liu Z, Luo P, Qiu S, Wang X, Tang X (2020) Deepfashion: powering robustclothes recognition and retrieval with rich annotations. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1096–1104
8. Tateno K, Takagi N, Sawai K, Masuta H, Motoyoshi T (2020) Method for generating captions for clothing images to support visually impaired people. In: 2020 Joint 11th international conference on soft computing and intelligent systems and 21st international symposium on advanced intelligent systems (SCIS-ISIS). IEEE, pp 1–5
9. Yang X, Zhang H, Jin D, Liu Y, Wu CH, Tan J, ... Wang X (2020) Fashion captioning: Towards generating accurate descriptions with semantic rewards. In: Computer vision–ECCV 2020: 16th European conference, Glasgow, UK, 23–28 Aug 2020, Proceedings, Part XIII 16. Springer International Publishing, pp 1–17
10. Joshi S, Mathur A, Dhabai H (2020) A review: recognizing clothes patternsand colours for blind people using neural network. *Int J Adv Res Comput Sci* 8

# Handwritten Signature Verification and Forgery Detection using Deep Learning



**Harsh Vardhan, Gaurav Kumar Gautam, Harshit Gupta, and Rahul Katarya**

**Abstract** This paper studies handwritten signature verification and the authenticity of a given signature. We have investigated the potential of ensemble model among other machine learning methods for application in the verification of signatures. The purpose of this study is to develop a web-based software that can authenticate the user's signature, either genuine or forged. Comparison of the results achieved by our model with previously established models in the scope of our work on the basis of classification metrics, viz. accuracy, precision, recall, and F1-score. For the development of web application, Python, Django, TensorFlow, and React frameworks have been used. To get approved access, users will have to register themselves and log in using the web app features granted by the system. The validity of the signature is being determined by analyzing the user's writing features, such as the length, width, and depth of each stroke. The ability to verify signatures more accurately over time will be enhanced using machine learning algorithms in the application.

**Keywords** Convolutional neural networks (CNN) · Deep learning · Forged signature · Handwritten signature · Signature verification

## 1 Introduction

The authentication of handwritten signatures is a crucial step in many situations. Various processes got digitalized due to digital revolution but traditional signatures are still required in a lot of places, such as check payments, government offices, confirming one's identity, financial services, and legal documentation. The advancement of machine learning has made automatic signature verification possible, which has greatly improved the efficiency, precision, and dependability of the process. A machine learning-based web application can be developed by developers to verify the authenticity of a signature. The program can examine a picture of the signature and the application can determine whether it is genuine or forged.

---

H. Vardhan · G. K. Gautam (✉) · H. Gupta · R. Katarya  
Delhi Technological University, New Delhi, India

Machine learning models can be trained using large datasets of signatures. Using different features such as size, shape, and stroke patterns, models can be taught to recognize a genuine signature. These can also be designed to provide intuitive programs for users to upload signature photos and receive results almost immediately after submission. This paper will investigate one of many possible machine learning techniques applicable to handwritten signature validation. In these discussions, we will discuss how to test a model and different approaches in feature extraction, preprocessing strategy, assessment metrics, and others. Finally, we will launch the web app with machine learning capabilities to analyze an image of a signature and conclude if it is genuine or forged. The following are our major contributions:

- To develop a web-based software that can authenticate the user's signature, either genuine or forged.
- For the development of web application, Python, Django, TensorFlow, and React framework has been used.
- Comparison of the results achieved by our model with previously established models in the scope of our work on the basis of classification metrics viz. accuracy, precision, recall, and F1-score.

## 2 Related Work

The significance of handwritten signatures in biometric systems as personal verifiers has attracted researchers from a wide range of institutions and organizations to the field of signature verification [1]. The verification model's performance is influenced by the set of attributes that are applied to it. Offline signature verification has been extensively worked on, using many feature sets to run the model. Most of the works share geometric data, topology, gradient, structural data, and concavity bases [2]. Research on offline handwritten signature recognition began in the early nineties. Many alternative approaches to solving this problem have been developed since then. It might be difficult to verify someone's identity using a handwritten signature if someone unauthorized tries to replicate it [3].

The dynamic information of signature writing process is lost in the offline (static) signature verification process, making it difficult to create a high-quality feature extractor that can differentiate between expertly faked and real signatures. The presented approach uses both global and local features to verify online handwritten signatures automatically. Different facets of the signature's form and creation dynamics are captured by the global and local elements, showing that signature verification performance can be greatly enhanced by supplementing global features with a local feature based on the signature likelihood acquired via HMM [4]. The current build of the software has an error rate of 2.5%. The false acceptance (FA) rate was decreased from 13% to 5% when local information was added to the algorithm using only global features at the 1% FR threshold.

There are two main types of signature verifications: static and dynamic. While static (offline) verifications take place after the fact when confirming an electronic or

paper signature, dynamic (online) verifications take place when a person signs on a digital tablet or other such devices [12]. When dealing with a high number of papers, offline signature verifications are inefficient and time-consuming. There has been a rise in the use of fingerprinting, iris scanning, and other forms of online biometric personal verification as a means to circumvent the problems associated with offline signature verification [13]. This study developed a Python-based CNN model for offline signature validation, and its testing accuracy reached 99.70% after training and validation.

The most precise and trustworthy signature identification and verification system has been the subject of extensive study [14]. Both of those issues are investigated in this work. The study's major objective is to identify the optimal algorithms for signature recognition in terms of signature type. A PRISMA flowchart was used to organize this comprehensive literature evaluation. The findings point to the usage of Convolutional Neural Networks (CNN) for the recognition of offline signatures, whereas Recurrent Neural Networks (RNN) of various architectures are used for online signatures [15]. The implementation of deep learning artificial neural networks has led to advancements in this field.

Due to factors including the large number of authors, the small number of training samples per writer, the high intra-class variability, and the significantly skewed class distributions, offline handwritten signature verification (HSV) presents several difficulties [16]. A writer-independent (WI) paradigm gives another feasible approach to resolving these challenges. By applying the dichotomy transformation to construct a dissimilarity space, a single model in WI systems can validate the signatures of all authors. This system can be scaled to meet these issues, and it's also simple to implement and manage new writers, making it suitable for usage in a transfer learning setting.

Another approach that has been presented is using wavelets for offline approach to authenticate handwritten signatures. Determining whether a signature is genuine or fake can be done by identifying useful and common features within different signatures of same person. In the first step of this process, a closed-contour tracing algorithm is required. Wavelet transforms are used to decompose the curvature information of the tracked closed contours into multiresolution signals [20]. In the next step, it takes the curvature information and uses the zero-crossings that occur there as a matching feature. Apart from this, a statistical metric is developed to decide which of a writer's closed contours and the associated frequency statistics are the most stable and discriminatory [21]. The precision of the feature extraction procedure is regulated by the threshold value, which is determined using these numbers. Both real-time and offline signature verification systems can benefit from the proposed method. The average success rate for an English signature was found to be 92.57%, whereas the average success rate for a Chinese signature was found to be 93.68% [22].

With the aid of VGG16, VGG19, ResNet50, and InceptionV3, four of the most widely used general-purpose models for computer vision applications, as well as SigNet and SigNet-F, two pre-trained models provided especially for signature processing tasks, achieved this [23]. They used the UTSig and FUM-PHSD Persian

datasets as benchmarks and the GPDS Synthetic signature and MCYT-75 Latin signature datasets for their experiments. The experimental findings that were obtained confirm the efficacy of the SigNet and VGG16 models for verifying signatures and the superiority of VGG16 in the signature recognition task, as demonstrated by comparison with earlier studies. The implementation of deep learning artificial neural networks has led to advancements in the field of pattern identification, particularly in the recognition of handwritten text. For every step of the process—online or offline method, signature verification, writing or writer identification, segmentation, or feature extraction—a broad spectrum of artificial neural network (also called ANN) approaches is used [25]. This area of study now needs to evaluate the viability of recent findings in order to plan an organized course for the future (Table 1).

### 3 Preliminary

Traditionally, forged signature detection was manual work, but recently, with the advancement of computer science and computational power, many methods have been designed to check the signature from the image itself without analyzing the signature manually.

#### 3.1 Traditional Method

Traditional methods use image operations and extract useful information from the image, such as character size, orientation, ink width, etc. These approaches are available with different techniques and they are implemented using the OpenCV library. In conventional methods, for detecting signature forgery using OpenCV has several disadvantages. It can be vulnerable to noise and fluctuations in signatures. These approaches are computationally expensive and it mainly depends on known samples of the signer's handwriting. Apart from this, these methods are not adaptable enough to recognize novel types of forgeries. Moreover, it might have trouble identifying intricate ones. So, these constraints need to be taken into consideration while choosing a signature forgery detection technology and the method's applicability must be evaluated.

#### 3.2 Deep Learning Method

When artificial neural networks combined with deep learning methods have proven to be successful in identifying and detecting fake signatures. For differentiating between real and forged signatures, these types of systems are helpful by finding patterns in the signatures. In this method, the model is trained on a large dataset of authentic

**Table 1** The following table contains different techniques opted by researchers for handwritten signature verification and forgery detection

Author, Year	Dataset	Approach	Accuracy and outcomes
Alsuhimat and Mohamad [2]	In experimental procedure CEDAR and UTSig datasets were used	LSTM, SVM, and KNN. CNN and HOG methods were also applied	92% and 1.67 seconds, respectively, for LSTM accuracy on the USTig dataset and 76% and 20.3 s on the CEDAR dataset
Wei et al. [25]	Used the SVC2004 database	Spline interpolation and two different kinds of ANNs	Achieved 87.7% accuracy on the SVC2004 database
Vinod and Das [24]	Assessed more than 20,000 images in 19 distinct classifications	CNN, Oriented FAST and Robust BRIEF (ORB), Scale Invariant Feature Transform (SIFT) and Mean Square Error (MSE)	Here, convolution layer produces a 4D array and it includes depth, batch size, width, and height as its output. CNN's accuracy as a result is 95%
Reyes et al. [20]	300 images in all were used, 150 of which were used for authentic signatures and 150 for fake ones	To identify signature forgeries, You-Only-Look-Once (YOLO) architecture is implemented	When a visible spectrum camera picture is used to train YOLO v3, the performance of the YOLO network for signature forgery detection is still better
Okawa [16]	MCYT-100 and SVC2004 Task1/Task2 are standard internet signature datasets	Makes use of a mean template set and weighted multiple dynamic temporal warping (DTW) distances	Suggested an innovative technique for online signature verification by using a single template that combines a weighting scheme with a mean template set
Poddar et al. [17]	Training dataset of 1320 images	Utilizing the Crest-Trough approach, CNN, and the Harris and Surf algorithms for forgery detection	Identification with neural networks yields an accuracy or efficiency of 94 percent
Foroozandeh et al. [4]	Multi-scripted signature databases	Hybrid neural network based on CNN-BiLSTM	It combines the long-term data dependence learning property of BiLSTM with the high-level local features learning power of CNN

and forged signatures to identify the unique characteristics or features of authentic signatures. Moreover, it helps to determine possible forgeries. It also helps in identifying the differentiating features of authentic signatures. For this approach, convolutional neural networks are helpful in extracting relevant and useful information from signature images like shape, texture, and stroke direction. Then, this information is examined to confirm the authenticity of the signature.

### 3.3 CNN-Based Detection

The design of convolutional neural networks and structure of human brain's connection are exact replicas. The arrangement of convolutional neural network neurons is similar to the way the brain's frontal lobe handles visual data. It can be used to classify images. An input image is fed into a CNN model, which then analyzes and categorizes it. A neural network consisting of many layers is the fundamental component of a CNN. It simply makes use of several layers of image processing algorithms and various computations on the pixels to comprehend and learn.

The CNN levels are as follows: Pooling Layer, Activation Layer, Fully Connected Layer (Dense Layer), and Dropout Layer. A CNN-based image model receives the input image of a handwritten character in a conventional handwriting recognition system. To determine the critical characteristics that distinguish one character from another, the model applies several convolutional and pooling layers to assess the image. After that, a classification layer is applied to the model's output in order to forecast the character that the input picture depicts. Mechanizing the process of reading handwritten writing has increased accuracy and efficiency. Several industries, including banking, education, and the postal service, have used CNN-based image models for handwriting recognition.

## 4 Experiments

### 4.1 Dataset

The Handwritten Signature Datasets by Ishani Kathuriya Kaggle [11] have been used for these experiments. The dataset used for the experiments contains the signatures of thirty persons, both actual and fraudulent. Every person has five valid signatures that they generated themselves and five fraudulent signatures that were created by some other person. The naming of the images is as follows: An image of person number 023's signature made by person 06 is NFI- 00602023. This signature is fake. An image of person number 021's handwritten signature, NFI-02103021. This signature is authentic. The length of the dataset is low; hence, data augmentation was required.

### 4.2 Dataset Preprocessing

Kera's Image Data generator function was used to divide the dataset into three parts. They are training, validation, and testing sets. The Image Data Generator function allows the execution of image data augmentation and preprocessing, as well as producing batches of pictures for training and evaluation. The flow from the directory method's validation split argument allows for selecting the proportion of

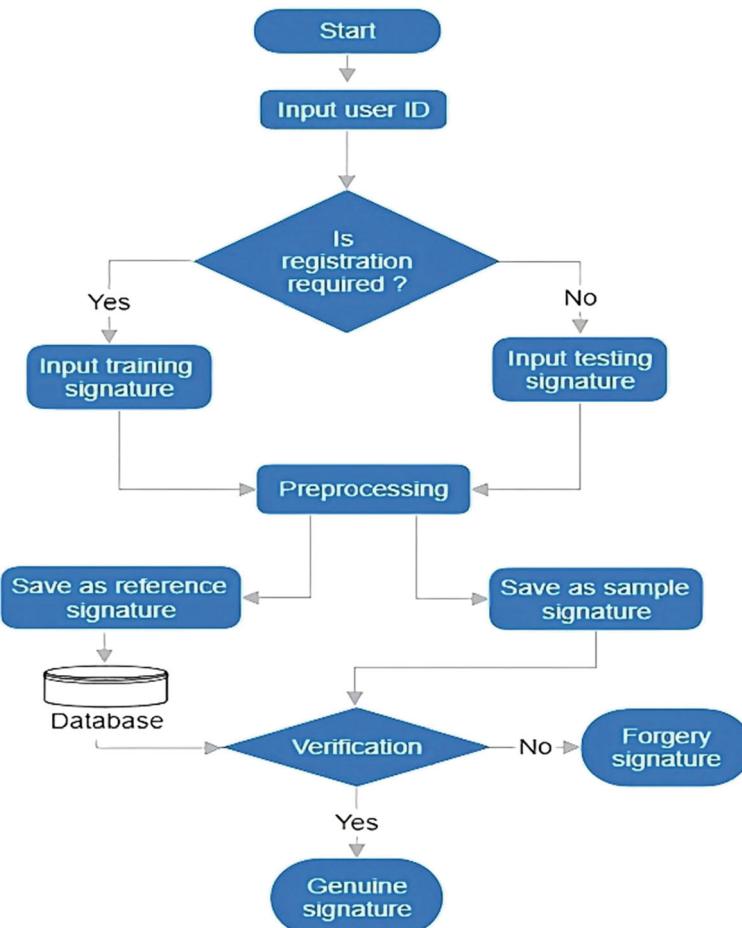
photos to be utilized for validation. The models can be trained and accessed after creating the batches of images for training and validation. It is also important to save a distinct collection of images for testing purpose so that it will be used for evaluating the model's ultimate efficiency. Solving the problem with a model is not enough, providing an interface to the backend model is very much helpful for naïve users to leverage and use the power of deep learning without much understanding the input output of the model. The interface will do that work for them, thus everyone can check a signature image if it is forged or not.

The efficiency of the retraining process is greatly influenced by the pre-trained model selection, the architecture modifications applied to it, and the training dataset choice. Many computer vision applications, including segmentation, object identification, and picture classification, can make use of the retrained model that is produced. During the registration process in this web application, the model is retrained using the new data that the user provides. As a result, during the procedure, the weights are adjusted. The dataset for retraining is kept small with smaller learning rate so that it will not take much time during retraining and the smaller learning rate won't affect the base model much, that means only minor changes in the weights are made during the process of retraining in the registration process (Fig. 1).

The performance of model can be enhanced over time by continuously retraining it with new data. This will help in making it an effective tool for applications that need for ongoing learning and adaptation. The performance of convolutional neural network model on a specific task can be improved by retraining it. It involves training the model again on new data or with modifications to its architecture. The steps that are included during this process are preparing the dataset, selecting a pre-trained convolutional neural network model, modifying the architecture, training the model, validating its performance, and testing it on unseen data. Retraining this type of model can be a time consuming and computationally expensive task, but it is required to increase its precision and effectiveness on particular tasks.

## 5 Results and Discussions

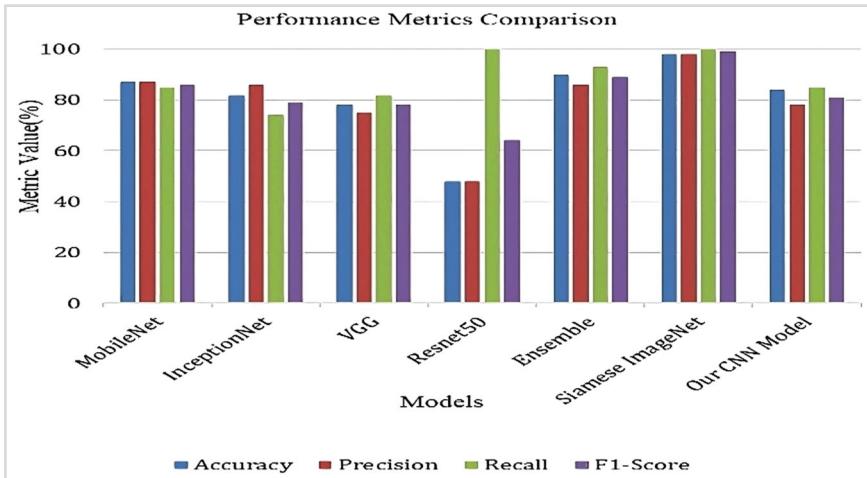
The ensemble model and the Siamese ImageNet model produced the best results, from the following seven models. The accuracy of the basic CNN model is 0.84, that of the MobileNet model is 0.87, that of the Inception model is 0.82, that of the VGG model is 0.78, and the accuracy of the ensemble of the four models that came before it was 90 percent. With an accuracy of 0.98 on the training set, the Siamese image model performed best, however, it was heavily biased to only recognize handwritten images that were authentic. Thus, when considering the precise classification of real and false images, the ensemble model surpasses all other methods (Fig. 2 and Table 2).



**Fig. 1** Flowchart of the handwritten signature verification and forgery detection for web application

## 6 Conclusion and Future Scope

Users of the application can upload photos of signatures to get a report on the signature's legitimacy. For the purpose of performing the signature forgery detection task, the TensorFlow model utilized in the backend was pre-trained on a sizable dataset and fine-tuned on a smaller dataset. A dynamic and responsive user experience is possible because of the frontend's use of React. The application allows users to upload photos of their signatures and receive the findings of the analysis. Processing user input and putting the signature images through the TensorFlow model to check for forgeries are the responsibilities of the Django backend. The findings are shown on the frontend for the user. Overall, it provides the users a practical and simple approach to assess the reliability of signatures. While the React frontend offers a simple and easy-to-use



**Fig. 2** Performance metrics comparison of the models

**Table 2** Performance comparison

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
MobileNet [6]	87	87	85	86
InceptionNet [5]	82	86	74	79
VGG [8]	78	75	82	78
Resnet50 [7]	48	48	100	64
Ensemble [10]	90	86	93	99
Siamese ImageNet [9]	98	98	100	99
Our CNN Model	84	78	85	81

user experience, the usage of Django and TensorFlow enables powerful and precise signature forgery detection.

The model used is Mobile Net V2 with some extracted dense layers for prediction according to the data collected, and it was a trade-off to use the model with big architecture and training time of the model. The various techniques implemented are covered in the section on deep learning methods, and finally, we decided to implement the backend with a comparatively smaller model, which takes less time to train and retrain. The model is trained initially on a large dataset that covers a wide range of signatures, and later, while registration, the model is again retained with a smaller set of data uploaded by the user with a smaller learning rate so that the initial model does not get biased towards the latest data. In this way, we are able to manage large datasets and update the model with the newer dataset. By improving the feature-extraction procedure, future authentication of signatures efficiency and prediction potential can be enhanced.

## References

1. Alajrami E, Ashqar BAM, Abu-Nasser BS, Khalil AJ, Musleh MM, Barhoom AM, Abu-Naser SS (2020) Handwritten signature verification using deep learning. *Int J Acad Multidiscip Res (IJAMR)* 3(12):39–44. <https://philpapers.org/rec/ALAHSV>
2. Alsuhimat FM, Mohamad FS (2023) A hybrid method of feature extraction for signatures verification using CNN and HOG a Multi-classification approach. *IEEE Access* 11:21873–21882. <https://doi.org/10.1109/access.2023.3252022>
3. Blessy P, Kathiresan K, Yuvaraj N (2023) Deep learning approach to offline signature forgery prevention. *IEEE Xplore*. <https://doi.org/10.1109/icaccs57279.2023.10112906>
4. Foroozandeh A, Askari Hemmat A, Rabbani H (2020) Offline handwritten signature verification and recognition based on deep transfer learning. In: 2020 international conference on machine vision and image processing (MVIP). <https://doi.org/10.1109/mvip49855.2020.9187481>
5. <https://keras.io/api/applications/inceptionv3/>
6. <https://keras.io/api/applications/mobilenet/>
7. <https://keras.io/api/applications/resnet/>
8. <https://keras.io/api/applications/vgg/>
9. [https://keras.io/examples/vision/siamese\\_network/](https://keras.io/examples/vision/siamese_network/)
10. <https://sailajakarra.medium.com/ensemble-scikit-learn-and-keras-be93206c54c4>
11. <https://www.kaggle.com/datasets/ishanikathuria/handwritten-signature-datasets>
12. Longjam T, Kisku DR, Gupta P (2023) Writer independent handwritten signature verification on multi-scripted signatures using hybrid CNN-BiLSTM: a novel approach. *Exp Syst Appl* 214:119111. <https://doi.org/10.1016/j.eswa.2022.119111>
13. Lopes JAP, Baptista B, Lavado N, Mendes M (2022) Offline handwritten signature verification using deep neural networks. *Energies* 15(20):7611. <https://doi.org/10.3390/en15207611>
14. Lopes JAP, Baptista B, Lavado N, Mendes M (2022) Offline handwritten signature verification using deep neural networks. *Energies* 15 (20):7611. <https://doi.org/10.3390/en15207611>
15. Muhtar Y, Kang W, Rexit A, Mahpirat, Ubul K (2022) A survey of offline handwritten signature verification based on deep learning. *IEEE Exp.* <https://doi.org/10.1109/prml56267.2022.9882188>
16. Okawa M (2020) Online signature verification using single-template matching with time-series averaging and gradient boosting. *Pattern Recogn* 102:107227. <https://doi.org/10.1016/j.patcog.2020.107227>
17. Poddar J, Parikh V, Bharti SK (2020) Offline signature recognition and forgery detection using deep learning. *Procedia Comput Sci* 170:610–617. <https://doi.org/10.1016/j.procs.2020.03.133>
18. Remaida A, Aniss Moumen Y, Sabri Z (2020) Handwriting recognition with artificial neural networks a decade literature review. *Int Conf Netw*. <https://doi.org/10.1145/3386723.3387884>
19. Ren Y, Wang C, Chen Y, Chuah MC, Yang J (2020) Signature verification using critical segments for securing mobile transactions. *IEEE Trans Mobile Comput* 19(3):724–739. <https://doi.org/10.1109/TMC.2019.2897657>
20. Reyes RC, Polinar MJ, Dasalla RM, Zapanta GS, Melegrito MP, Maaliw RR (2022) Computer vision-based signature forgery detection system using deep learning: a supervised learning approach. In: IEEE international conference on electronics, computing and communication technologies (CONECCT), Bangalore, India, pp 1–6. <https://doi.org/10.1109/CONECCT55679.2022.9865776>
21. Soelistio EA, Hananto Kusumo RE, Martan ZV, Irwansyah E (2021) A review of signature recognition using machine learning. *IEEE Xplore*. <https://doi.org/10.1109/ICCSAI53272.2021.9609732>
22. Souza VLF, Oliveira ALI, Cruz RMO, Sabourin R (2020) A white-box analysis on the writer-independent dichotomy transformation applied to offline handwritten signature verification. *Exp Syst Appl* 154:113397. <https://doi.org/10.1016/j.eswa.2020.113397>

23. Tolosana R, Vera-Rodríguez R, González-García C, Fiérrez-Aguilar J, Morales A, Ortega-García J, Ruiz-García JC, Romero-Tapiador S, Rengifo S, Caruana M, Jiang J, Lai S, Jin L, Zhu Y, Galbally J, Diaz M, Ferrer MA, Gomez-Barrero M, Hodashinsky IA, . . . Jabin S (2022) SVC-onGoing: signature verification competition. Pattern Recogn 127:108609. <https://doi.org/10.1016/j.patcog.2022.108609>
24. Vinod B, Das S (2023) Handwritten signature identification and fraud detection using deep learning and computer vision. IEEE Xplore. <https://doi.org/10.1109/icscd56580.2023.10104929>
25. Wei W, Ke Q, Połap D, Woźniak M (2023) Spline Interpolation and Deep Neural Networks as Feature Extractors for Signature Verification Purposes. IEEE Internet Things J 10(3):2152–2161. <https://doi.org/10.1109/JIOT.2021.3086034>

# Exploring Methodologies for Computing Sentence Similarity in Natural Language Processing



Sagar Mondal, Abirami Gurushanker, Mirudhula Loganath,  
Rishima Chowdhury, Sankari Karthik, Lekshmi Kalinathan<sup>✉</sup>,  
Janaki Meena Murugan, Marimuthu Marimuthu, and Saravanan Palani

**Abstract** The primary objective of this paper is to provide a thorough examination and comparative analysis of various methodologies for computing sentence similarity within the domain of natural language processing (NLP). By exploring a wide range of approaches—string-based, syntax-based, graph-based, transformer-based, and prompt engineering-based methods—the paper aims to evaluate the effectiveness and limitations of each technique. Specific methods discussed include Word2Vec, WordNet, GUSUM, BERT, SGPT, and Angle. Our findings highlight the broader implications of semantic similarity over string-based techniques, with a notable shift towards Language Model Models (LLMs) alongside transformer

---

S. Mondal · A. Gurushanker · M. Loganath · R. Chowdhury · S. Karthik · L. Kalinathan (✉) ·

J. M. Murugan · M. Marimuthu · S. Palani

Vellore Institute of Technology, Chennai, India

e-mail: [lekshmi.k@vit.ac.in](mailto:lekshmi.k@vit.ac.in)

S. Mondal

e-mail: [sagar.mondal2021@vitstudent.ac.in](mailto:sagar.mondal2021@vitstudent.ac.in)

A. Gurushanker

e-mail: [abirami.gurushanker2021@vitstudent.ac.in](mailto:abirami.gurushanker2021@vitstudent.ac.in)

M. Loganath

e-mail: [mirudhula.l2021@vitstudent.ac.in](mailto:mirudhula.l2021@vitstudent.ac.in)

R. Chowdhury

e-mail: [rishima.chowdhury2021@vitstudent.ac.in](mailto:rishima.chowdhury2021@vitstudent.ac.in)

S. Karthik

e-mail: [sankari.karthik2021@vitstudent.ac.in](mailto:sankari.karthik2021@vitstudent.ac.in)

J. M. Murugan

e-mail: [janakimeena.m@vit.ac.in](mailto:janakimeena.m@vit.ac.in)

M. Marimuthu

e-mail: [marimuthu.m@vitstudent.ac.in](mailto:marimuthu.m@vitstudent.ac.in)

S. Palani

e-mail: [saravanan.p@vitstudent.ac.in](mailto:saravanan.p@vitstudent.ac.in)

dominance. Future research suggestions include developing algorithms for unsupervised sentence embeddings, handling datasets exceeding sequence length limits, and exploring context regularization methods. Analyzing challenges in sentence-based similarity via graph-based approaches and experimenting with prompt engineering techniques are also recommended, providing a roadmap for advancing sentence similarity computation in NLP.

**Keywords** Natural Language Processing · Semantic similarity · Transfer learning techniques · Sentence representation · Word embeddings · Cosine similarity

## 1 Introduction

The computation of sentence similarity is a crucial aspect in natural language processing, with applications in semantic search, summarization, question answering, document classification, and plagiarism detection. The accuracy of measuring sentence similarity has garnered significant attention, leading to the proposal of various methods to quantify the closeness between sentences. These methods can be broadly categorized into string-based similarity and semantic similarity, with the latter focusing on the meaning of sentences rather than their literal composition.

String-based similarity methods compare sentence composition based on literal character sequences using techniques like Levenshtein distance, Jaccard similarity, and cosine similarity, offering computational efficiency and ease of implementation in scenarios prioritizing straightforwardness over semantic nuances. Semantic similarity methods, in contrast, involve corpus-based statistical analysis, knowledge-based semantic networks like WordNet, and structure-based inference, leveraging deep learning to capture semantic features of words and sentences. These methods find applications in diverse domains such as requirements engineering, internet search, biomedical fields, information retrieval, text summarization, and sentiment analysis, each requiring tailored algorithms while adhering to the fundamental concept of semantic similarity computation.

To the best of the authors' knowledge, the methods discussed for sentence similarity have been scarcely explored [1–3]. In this paper, we will be exploring various methodologies for computing sentence similarity. Each method is systematically reviewed, highlighting the core features, advantages, and drawbacks. Finally, we also provide suggestions for improvement and future work.

## 2 Background Details and Related Work

In the word-based methods for sentence similarity, [4] utilizes Iterative thresholded subgraph reconstruction and semantic relation path (SRP) for Word Sense Disambiguation. Ruby and Daya [5] introduce a model using semantic feature-based weighted word embeddings for text summarization, capturing semantic relationships, and incorporating TF-TDF for efficient distribution and representation learning. The irrelevant sentences are then eliminated using k-means clustering. In [6], the authors introduce two approaches: a feature-based approach that uses ELMo/BERT embeddings, and a fine-tuning approach that employs BERT and RoBERTa models. The sentence embedding vector [7] is obtained by analyzing alignment and novelty properties of word representations across multiple layers and calculating the weighted average using word importance measures.

Syntax-based approaches in sentence similarity evaluation incorporate syntactic structures and word order information. Wei et al. [8] propose a comprehensive solution considering factors like word order, syntactic structure, and semantics, aiming for thorough and precise evaluation. Additionally, Oya [9] introduces a tree kernel-based computation method (CPT-TK) for constituency parse trees to enhance syntactic similarity assessment in short texts. Gupta et al. [10] suggest a WordNet-based approach, leveraging lexical databases and corpus statistics to disambiguate words, create semantic vectors, and consider word order for efficient and accurate similarity computation.

Transformer-based methods, like Yu et al.'s [11] study, use ensemble techniques and text preprocessing to improve semantic similarity accuracy in patent documents, facing challenges in document complexity and scalability. Another approach [12] employs automated thresholds on word vectors for semantic sentence representation, exhibiting competitive performance across multilingual datasets. Additionally, transfer learning and contextual adjustments [13, 14] refine BERT models for better sentence relation prediction in NLP tasks. Furthermore, enhancements to BERT's capabilities [1, 15] in generating meaningful sentence embeddings and measuring similarity through contextual probabilities are explored, utilizing surrogate models for efficient cost estimates despite computational complexities.

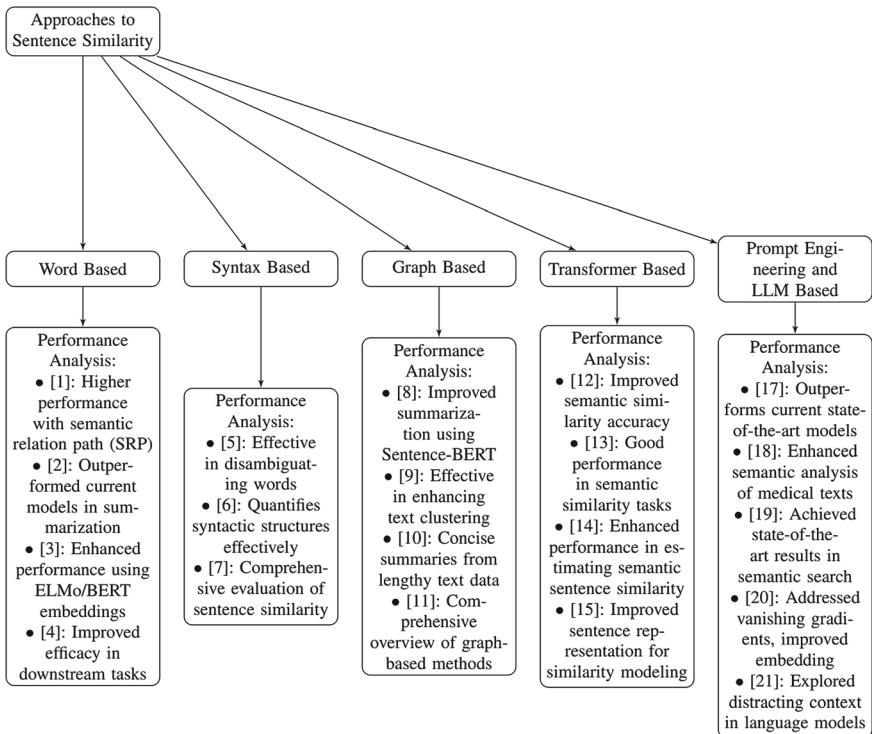
Text is modeled as a graph, with nodes as sentence embeddings and edges indicating semantic similarities [16, 17]. Sentence embeddings are computed using models like Sentence-BERT and roberta-base-nli-stsb-mean-tokens [16]. Sentence importance is determined iteratively based on surrounding sentences, and a modified page ranking algorithm is employed for graph ranking [18].

The final approach we explore is using LLMs and prompt engineering. PromCSE improves Universal Sentence Embeddings by using soft prompts and a different loss function [1]. Leveraging GPT-4, [19] proposes a novel methodology for semantic analysis of medical texts, aiming to bridge the gap between AI-generated language and ground truth data in medical contexts. SGPT utilizes decoder-only transformers for sentence embeddings and semantic search, achieving state-of-the-art results [2]. AnglE introduces an angle-optimized text embedding model to address vanishing

gradients, optimizing text embeddings by focusing on angles, which is combined with an ensemble of LLMs [3]. LLMs like Codex and GPT-3.5 can be distracted by irrelevant context. Shi et al. [20] explore prompting techniques to enhance problem-solving accuracy, introducing the Grade-School Math with Irrelevant Context (GSM-IC) dataset and proposing strategies to mitigate distractions.

### 3 Approaches to Sentence Similarity

Fig. 1 presents a comprehensive overview of various approaches to sentence similarity, categorized into different methodologies: Word Based, Syntax Based, Graph Based, Transformer Based, Prompt Engineering, and Language Model (LLM) Based. Each approach is accompanied by performance analysis findings from referenced studies.



**Fig. 1** Various Approaches to Sentence Similarity

### ***3.1 Word Based***

In [4], a unique iterative thresholded subgraph reconstruction approach is introduced, using word vector representations and knowledge graphs. The method efficiently resolves ambiguity in words by grouping them according to semantic notions. By employing the BFS algorithm for semantic relation path search and SRP2Vec for word embeddings, the technique shows higher performance. However, it may have issues such as the spread of errors and possible enhancements in handling adjectives and adverbs.

Ruby and Daya [5] integrate TF-IDF and Word2Vec skip-gram for word embeddings in text summarization, employing the Summary Generation Algorithm for sentence selection via k-means clustering and weighted techniques, outperforming existing models. Md et al. [6] introduce two methods utilizing contextualized embeddings from ELMo/BERT models for sentence comparison, achieving enhanced performance through fine-tuning with BERT and RoBERTa models, particularly with improved pre-training methods, validated on six datasets. [7] determines sentence embeddings considering alignment and novelty features of word representations, effectively capturing semantic information across layers and word significance, demonstrating adaptability and resilience in various tasks. In summary, the research underscores progress in disambiguating words, streamlining text summarization, and improving sentence comparisons. However, it acknowledges challenges such as error propagation and managing adjectives and adverbs.

### ***3.2 Syntax Based***

The study delves into syntax-based methods for evaluating sentence similarity, integrating syntactic structures and word order information [8] through techniques such as syntactic parse trees, dependency trees, and graph centrality. Utilizing a multi-feature fusion approach, it combines semantic, syntactic, and word order similarity to thoroughly assess sentence similarity, considering factors like word order, syntactic structure, and semantics. Additionally, the WordNet-based method involves disambiguating words, creating semantic vectors, and considering word order for similarity computation [10], leveraging WordNet noun IS-A and verb relationships between sentences, and suggesting integration with domain-specific data, treating course objectives as sentences in NLP, traversing the lexical database, computing similarity, and comparing results with existing algorithms over standard benchmarks. In [9], a syntactic dependency tree-based approach is introduced for measuring sentence similarity in a multi-lingual parallel corpus, utilizing the Euclidean distance of dependency trees to quantify syntactic structures and translation pairs, offering insights into language structural differences and similarities through corpus analysis. Furthermore, the Short-Text Similarity Model in [21] integrates semantic and syntactic information for a comprehensive short-text similarity evaluation, presenting

the KEBERT-GCN semantic similarity model that integrates external knowledge into BERT to effectively use fine-grained word relationship information. This model employs a tree kernel-based computation method for constituency parse trees (CPT-TK) to obtain syntactic structure information and judge syntactic similarity, aiming to enhance word similarity matrix construction, resolve word ambiguity in knowledge bases, and explore alternative BERT variants for enhancing the KEBERT-GCN model. Covering the syntax-based approach to assess sentence equivalence, this summary highlights the consideration of syntactic structures and word orders through tools like syntactic parse trees, dependency trees, and graph centrality. It explores WordNet-based and syntactic dependency tree-based approaches, alongside the Short-Text Similarity Model. This model utilizes the KEBERT-GCN model to merge semantic and syntactic information, aiming to enhance classification accuracy.

### 3.3 *Graph Based*

The GUSUM method [16] enhances unsupervised extractive document summarization by filtering similar sentences and employing Sentence-BERT and roberta-base-nli-stsb-mean-tokens models for sentence representation, constructing an undirected graph based on semantic similarities between sentence embeddings to identify crucial sentences, albeit facing challenges in summarizing lengthy documents. In [17], a novel approach for text similarity measurement is introduced, leveraging named entities to improve text clustering, extracting top-ranked entities, mapping them into an undirected n-gram graph, and assessing graph similarity, although it may struggle with many categories and sparse feature space, and may not fully capture semantic meanings. Moreover, [18] adopts a supervised learning approach for generating concise summaries from lengthy text data, combining multiple similarity measures via a regression model for enhanced estimation accuracy, yet the supervised similarity learning algorithm's performance improvement is limited, likely due to data sparsity and the constrained scope of similarity measures. Additionally, [22] presents a comprehensive overview of graph-based methods in NLP, focusing on classification, similarity measurement, and representation using nouns, verbs, and phrases, underscoring word sense disambiguation's significance, delineating algorithm limitations, and advocating for unsupervised learning in Word Sense Disambiguation (WSD). The discussed methods offer diverse approaches to document summarization and text similarity, each with specific strengths and limitations. Further research is needed to address challenges such as long document summarization and capturing comprehensive semantic meanings.

### 3.4 Transformer Based

The study introduces innovations to enhance semantic similarity in patent analysis [11], proposing an ensemble method merging BERT observations and a novel pre-processing approach for license forms, although scalability issues persist, requiring further refinement. Another study presents a semantic sentence representation approach using feature vector thresholds and pre-trained models, displaying strong performance in semantic similarity tasks [12], yet facing challenges with short sentences and requiring exploration across languages and regions. Moreover, the paper optimizes the BERT model for semantic sentence similarity estimation in NLP [13], stressing contextual representation and transfer learning's significance, while also reviewing related work on BERT, DistilBERT, and ALBERT models. Contextual interpolation techniques like ELMo and BERT further enhance sentence representation for similarity modeling [14], though areas like information retrieval and sensitivity analysis remain unexplored. Sentence-BERT, adapting BERT for semantic text similarity tasks using Siamese triplet networks [15], proves effective but requires careful hyperparameter tuning and lacks exploration in tasks beyond NLI and paraphrase recognition. Introducing enhancements to semantic similarity analysis, this section explores the integration of BERT insights and a novel pre-processing method. Despite these advancements, scalability issues persist. Furthermore, it showcases the utilization of semantic sentence feature vectors and pre-trained models as representation approaches, which, while effective, are limited by short sentences and language diversity.

### 3.5 Prompt Engineering and LLM Based

The PromCSE method in [1] enhances universal sentence embeddings, improving performance with small-scale Soft Prompts and an energy-based Hinge loss, outperforming current models in semantic textual similarity tasks. However, it doesn't boost SimCSE's performance on supervised transfer tasks and lacks discussion on automatically sampling hard negatives. Xu et al. [19] introduce a methodology using GPT-4 for semantic analysis of medical texts, aiming to bridge the gap between AI-generated language and medical ground truth data, though accuracy and clinical validity could be impacted without direct human expert involvement, relying heavily on GPT-4 capabilities. Miklas [2] SGPT employs a decoder-only transformer for sentence embeddings, enhancing search accuracy leveraging pre-trained models despite bias issues and computational resources. The Angle model in [3] addresses vanishing gradients with an angle-optimized text embedding approach but faces limitations with small-scale training sets and challenges in utilizing hard negatives. Meanwhile, [20] explores distractions in LLMs like Codex and GPT-3.5, proposing diverse prompting techniques and introducing the GSM-IC dataset, with suggested

strategies to improve performance amidst irrelevant information, despite acknowledged sensitivity and potential negative impacts of complex prompts on robustness. The referenced studies provide valuable insights into the progress and methodologies concerning Large Language Models (LLMs). These methodologies primarily concentrate on refining sentence embeddings within these models for specific use cases. However, a significant drawback of such techniques is their heavy reliance on computational resources and the potential introduction of bias.

## 4 Results and Discussion

The study presents a comprehensive exploration of methodologies within natural language processing (NLP), spanning various paradigms including word-based, syntax-based, graph-based, transformer-based, and prompt engineering, alongside large language model (LLM)-based approaches. Firstly, word-based techniques are investigated, emphasizing advancements in word sense disambiguation and semantic clustering, with novel contributions enhancing word similarity measurements and disambiguation accuracy. Secondly, syntax-based methods are examined, aiming to integrate syntactic structures and word order information for improved sentence similarity evaluation, addressing limitations of semantic-focused approaches. Thirdly, graph-based approaches are explored, leveraging graph structures and centrality measures for quantifying sentence similarity and document summarization, showcasing innovation in unsupervised extractive summarization and text clustering tasks. Additionally, attention is devoted to transformer-based models, including variants of BERT, revealing their potential for enhancing semantic understanding and reasoning abilities. Finally, the study suggests the introduction of novel techniques in prompt engineering and LLM-based methodologies, focusing on enhancing sentence embeddings, semantic analysis in medical texts, and addressing the challenge of distractibility. Table 1 provides a comprehensive overview of different approaches and their corresponding results in semantic similarity assessment and document summarization tasks, highlighting their novelty and efficacy. Despite these advancements, research gaps persist, particularly in areas such as improving syntactic representation learning, enhancing the robustness of graph-based summarization techniques, and effectively handling domain shifts and bias in LLMs. Moreover, there is a pressing need for more comprehensive evaluation frameworks and standardized benchmarks to facilitate comparative analysis and benchmarking across different NLP approaches, fostering further innovation and advancement in the field.

**Table 1** Comparison of semantic similarity methods and their findings

Approach	Results	Findings
Word-net	Pearson correlation coefficient 0.875 [23]	Computation of Similarity Between Two Pair of Sentence Using Word-Net-2023
Microsoft/DeBERTa-v3-large	Cross-validation score 0.8512 [24]	Semantic Similarity Matching for Patent Documents Using Ensemble BERT-related Model and Novel Text Processing Method-2024
Graph based similarity function	On DUC2006 the results obtained by Rogue1-0.4083 Rogue2-0.0911 Rogue3-0.1485 [25]	Learning similarity function: On DUC2006, obtained better performance results
BERT-NLI-large-last2avg + flow-target	Spearman's rank correlation: $81.10 \pm 0.55$ [26]	On the sentence embeddings from pre-trained language model obtained good Spearman's rank correlation
Mirror-BERT	Spearman's P correlation from 0.526 to 0.755 [27]	Fast, effective, and self-supervised: transforming masked language models into universal lexical and sentence encoders, 2021

## 5 Conclusion

In conclusion, our research provides a comprehensive analysis of various methodologies for computing sentence similarity in natural language processing, including word-based techniques, prompt engineering, and Language Model Models (LLMs). Our findings underscore the broader applications and implications of semantic similarity over string-based methods. The growth of such semantic techniques leads to the rise of transformers. While transformers currently dominate the field, there is a noticeable increase in the adoption of LLMs. This highlights the evolving landscape and the growing recognition of semantic approaches in advancing sentence similarity computation. For future research, we suggest developing algorithms to generate hard negatives from unlabeled data to enhance unsupervised sentence embeddings and mitigate biases from pre-trained language models. Additionally, exploring ways to handle datasets exceeding sequence length limits, optimizing computational efficiency for LLMs, and using context regularization methods to create large-scale datasets with semantic similarity scores are crucial. Analyzing challenges in sentence-based similarity using a graph-based approach and exploring models like S-BERT and roberta-base-nli-stsb-mean-tokens are also recommended. Finally, we propose experimenting with prompt engineering techniques to further enhance text embeddings, providing a roadmap for future research in sentence similarity computation and natural language processing.

## References

1. Jiang Y, Zhang L, Wang W (2022) Improved universal sentence embeddings with prompt-based contrastive learning and energy-based learning. *Find Assoc Comput Ling: EMNLP*
2. Miklas N (2022) Sgpt: Gpt sentence embeddings for semantic search
3. Li X, Li J (2023) Angle-optimized text embeddings
4. Sunjae K, Dongduk O, Youngjoong K (2021) Word sense disambiguation based on context selection using knowledge-based word similarity. *Inf Process Manage* 58(4):102551
5. Ruby R, Daya K (2021) A weighted word embedding based approach for extractive text summarization. *Expert Syst Appl* 186:115867
6. Md TRL, Jimmy XH, Enamul H (2020) Contextualized embeddings based transformer encoder for sentence similarity modeling in answer selection task. In: Proceedings of the twelfth language resources and evaluation conference, Marseille, France, pp 5505–5514
7. Wang B, Kuo CCJ (2020) SBERT-WK: A Sentence Embedding Method by Dissecting BERT-based Word Models. *IEEE/ACM Trans Audio Speech Lang Process* 28:2146–2157
8. Wei C, Wang B, Kuo CCJ (2023) SynWMD: syntax-aware word mover's distance for sentence similarity evaluation. *Pattern Recogn Lett* 170:48–55
9. Oya M (2020) Syntactic similarity of the sentences in a multi-lingual parallel corpus based on the Euclidean distance of their dependency trees. In: Pacific Asia conference on language, information and computation
10. Gupta A, Sharma K, Goyal KK (2023) Computation of similarity between two pair of sentence using word-net. *Int J Intell Syst Appl Eng* 11(5)
11. Yu L, Liu B, Lin Q, Zhao X, Che C (2024) Semantic similarity matching for patent documents using ensemble BERT-related model and novel text processing method. [arXiv:2401.06782](https://arxiv.org/abs/2401.06782)
12. Shajalal M, Atabuzzaman Md, Baby Md, Karim R, Boden Md (2023) Textual entailment recognition with semantic features from empirical text representation. *Speech and language technologies for low-resource languages*, vol 1802, pp 183
13. Yerramreddy DR, Marasani J, Venkata Gowtham PS, Abhishek S, Anjali (2023) An empirical analysis of topic categorization using PaLM, GPT and BERT models. In: Innovations in power and advanced computing technologies (i-PACT), pp 1–6
14. Tahmid Rahman Laskar M, Huang JX, Hoque E (2020) Contextualized embeddings based transformer encoder for sentence similarity modeling in answer selection task. In: Proceedings of the twelfth language resources and evaluation conference. European Language Resources Association, Marseille, France, pp 5505–5514
15. Reimers N, Gurevych I (2019) Sentence-BERT: sentence Embeddings using Siamese BERT-Networks. In: Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP), Hong Kong, China, pp 3982–3992
16. Gokhan GT, Smith P, Lee M (2022) GUSUM: graph-based unsupervised summarization using sentence features scoring and Sentence-BERT. In: Proceedings of TextGraphs-16: graph-based methods for natural language processing, pp 44–53
17. Tsekouras L, Varlamis I, Giannakopoulos G (2017) A graph-based text similarity measure that employs named entity information. In: Proceedings of the international conference recent advances in natural language processing, pp 765–771
18. Ouyang Y, Li W, Wei F, Lu Q (2009) Learning similarity functions in graph-based document summarization. In: Computer processing of oriental languages. language technology for the knowledge-based economy. Springer, pp 189–200
19. Xu S, Wu Z, Zhao H, Shu P, Liu Z, Liao W, Li S, Sikora A, Liu T, Li X (2024) Reasoning before comparison: LLM-Enhanced semantic similarity metrics for domain specialized text analysis
20. Shi F, Chen X, Misra K, Scales N, Dohan D, Chi EH, Schae'ri N, Zhou D (2023) Large language models can be easily distracted by irrelevant context. In: International conference on machine learning, pp 31210–31227

21. Zhou Y, Li C, Huang G, Guo Q, Li H, Wei X (2023) A short-text similarity model combining semantic and syntactic information. *Electronics* 12(14). <https://doi.org/10.3390/electronics12143126>
22. Patel R, Kyada B (2016) Graph based methods for classification, similarity and representation using noun, verbs –a survey. *Int J Eng Res Technol (IJERT)* 05(04)
23. Gupta A, Sharma K, Goyal KK (2023) Computation of similarity between two pair of sentence using word-net. *Int J Intell Syst Appl Eng* 11(5s):458–467
24. Liqiang Y, Liu B, Lin Q, Zhao X, Che C (2024) Semantic similarity matching for patent documents using ensemble BERT-related model and novel text processing method. [arXiv: 2401.06782](https://arxiv.org/abs/2401.06782)
25. Ouyang Y, Li W, Wei F, Lu Q, Learning similarity functions in graph-based document summarization. In: Computer processing of oriental languages. Language technology for the knowledge-based economy: 22nd international conference, ICCPOL 2009, Hong Kong, Proceedings 22. Springer, Berlin, Heidelberg, pp 189–200
26. Li B, Zhou H, He J, Wang M, Yang Y, Li L (2020) On the sentence embeddings from pre-trained language models. [arXiv:2011.05864](https://arxiv.org/abs/2011.05864)
27. Liu F, Vulić I, Korhonen A, Collier N (2021.) Fast, effective, and self-supervised: transforming masked language models into universal lexical and sentence encoders. [arXiv:2104.08027](https://arxiv.org/abs/2104.08027)

# Enhancing Real-Time Gesture Recognition Systems for Virtual Reality Applications Using Deep Learning Techniques



Rahul Kumar, Lekshmi Kalinathan, and Janaki Meena Murugan

**Abstract** Virtual Reality (VR) technology has seen significant advancements, with gesture control emerging as a promising avenue for enhancing user interaction and accessibility, particularly for individuals with physical disabilities. This research paper presents a comprehensive study on real-time gesture recognition systems aimed at improving the overall user experience in VR environments. By addressing the limitations of existing techniques such as K-Nearest Neighbors (KNN) and Lucas-Kanade Pyramidal Optical Flow (LKPOF), this paper proposes the integration of Support Vector Machines (SVM), Convolutional Neural Networks (CNN), and YOLOv3 to achieve more accurate and efficient hand gesture detection and analysis. The incorporation of machine learning methods significantly enhances gesture recognition capabilities, paving the way for practical implementation in VR systems. Additionally, this paper reviews existing studies focused on improving hand gesture recognition while maintaining speed and response time. Looking ahead, future research will focus on deploying these models into VR platforms to minimize latency and ensure robust real-time performance, facilitating real-world validation and refinement of the proposed solutions in immersive settings.

**Keywords** Gesture Control Technology · Human–Machine Interaction · Virtual Reality Technology · Hand Gestures and Machine Learning

---

R. Kumar · L. Kalinathan () · J. M. Murugan  
Vellore Institute of Technology, Chennai, India  
e-mail: [lekhsmi.k@vit.ac.in](mailto:lekhsmi.k@vit.ac.in)

R. Kumar  
e-mail: [rahulkumar.2023@vitstudent.ac.in](mailto:rahulkumar.2023@vitstudent.ac.in)

J. M. Murugan  
e-mail: [janakimeena.m@vit.ac.in](mailto:janakimeena.m@vit.ac.in)

## 1 Introduction

In the realm of virtual reality, the advent of gesture-controlled audio and video playback represents a significant leap in human–computer interaction, where the nuanced recognition of human hand gestures paves the way for a more intuitive and natural user experience. This technology transcends traditional haptic and touchscreen interfaces, particularly in applications where such modalities may be limiting, offering a touchless, convenient alternative that is gaining traction across various domains, including automotive safety, smart homes, wearable devices, and beyond. Notably, hand gesture recognition holds immense promise for revolutionizing the way we control multimedia content, fostering more immersive and interactive user experiences by allowing seamless control of smart TVs, media players, and personal devices through simple hand motions. However, realizing this vision necessitates tackling significant challenges in real-time gesture recognition for virtual reality applications, particularly concerning accessibility and the nuanced differentiation of gestures. This research endeavors to address these obstacles by accumulating a diverse array of hand movement images, either through simulation or collaborative efforts, and employing advanced machine learning strategies to refine gesture recognition. By educating the system with a plethora of examples, particularly for complex gestures, and enhancing its ability to discern subtle differences in similar hand movements, the aim is to cultivate an algorithm capable of sophisticated interpretation of hand gestures, thereby elevating the potential for practical implementation in VR contexts and enriching the overall user experience.

## 2 Background Details and Related Work

Gesture detection and recognition have emerged as crucial elements in advancing the realm of human-computer interaction, with numerous researchers contributing innovative solutions to this burgeoning field. The studies introduced a pioneering media player control system that harnesses the power of computer vision and deep learning techniques [1–3]. The approach involved the creation of a bespoke dataset encompassing seven distinct hand gesture classes, aimed at alleviating the inconveniences associated with manual controls. Through their endeavors, they achieved an impressive level of accuracy, thereby marking a significant stride forward in the domain of remote device manipulation.

Pradnya Kedari and Kadam [4] addressed the complexities of hand gesture feature extraction by proposing a deep learning model for computer control without requiring supplementary hardware. Their solution exhibited impressive real-time gesture recognition accuracy [5], propelling progress in this field. Tumuluru et al. [6] explored ResNet and attention mechanisms in interactive projection technology, aiming for high accuracy in diverse gesture recognition [7], advancing interactive systems capabilities.

In sentiment analysis, Waghale et al. [8] highlighted Naive Bayes and Support Vector Machines' superiority in classifying social media sentiments. Meanwhile, a transformer-based model in hand gesture recognition [9] showcased enhanced user experience over traditional methods through comprehensive Quality of Experience (QoE) assessments. Dayanandan et al. [10] developed a Gesture Controlled Media Player utilizing the Tiny Yolov3 model, contributing to real-time gesture interpretation for media control and advancing interaction paradigms.

Li and Chen [11] introduced “GesPlayer,” enhancing video playback with intuitive gestures, expanding interactive media consumption. Another study [12] introduces an AR pop-up book with natural gesture interaction, blending physical and digital worlds seamlessly. Mitre Ortiz et al. [13] introduced an evaluation model combining user-centric methods and motion recognition for enhanced VR user experience, while Chappell et al. [14] proposed VR pre-prosthetic hand training with physics simulation, facilitating effective rehabilitation through real-time feedback. Patil et al. [15] devised a VLC Media Player control system using transfer learning for accurate gesture recognition, enhancing media interaction. Kumar et al. [16] introduced a game control system based on hand gestures, enhancing gaming experience and engagement. Chandu [17] developed an audio player control system utilizing hand gestures for intuitive management, improving user experience, and advancing engineering technology. A system for video player control via hand gestures is introduced, enhancing user experience and advancing hybrid intelligent systems [18]. This seamless control system provides an intuitive way to interact with video content, contributing to hybrid intelligent systems’ progress. Development of a media player controller system using hand gestures facilitates simple media playback control, improving user interaction and advancing mobile computing [19]. This innovative system offers convenience, contributing to sustainable informatics and enhancing media device interaction.

The surveyed literature collectively underscores several pivotal aspects in the evolution of gesture recognition and its integration into interactive technology. At the forefront, research by [1, 20] and [2, 21–23] delves into enhancing media player control through hand gesture recognition, applying computer vision and deep learning techniques to facilitate more natural user interaction with multimedia content. This points to a broader trend where deep learning, especially through models like CNNs and ResNet [6], demonstrates its robustness in hand gesture recognition, proving to be a powerful tool across varied applications. Notably, recent advancements focus on enhancing media player control through hand gesture recognition, utilizing computer vision and deep learning techniques to facilitate natural user interaction with multimedia content. This trend emphasizes the robustness of deep learning models like CNNs and ResNet in hand gesture recognition, showcasing their efficacy across various applications.

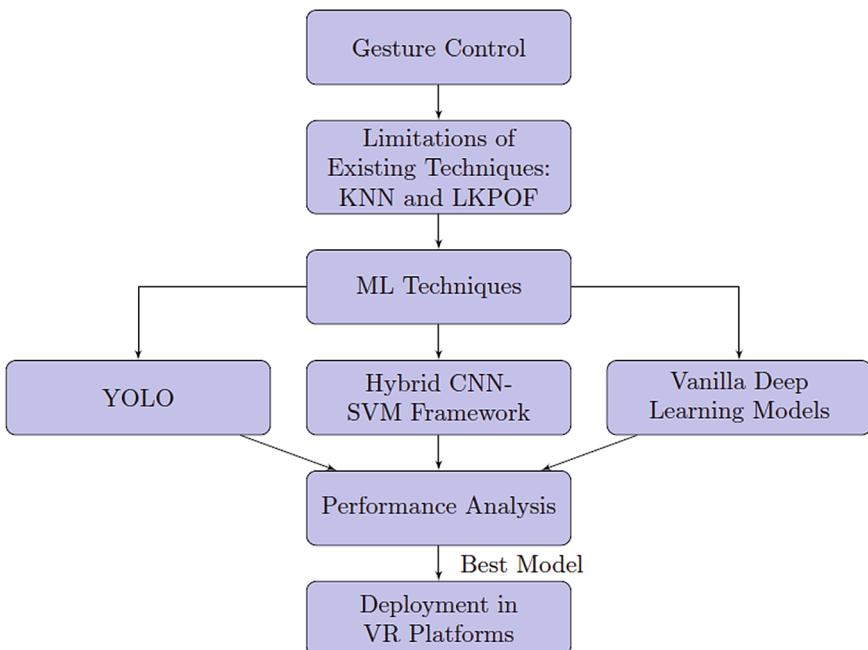
Real-time interaction capabilities stand out as an essential feature across the board, enabling users to interact instantaneously and effortlessly with digital interfaces, whether that’s through controlling media players, engaging with video platforms, or navigating virtual environments [24]. Such real-time responsiveness is central to the goal of seamless human–computer interface. Despite the progress, there are

challenges that remain unaddressed in some studies. While [4, 25, 26], for instance, tackles background noise and the risk of overfitting models to datasets, there's a gap in the collective discussion around nuanced issues like the recognition of complex gestures, the ambiguity between similar gestures, and the rapid identification of incorrect gestures, which are critical for advancing the technology's reliability and user experience.

### 3 Proposed Approach

#### 3.1 Hybrid CNN-SVM Framework for Enhanced Gesture Recognition

The combination of Convolutional Neural Networks (CNNs) and Support Vector Machines (SVMs) can be a powerful and effective option for a hand gesture detection system for several reasons. This hybrid approach leverages the strengths of both CNNs and SVMs to improve accuracy and robustness of the system. Figure 1 shows the framework for gesture control in virtual reality.



**Fig. 1** Overview of gesture control in virtual reality

**Feature Extraction by CNN.** Convolutional Neural Networks (CNNs) have emerged as formidable tools for feature extraction from visual data. Specifically within the domain of hand gesture recognition, CNNs exhibit remarkable prowess in autonomously discerning and extracting pertinent features from images. These features encompass various aspects such as the spatial configuration of fingers, the contours of hand shapes, and the nuanced intricacies inherent in different gestures. Leveraging their inherent architecture, CNNs excel in capturing intricate patterns and hierarchical representations of features, rendering them highly adept at tackling the complexities inherent in hand gesture recognition tasks. With their capability to automatically discern and extract salient features, CNNs stand out as particularly effective tools in facilitating accurate and robust recognition of hand gestures.

**High-Dimensional Feature Space.** The features extracted by Convolutional Neural Networks (CNNs) frequently inhabit a high-dimensional feature space, presenting a formidable challenge in direct manipulation and interpretation. However, Support Vector Machines (SVMs) offer a complementary approach, particularly well-suited for tasks involving classification within such high-dimensional feature spaces. SVMs excel in efficiently segregating data points and delineating decision boundaries, attributes that render them exceptionally suitable for the intricate task of recognizing complex hand gestures. Leveraging their capacity to discern subtle patterns and relationships within the feature space, SVMs serve as powerful allies in the process of accurately identifying and classifying diverse hand gestures, thus enhancing the efficacy of gesture recognition systems.

**Robust Classification.** Support Vector Machines (SVMs) are renowned for their robustness and adeptness in generalizing patterns within data. When coupled with Convolutional Neural Networks (CNNs), which excel at extracting intricate features from visual data, a synergistic relationship emerges. This combination harnesses the strengths of both methodologies: the CNNs' proficiency in feature extraction and the SVMs' robust decision-making capabilities.

By integrating CNNs for feature extraction and SVMs for classification, practitioners can capitalize on the robust decision-making process of SVMs. This symbiotic approach not only enhances the model's accuracy but also mitigates the risk of overfitting, a common concern in machine learning tasks. SVMs, with their ability to delineate clear decision boundaries in high-dimensional feature spaces, contribute to the model's resilience against noisy or ambiguous data, thereby bolstering its generalization capabilities.

Ultimately, the fusion of CNNs and SVMs represents a potent strategy for developing highly effective and robust gesture recognition systems. This amalgamation facilitates nuanced and accurate classification of hand gestures while ensuring the model's ability to generalize well to unseen data, thus elevating the overall performance and reliability of the system.

**Handling Small Datasets.** Support vector machines maintain robustness with limited datasets common in hand gesture recognition, despite challenges in data collection. Integrating convolutional neural networks with SVMs allows leveraging deep learning's benefits for feature extraction and SVMs' classification strength,

proving beneficial in scenarios with scarce labeled data. This hybrid approach optimally utilizes CNNs' intricate feature capturing and SVMs' effective data separation, offering a pragmatic solution to data scarcity in hand gesture recognition, enabling robust performance with modest datasets.

**Interpretable Results.** Support vector machines offer interpretable results, providing insights into significant features crucial for classification and advantageous in hand gesture recognition tasks. This transparency aids in understanding individual feature contributions, enhancing trust in classification outcomes, and empowering users to comprehend classification rationale. SVMs not only ensure accurate gesture classification but also provide meaningful insights into feature-classification relationships, fostering confidence and aiding in feature selection refinement. Their transparency underscores their utility in elucidating complex dynamics in hand gesture recognition, enhancing interpretability and classification result utility.

**Hybrid Modeling.** The fusion of convolutional neural networks and support vector machines creates an ensemble model, mitigating biases and errors by integrating diverse learning algorithms. Leveraging CNNs' feature extraction and SVMs' classification strengths, this approach maximizes efficacy in hand gesture recognition. Fusion offsets biases in individual models, enabling the ensemble to discern intricate patterns and achieve superior performance. This collaborative approach highlights ensemble learning's effectiveness in enhancing adaptability and resilience to dataset variability, surpassing individual models in complex recognition tasks.

Recent advancements in cancer research have demonstrated the pivotal role of accurate identification and detection of fusion genes in non-small cell lung cancer (NSCLC) diagnosis. For instance, Karlson et al. utilized NanoString Technology to develop the “Single Sample Predictor” (SSP), a tool capable of determining NSCLC type and detecting fusion genes like EML4-ALK, KIF5B-RET, CD74-NRG1, and MET exon 14 skipping, thus aiding in diagnosis [21]. Additionally, Meiling Cai et al. proposed the “Multimodal Multi-scale Attention Model (MMAM),” which integrates histopathology and genetic data to predict lung cancer stage, achieving an AUC of 88.51.

AI tools have also emerged as essential aids in lung disease characterization, diagnosis, and risk assessment. Vishwanathan et al. discuss the development of AI tools for lung diseases, utilizing manually designed features and deep learning techniques to enhance diagnosis and predict treatment responses across various lung conditions, including cancer [23]. Moreover, Tankeyych et al. aimed to predict NSCLC patients' response to immunotherapy using PET/CT scans, demonstrating the potential of detailed image data in personalizing treatment [27].

Combining radiomic and genomic data offers deeper insights into NSCLC. Kirienko et al. examined the association between radiomic and genomic features with lung cancer type and recurrence, with machine learning yielding high accuracy in predictions [27]. Steyaert et al. developed a deep learning system to predict brain tumor progression by integrating medical images and genetic information, achieving high prediction accuracy and identifying crucial biological pathways [24].

Moreover, Caruso et al. explored multimodal learning to enhance lung cancer treatment by combining CT images and clinical data, demonstrating improved prediction accuracy by selecting the best models from multiple data sources [25]. These studies collectively underscore the multifaceted methodologies employed in multimodal fusion research, aiming to enhance cancer diagnosis and treatment precision while yielding significant performance outcomes.

### 3.2 *YOLO (You Only Look Once) Model*

You Only Look Once (YOLO) is a state-of-the-art object detection algorithm that has gained significant popularity in recent years [5]. It is a real-time object detection system that can detect multiple objects in an image or video stream with high accuracy and speed. YOLO is based on a deep neural network architecture that divides the input image into a grid of cells and predicts the bounding boxes and class probabilities for each cell. This approach allows YOLO to detect objects in a single pass through the network, making it much faster than other object detection algorithms. YOLO has been used in a wide range of applications, including autonomous vehicles, surveillance systems, and robotics.

YOLOv3-based gesture recognition system has high recognition accuracy for four custom dynamic hand gestures. The effectiveness of the proposed method is verified by the recognition confusion matrix. Additionally, YOLOv3 is reported to be on par with Focal Loss in terms of mean average precision (mAP) measured at 0.5 IOU, but about  $4 \times$  faster. The speed and accuracy of YOLOv3 can be easily adjusted by changing the size of the model, without requiring retraining. Altogether, YOLOv3 is considered to be an extremely fast and accurate object detection algorithm.

## 4 Results and Discussion

The comparative analysis provided in Table 1 reveals varying levels of accuracy and application-specific robustness among different hand gesture recognition methods. CNN-based approaches demonstrate a wide accuracy spectrum, with the highest reported accuracy reaching 96.83% [1], indicating their effectiveness in feature extraction for gesture recognition. However, challenges such as environmental factors and dataset complexity have led to lower accuracies in some studies, dropping as low as 85% [13].

The incorporation of Support Vector Machines (SVMs) with CNNs, as demonstrated in the study referenced as [6], achieved an accuracy of 93.5%, highlighting the efficacy of hybrid models in handling high-dimensional data and enhancing classification robustness. The application of YOLO models, particularly Tiny YOLOv3, resulted in exceptionally high recognition rates, reaching up to 99.9741% for specific gestures [10]. Another YOLO implementation maintained 98% accuracy even under

low-light conditions [15]. These findings underscore the adaptability and efficiency of the YOLO algorithm in real-time gesture recognition scenarios.

Method	Result Analysis	Findings
CNN	[1] achieved 96.83%, [4] achieved 85.90%, [13] achieved 85%, [19] achieved 90% accuracy	Larger, diverse datasets could enhance training, environmental conditions, and cultural variations may affect recognition accuracy
YOLO	[8, 15] showed high recognition accuracy with 99% or specific gestures and 98% in low light	Effective real-time gesture detection; low processing time and robustness in varying conditions, including low light
Machine Learning	[11] demonstrated how semantic gestures enhance user experience with video players	Semantic gesture recognition could improve the intuitiveness of control over video players
KNN	[17] obtained a mean precision of 95.7% in palm identification	Challenges include occlusion, clutter, or variations in hand pose that could affect the precision of the system
Hybrid	CNN-SVM framework[6] reached an overall accuracy of 93.5% in hand gesture recognition	<b>Novelty:</b> The integration of Convolutional Neural Networks (CNN) with Support Vector Machines (SVM) has shown significant improvements in achieving higher accuracy in real-time hand gesture recognition from <b>live video feeds</b> , highlighting their potential for efficient deployment in practical Virtual Reality (VR) applications

Machine learning techniques, as exemplified in [11], though not solely focused on accuracy, have played a crucial role in enhancing user interaction, emphasizing the importance of user-centric design in the development of gesture-based systems. Furthermore, the K-Nearest Neighbors (KNN) algorithm demonstrated a high mean precision rate of 95.7% for palm detection [17], indicating its suitability for straightforward tasks.

The data suggests that combining CNNs with SVMs offers a promising approach for those seeking a balance between real-time functionality and robustness, particularly in scenarios with limited data. YOLO excels in dynamic settings, while KNN maintains its strong case for simplicity and precision. Selecting an appropriate gesture recognition model, therefore, necessitates a careful evaluation of specific operational demands, giving due consideration to both quantitative accuracy and qualitative user experience factors.

In comparison to existing techniques, the hybrid CNN-SVM framework demonstrates superior performance in hand gesture recognition. Additionally, certain hybrid

CNN frameworks have shown promising results, particularly with Surface EMG signals. However, there remains a need for further improvement in gesture control, particularly in the context of live feed videos, a requirement that can be effectively addressed through the hybrid CNN-SVM framework.

## 5 Conclusions

Our research underscores the potential of gesture control technology to revolutionize user interaction within virtual reality, offering notable enhancements in accessibility and user experience, a boon particularly for individuals with disabilities. By addressing the limitations inherent in K-Nearest Neighbors and optical flow techniques, we propose the use of sophisticated methodologies such as support vector machines, convolutional neural networks, and YOLOv3 to substantially increase the precision of hand gesture recognition. This advancement is not merely theoretical but paves the way for the technology's practical application in virtual reality systems, which stands as a significant contribution toward evolving more natural and fluid human-computer interactions. Looking ahead, our future work will concentrate on the hands-on deployment of these models into VR platforms, with an emphasis on minimizing latency and ensuring robust real-time performance. This next step is critical to the real-world validation and refinement of our proposed solutions in immersive settings.

## References

1. Nagalapuram GD, Roopashree S, Varshashree D, Dheeraj D, Nazareth DJ (2021) Controlling media player with hand gestures using convolutional neural network. In: Proceedings of the 2021 IEEE Mysore sub section international conference (MysuruCon), pp 79–86 (2021)
2. Sampath M, Raghavendran V, Sumithra M (2022) Controlling media player using hand gestures with VLC media player. World J Adv Res Rev 14(3):466–472
3. Satyam K, Suma P (2024) Hand controlled media player using hand gestures through machine learning
4. Kedari P, Kadam S, Prasad R (2022) Controlling the computer using hand gestures. Multimedia Res 5(3). <https://doi.org/10.46253/j.mr.v5i3.a2>
5. Sabab SA, Islam SS, Hossain M, Shahreen M (2018) Hand swifter: a real-time computer controlling system using hand gestures. In: Proceedings of the 2018 4th international conference on electrical engineering and information and communication technology (iCEEiCT), pp 9–14
6. Tumuluru P, Lakshmi G, Reddy VSN, Gayatri BN, Varshitha C, Alekhyaa MS (2023) Interactive projection technology using hand gesture recognition with attention mechanism and ResNet. In: Proceedings of the 2023 2nd international conference on applied artificial intelligence and computing (ICAAIC), pp 650–654
7. Guo L, Lu Z, Yao L (2021) Human-machine interaction sensing technology based on hand gesture recognition: a review. IEEE Trans Hum-Mach Syst 51(4):300–309
8. Waghale S, Bankhele S, Badade O, Raikar P (2023) Survey on Vision Based Hand Gesture Interface For Controlling Multimedia Player. Int Res J Mod Eng Technol Sci. <https://doi.org/10.56726/irjmets33682>

9. Floris A, Porcu S, Atzori L (2024) Controlling media player with hands: a transformer approach and a quality of experience assessment. *ACM Trans Multimed Comput Commun Appl* 20(5):1–22
10. Dayanandan A, Chakkungal A, Kommeri A, Koppuliparambil D, Nitnaware P (2020) Gesture controlled media player using TinyYoloV3
11. Li X, Chen Y, Tang X (2022) GesPlayer: using augmented gestures to empower video players. In: Companion proceedings of the 2022 conference on interactive surfaces and spaces, pp 4–8
12. Nor'a MNA, Ismail AW, Aladin MYF (2024) Interactive augmented reality pop-up book with natural gesture interaction for handheld. In: Encyclopedia of computer graphics and games. Springer International Publishing, Cham, pp 984–993
13. Mitre-Ortiz A, Muñoz-Arteaga J, Cardona-Reyes H (2023) Developing a model to evaluate and improve user experience with hand motions in virtual reality environments. *Univ Access Inf Soc* 22(3):825–839
14. Chappell D, Son HW, Clark AB, Yang Z, Bello F, Kormushev P, Rojas N (2022) Virtual reality pre-prosthetic hand training with physics simulation and robotic force interaction. *IEEE Robot Autom Lett* 7(2):4550–4557
15. Patil A, Patil S (2022) Hand gesture recognition system for controlling VLC media player based on two stream transfer learning
16. Kumar GM, Manohar V, Ravi B, Prasad SVS, Paluvatla S, Sateesh R (2022) Game controlling using hand gestures. In: Proceedings of the 2022 international conference on advancements in smart, secure and intelligent computing (ASSIC), pp 1–5
17. Chandu R (2022) Audio player controlling based on hand gesture technique. *Int Res J Mod Eng Technol Sci* 4(6):2604–2611
18. Sangeetha RG, Hemanth C, Nair KS, Nair AR, Shine KN (2022) Hand Gesture control of video player. In: Proceedings of the international conference on hybrid intelligent systems. Springer Nature Switzerland, Cham, pp 726–735
19. Mane V, Baru H, Kashid A, Kshirsagar P, Kulkarni A, Londe P (2023) Media player controller using hand gestures. In: Proceedings of the mobile computing and sustainable informatics, pp 363–373
20. Oudah M, Al-Naji A, Chahl J (2020) Hand gesture recognition based on computer vision: a review of techniques. *J Imaging* 6(8):73
21. Chaudhry M, Kumar S, Ganie SQ (2023) Music recommendation system through hand gestures and facial emotions. In: Proceedings of the 2023 6th international conference on information systems and computer networks (ISCON), pp 1–7
22. Bakariya B, Singh A, Singh H, Raju P, Rajpoot R, Mohbey KK (2024) Facial emotion recognition and music recommendation system using CNN-based deep learning techniques. *Evol Syst* 15(2):641–658
23. Wijekoon R, Ekanayaka D, Wijekoon M, Perera D, Samarasinghe P, Seneweera O, Peiris A (2021) Optimum music: gesture controlled, personalized music recommendation system. In: Proceedings of the 2021 IEEE 16th international conference on industrial and information systems (ICIIS), pp 23–28
24. Shukla A, Katiyar D, Goel G (2022) Gesture recognition-based AI virtual mouse. *Int J Res Adv Eng Sci Technol* 10
25. Riad MOF, Ghosh S (2022) Developing music recommendation system by integrating an MGC with deep learning techniques. *Eurasia Proc Sci Technol Eng Math* 19:87–100
26. Lafci MT, Strzebkowski R, Chojecki P, Bosse S (2023) An evaluation of hand interaction metaphors for immersive environments. In: Proceedings of the 2023 15th international conference on quality of multimedia experience (QoMEX), pp 232–235
27. Spittle B, Frutos-Pascual M, Creed C, Williams I (2022) A review of interaction techniques for immersive environments. *IEEE Trans Vis Comput Graph*

# Machine Learning for Power Analysis: A New Paradigm in CMOS VLSI Design



Naiyya Mittal, Srishty Sharma, Tithi Pandey, and Shobha Sharma

**Abstract** This study explores the use of machine learning algorithms to estimate power consumption in CMOS circuits, aiming to offer a quicker and more efficient alternative to traditional simulation-based approaches. The research evaluates several algorithms, including Support Vector Machines (SVM), Linear Regression, and Random Forest, comparing them based on accuracy metrics, Root Mean Square Error (RMSE), and Mean Squared Error (MSE). The analysis also includes residual plots to assess the models' performance in capturing data patterns. Results indicate that Random Forest outperforms the other algorithms, followed closely by SVM, with Linear Regression delivering acceptable estimates. These findings underscore the promise of machine learning techniques in improving the accuracy and efficiency of power estimation in integrated circuit design.

**Keywords** CMOS · MSE · RMSE · Random Forest · SVM · Machine learning

## 1 Introduction

In recent years, the field of Very Large Scale Integration (VLSI) design has witnessed a significant paradigm shift with the integration of machine learning (ML) techniques for power estimation in complementary metal-oxide-semiconductor (CMOS)

---

Srishty Sharma, Tithi Pandey, and Shobha Sharma contributed equally to this work.

---

N. Mittal (✉) · S. Sharma · T. Pandey · S. Sharma

Electronics and Communication, Indira Gandhi Delhi Technical University for Women, Madrasa Road, Delhi 110006, India

e-mail: [naiyya134btece20@igdtuw.ac.in](mailto:naiyya134btece20@igdtuw.ac.in)

S. Sharma

e-mail: [srishty135btece20@igdtuw.ac.in](mailto:srishty135btece20@igdtuw.ac.in)

T. Pandey

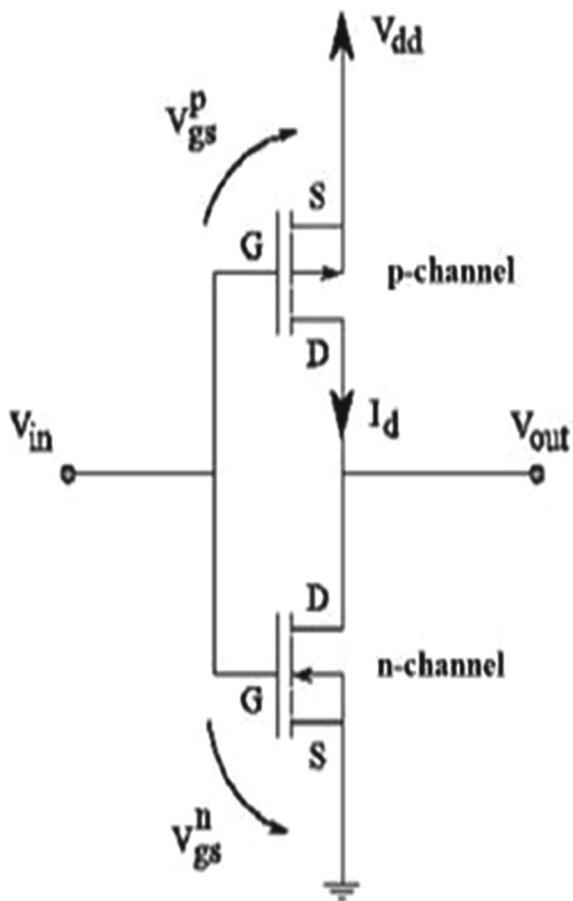
e-mail: [tithi152btece20@igdtuw.ac.in](mailto:tithi152btece20@igdtuw.ac.in)

S. Sharma

e-mail: [shobhasharma@igdtuw.ac.in](mailto:shobhasharma@igdtuw.ac.in)

circuits. VLSI design, concerned with the fabrication of integrated circuits (ICs) containing millions to billions of transistors on a single chip, plays a pivotal role in modern electronic systems, spanning from consumer electronics to advanced computing architectures.

Traditional methods for power estimation in CMOS circuits heavily rely on complex and time-consuming simulations, involving extensive circuit-level analyses and transistor-level models. However, with the ever-increasing complexity and scale of integrated circuits, there arises a pressing need for efficient and accurate power estimation methodologies. This demand has led researchers and practitioners to explore alternative approaches, leveraging the capabilities of machine learning. Machine learning, a subset of artificial intelligence, encompasses algorithms and techniques that enable computers to learn from data and make predictions or decisions without explicit programming. ML algorithms can identify patterns and relationships within datasets, allowing them to generate predictive models that capture the underlying behavior of complex systems. In the context of VLSI design, machine learning offers a promising avenue for power estimation by leveraging historical data from simulations or measurements. By training ML models on datasets containing various circuit parameters and corresponding power consumption values, these models can learn the intricate relationships between input features and power consumption, thus enabling accurate predictions for unseen circuit configurations. The integration of machine learning in power estimation tasks brings several advantages. Firstly, ML-based approaches can significantly reduce the computational burden associated with traditional simulation-based methods, thereby accelerating the design cycle and time-to-market for ICs. Secondly, ML models have the potential to capture non-linear dependencies and interactions among circuit parameters, which may be challenging to model using conventional analytical techniques. Additionally, ML-based power estimation techniques can adapt and generalize to new circuit designs or technology nodes, making them versatile and applicable across a wide range of VLSI applications. In addition to the transformative impact of machine learning on VLSI design, it is imperative to recognize the evolving landscape of CMOS technology and its implications on power estimation methodologies. With the relentless pursuit of miniaturization and performance enhancement in semiconductor fabrication, CMOS circuits continue to evolve, presenting new challenges and opportunities for power estimation. Advanced process nodes, such as 7nm and below, introduce unique characteristics and constraints that necessitate innovative approaches to power modeling and analysis. Moreover, emerging technologies like FinFETs and nanowires bring novel device architectures that demand sophisticated power estimation techniques capable of capturing their intricacies accurately. Furthermore, the integration of machine learning in power estimation heralds a paradigm shift in design methodologies and industry practices. Beyond its applications in power estimation, ML holds promise for optimizing various aspects of VLSI design, including performance, area, and reliability. Moreover, ML-driven design automation tools empower designers to explore a vast design space efficiently, accelerating the innovation cycle and fostering creativity in circuit design (Fig. 1).

**Fig. 1** CMOS pictorial view

## 2 Background Details

In the domain of power estimation for CMOS circuits, two primary methodologies emerge: non-simulating methods and simulating techniques. Non-simulating methods rely on statistical empirical equations, static power calculations, and dynamic power calculations to estimate power consumption without necessitating actual circuit simulations. Empirical equations, derived from statistical analyses of circuit characteristics, encompass both static and dynamic power components. While computationally efficient, these equations may exhibit reduced accuracy, especially in intricate circuit designs or advanced technology nodes. Static power calculations focus on estimating leakage power based on transistor parameters like threshold voltage and gate oxide thickness, while dynamic power calculations consider power dissipation during transitions between logic states, factoring in capacitance and switching frequency. Conversely, simulating techniques employ tools such

as LTspice, MATLAB, and Monte Carlo simulations to perform detailed circuit simulations. LTspice, a prevalent tool, employs SPICE algorithms to simulate CMOS circuit behavior under various conditions, yielding accurate estimates of both static and dynamic power components. MATLAB offers a versatile environment for circuit simulation, enabling the development of custom scripts or utilization of built-in functionalities like Simulink and Simscape for power estimation. Monte Carlo simulations utilize statistical sampling to evaluate power estimation variability and uncertainty, generating probabilistic power consumption distributions. These simulations are invaluable for assessing circuit design robustness against variations and environmental factors. In academic research or technical papers, this description offers a succinct overview of non-simulating and simulating techniques for power estimation in CMOS circuits. It elucidates the fundamental principles underlying each methodology, providing insights into their strengths and limitations. Such clarity aids readers in comprehending the methodologies utilized for power estimation and their implications for circuit design and optimization. For power estimation in CMOS circuits using supervised machine learning, various algorithms can be applied. These include Linear Regression, Decision Tree, Random Forest, Logistic Regression, and Support Vector Machine (SVM), each with its own strengths in modeling the relationship between circuit parameters and power consumption. Linear Regression is a fundamental algorithm that assumes a linear relationship between input features and the target variable, which in this case is power consumption. It estimates coefficients to fit the data points and predicts power consumption based on given features. Decision Trees offer versatility by recursively partitioning the feature space, making them adept at capturing complex interactions among features. This allows them to provide accurate estimates of power consumption by dividing the data into smaller subsets based on feature values. Random Forest, an ensemble method, combines multiple decision trees to enhance predictive accuracy. By aggregating predictions from individual trees, Random Forest reduces overfitting and handles noise effectively, making it suitable for power estimation tasks in CMOS circuits. Logistic Regression, primarily used for classification, can be adapted for regression tasks. It models the probability distribution of power consumption based on input features, providing insights into different power consumption levels. Support Vector Machine (SVM) excels in finding optimal hyperplanes to separate classes in classification tasks. In regression, SVM identifies the hyperplane that best separates different power consumption levels based on input features, making it effective for capturing complex relationships in high-dimensional spaces. Implementation of these algorithms typically involves utilizing machine learning libraries such as Scikit-learn in Python. Performance evaluation is conducted using metrics like Mean Absolute Error (MAE), Mean Squared Error (MSE), or Root Mean Squared Error (RMSE) on validation datasets to select the most suitable model for power estimation in CMOS circuits. Fine-tuning hyperparameters and selecting relevant features can further enhance the predictive capability of these models.

### 3 Experimental Setup and Methodology

#### 3.1 Dataset Description

This research focuses on power estimation in CMOS VLSI circuits through machine learning techniques, utilizing the ISCAS'89 benchmark circuit dataset. The dataset was sourced from previous studies, particularly the work of V. Govindaraj and B. Arunadevi. It includes crucial attributes for sequential circuits, such as the number of inputs (IN), outputs (OUT), D flip-flops (DFF), inverters (INV), and various gate types like AND, NAND, OR, and NOR gates. Additionally, power dissipation values for different input patterns were collected using the SIS (Synthesis and Optimization of Sequential Circuits) tool via zero-delay simulations. This dataset provides a comprehensive basis for analyzing power consumption across different circuit configurations.

#### 3.2 Data Partitioning

To effectively train and evaluate our machine learning models, we partitioned the dataset into two subsets. The first subset, comprising 20 benchmark ISCAS'89 sequential circuits, was used to train the models. The second subset, consisting of 5 circuits, was used for testing the models. This partitioning approach ensures that the models are tested on data they have not encountered during training, which provides an objective measure of their ability to generalize. By using a distinct testing set, we can more accurately assess each model's performance on unseen data.

#### 3.3 Machine Learning Models

We investigated several machine learning models to determine their suitability for predicting power consumption in CMOS circuits. The models included Linear Regression, Decision Tree, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Random Forest. Each model has unique strengths. Linear Regression is straightforward and interpretable, making it ideal for initial regression analyses. Decision Trees and Random Forests can model non-linear relationships and feature interactions. SVMs are effective in high-dimensional spaces and can capture non-linear patterns with kernel functions. KNN, a straightforward model, predicts outcomes based on the proximity of data points.

### 3.4 Model Training and Evaluation

Each model was trained using the training subset of the dataset. We used Mean Squared Error (MSE), Root Mean Squared Error (RMSE), R-squared ( $R^2$ ), and Cross-Validation Score as performance metrics. MSE measures the average squared difference between actual and predicted values, providing a key indicator of prediction accuracy. RMSE, as the square root of MSE, offers a direct interpretation of prediction errors.  $R^2$  reflects the proportion of variance in the dependent variable explained by the model, while the Cross-Validation Score evaluates the model's robustness and generalizability.

### 3.5 Residual Analysis

To further understand the models' performance, we conducted a residual analysis. Residual plots were created to visualize the residuals (differences between actual and predicted values) against the predicted values. Ideally, residuals should be randomly scattered around zero, indicating that the model's errors are evenly distributed. This randomness suggests a good fit, as it implies the model does not systematically overestimate or underestimate predictions at any level of the predicted values.

For Linear Regression, the residual plot typically shows residuals centered around zero, though some deviations can occur, indicating potential areas where the model could be refined. The absence of a clear pattern in the residuals suggests that the linear regression model does not exhibit systematic biases, although heteroscedasticity (uneven spread of residuals) might still be present.

For more complex models like Random Forest, the residual plot can reveal how well the model captures non-linear relationships. Random Forest models tend to show residuals that are also centered around zero but might exhibit larger residuals at higher predicted values. This could indicate that while the model captures the overall trends well, certain individual predictions still have significant errors. The distribution of residuals can be somewhat symmetric but may display deviations due to the complexity of interactions between features.

## 4 Final Evaluation

### 4.1 Mean Squared Error (MSE)

In our evaluation of power estimation algorithms for CMOS circuits, Mean Squared Error (MSE) emerged as a critical metric, indicating the average squared difference between actual and predicted power values. The Linear Regression model exhibited a low MSE of 0.0017, suggesting a strong fit to the training data. Conversely, the

Random Forest model demonstrated a slightly higher MSE of 0.0046, implying a comparable level of accuracy. Other algorithms, including SVM, Decision Tree, and K-Nearest Neighbors, yielded MSE values of 0.0076, 0.0055, and 0.0107, respectively. While Random Forest captured more complex data patterns, it did not outperform Linear Regression in terms of MSE. Overall, Linear Regression and Random Forest emerged as the top-performing algorithms, with MSE values of 0.0017 and 0.0046, respectively.

## 4.2 Root Mean Squared Error (RMSE)

The Root Mean Squared Error (RMSE) provides an interpretable measure of prediction error, representing the square root of the average squared errors. For Linear Regression, the RMSE was 0.04128052686303434, while for Random Forest, it was 0.0333364191421635. Despite the higher MSE, the Random Forest model had a lower RMSE, indicating that its predictions were, on average, closer to the actual values. This discrepancy suggests that Random Forest handles certain predictions more accurately, even if its overall error distribution is broader.

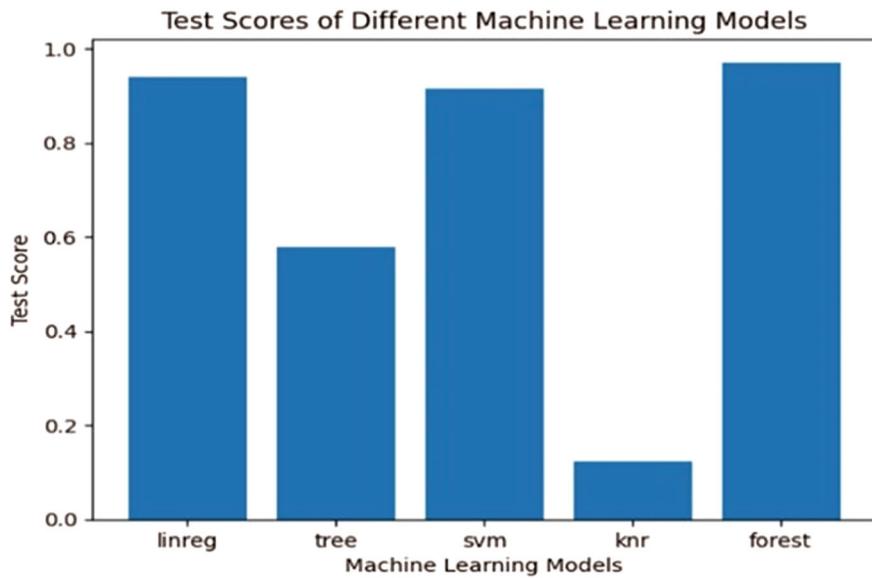
## 4.3 Residual Plot Analysis

Residual plots were instrumental in diagnosing the performance of both models. For Linear Regression, the residual plot displayed residuals largely clustered around zero, with some dispersion. This pattern indicates that the model performs consistently across different levels of predicted values but may have heteroscedasticity, as residuals tend to vary more at lower predicted values (Figs. 2 and 3).

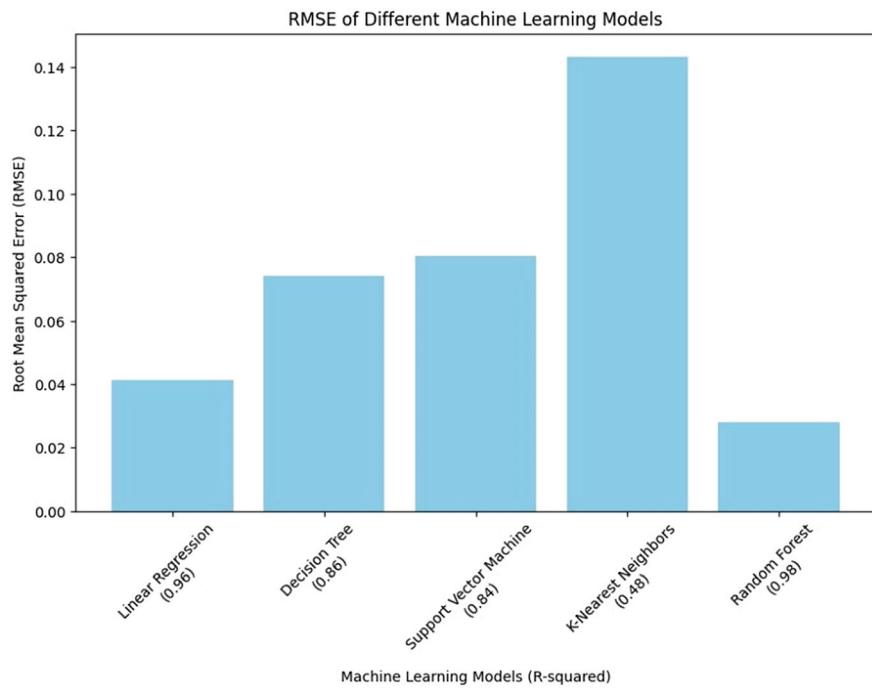
The Random Forest residual plot also showed residuals centered around zero, though with a few larger deviations at higher predicted values. This suggests that while Random Forest captures the underlying data structure well, it occasionally makes significant errors, especially with higher power consumption predictions. However, the absence of a clear pattern in the residuals for both models indicates that neither model suffers from severe biases or systematic errors.

## 4.4 Accuracy

Evaluating model accuracy based on the discrepancy between the Monte Carlo value and the model-generated values reveals insightful perspectives. Linear Regression closely aligns with the Monte Carlo value, indicating a high level of accuracy in its predictions. However, the Random Forest model displays a deviation from the



**Fig. 2** Test score comparison



**Fig. 3** RMSE comparison

Monte Carlo value, suggesting potential limitations in accuracy compared to Linear Regression.

## 5 Conclusion

After a thorough examination, the analysis reveals that while both the Linear Regression and Random Forest models display promising performance for power estimation in CMOS VLSI circuits, the Random Forest model emerges as the preferable choice. Despite Linear Regression demonstrating a competitive Mean Squared Error (MSE), the Random Forest model excels with a lower Root Mean Squared Error (RMSE) and a higher R-squared value, indicating superior predictive accuracy and explanatory power.

Additionally, the inspection of residual plots provides valuable insights. Both models exhibit residuals clustered around zero, suggesting overall adequacy. However, the Random Forest model demonstrates fewer significant deviations at higher predicted values, indicating better performance in capturing complex data relationships.

Based on the residual plot, residual distribution, feature importance, and the MSE value, it appears that the Random Forest model is not as good as the linear regression model for your dataset. The higher MSE value indicates that the Random Forest model has a worse fit compared to the linear regression model.

However, Random Forest models are generally more robust to outliers and can capture non-linear relationships better than linear regression models. If your dataset has non-linear relationships or interactions between features, the Random Forest model might still be a good choice despite the higher MSE value.

In summary, while Linear Regression offers simplicity and respectable performance, the Random Forest model's ability to provide more accurate predictions and capture intricate data structures makes it the recommended choice for power estimation in CMOS VLSI circuits [1–12].

## 6 Future Scope

In our future endeavors, we aspire to elevate our modeling approach beyond conventional performance metrics like RMSE and MSE, prioritizing comprehensive accuracy across multiple dimensions. To achieve this, we intend to delve deeper into feature engineering, leveraging domain-specific insights and sophisticated techniques to extract meaningful predictors. Additionally, our exploration will encompass a diverse array of algorithms and ensemble methods, including gradient boosting and stacking, to harness the collective intelligence of multiple models. Hyperparameter tuning will play a pivotal role, as we meticulously optimize model configurations to strike a balance between bias and variance, ultimately enhancing generalization

Model	Test Score	RMSE	MSE	Accuracy
Linear Regression	0.956909	0.041281	0.001704	0.005969
Random Forest	0.971898	0.033336	0.001111	0.005276

**Fig. 4** Comparison between linear regression and random forest (accuracy—comparative with Monte Carlo method)

capabilities. Furthermore, we are committed to addressing data intricacies such as imbalance and outliers, employing tailored strategies like oversampling and robust algorithms to ensure equitable representation and robustness. Our evaluation framework will extend beyond traditional measures, embracing alternative metrics like mean absolute error and precision-recall curves to provide a nuanced understanding of model performance. Domain-specific validation will be paramount, enabling us to validate the model's efficacy in diverse scenarios and real-world applications. Moreover, we recognize the importance of continuous model monitoring and refinement, establishing protocols for ongoing updates and adaptation to evolving data landscapes. By integrating these strategies, our future model endeavors to not only achieve superior accuracy but also deliver reliable predictions that resonate with the complexities of real-world contexts (Fig. 4).

**Acknowledgements** We extend our sincere appreciation to Dr. Shobha Sharma for their invaluable guidance, mentorship, and unwavering support, Indira Gandhi Delhi Technical University for women for providing us with the necessary resources, facilities, and funding, colleagues and peers for their valuable insights, feedback, and assistance throughout this research endeavor. Furthermore, we acknowledge the authors of the research publications referenced in this paper for their seminal contributions, which have provided the foundation for our research and inspired us to explore new avenues in the field of machine learning-based power estimation for CMOS circuits.

## References

1. Govindaraj V, Arunadevi B (2021) Machine learning based control estimation for CMOS VLSI circuits
2. Hou L, Zheng L, Wu W (2006) Neural network based VLSI power estimation. In: 2006 8th international conference on solid-state and integrated circuit technology proceedings. <https://doi.org/10.1109/icsict.2006.306506>
3. Kozhaya JN, Najm FN (2001) Control estimation for expansive successive circuits. IEEE Exch Except Expans Scale Integr Fram 9(2):400–407. <https://doi.org/10.1109/ICCAD.1997.643581>
4. Liang HG, Zheng T (2019) Real-time impedance estimation for control line communication. IEEE Get to 7:88107–88115. <https://doi.org/10.1109/ACCESS.2019.2925464>
5. Harris CJ (1994) Propels in brilliantly control. CRC Press. ISBN: 9780748400669
6. Kirei SCF, Topa MD (2019) Control and zone estimation of discrete channels in CMOS coordinates circuits. In: Flag handling: calculations, structures, courses of action, and applications (SPA), pp 67–70. <https://doi.org/10.23919/SPA.2019.8936762>

7. Burch R, Najm FN, Yang P, Trap TN (1993) A Monte Carlo approach for control estimation. IEEE Exch Except Huge Scale Integr Fram 2(1):63–71. <https://doi.org/10.1109/92.219908>
8. Buyuks MK, Najm FN (2006) Early control estimation for VLSI circuits. IEEE Exch Comput-Aided Plan Coord Circuits Fram 24(7):1076–1088. <https://doi.org/10.1109/TCAD.2005.850904>
9. Das AK, Dalai S, Chatterjee B (2020) Cross stockwell change sup-ported arbitrary woodland based surface condition recognizable proof of metal oxide surge arrester utilizing spillage current flag. In: 2020 IEEE locale 10 symposium (TENSYMP). IEEE, pp 1775–1778. <https://doi.org/10.1109/TENSYMP50017.2020.9230802>
10. Chen B, Chen P, Belkacem AN, Lu L, Xu R, Tan W, Wang C (2020) Neural exercises classification of cleared out and right finger signals amid engine execution and engine symbolism. Brain-Comput Interfacing 1–11. <https://doi.org/10.1080/2326263X.2020.1782124>
11. <https://en.wikipedia.org/wiki/CMOS>
12. <https://www.statisticshowto.com/residual-plot/>

# An Ensemble-Based Lexicon Dictionary Coupled with Annotated Fine-Grained Emotions and Sentiments



Shelley Gupta and Archana Singh

**Abstract** Emotions are the backlash that a person confronts in return for an act or bearing from their surroundings. Emotion detection is a cognitive theory that advises a person's speculations that supervise their emotions with respect to behavioral and psychological responses. The expressions posted by a person over social media are the supreme source of emotion detection nowadays. The proposed work aims at the formation of a lexicon dictionary, ESentiEmo, with 22 fine-grained emotions and 3 sentiments, consisting of 17,812 words. The proposed fine-grained dictionary has been annotated with these fine-grained emotions and sentiments. The ensemble-based deep learning models are created to effectuate the accuracy of the proposed lexicon dictionary. The proposed ensemble architecture comprises of three deep learning base architectures, namely, Long Short-Term Memory (LSTM), Support Vector Machine (SVM), and Convolution Neural Network (CNN). Our suggested framework has achieved maximal accuracy with 81.02% employing an ensemble approach coupled with maximum voting.

**Keywords** Lexicon dictionary · Annotation · Fine-grained emotions · Sentiments · ESentiEmo

## 1 Introduction

Sentiment analysis, also known as opinion mining, is a field within natural language processing (NLP) that involves determining the sentiment expressed in a piece of text. It aims to identify the emotional tone behind a body of text, which can be positive, negative, neutral, or more nuanced [1, 2].

---

S. Gupta (✉)

Department of Information Technology, ABES Engineering College, Ghaziabad, UP, India  
e-mail: [shelley.g17@gmail.com](mailto:shelley.g17@gmail.com)

A. Singh

Senior Data Scientist, Caliper, Noida, India  
e-mail: [archana.elina@gmail.com](mailto:archana.elina@gmail.com)

The process of human emotion identification is known as emotion recognition. Emotion detection, a subset of sentiment analysis, focuses on identifying and categorizing the emotions expressed in textual data. Unlike general sentiment analysis, which primarily identifies whether text is positive, negative, or neutral, emotion detection aims to pinpoint specific emotional states [1, 2]. Emotion detection typically categorizes text into basic emotions such as: joy, sadness, anger, fear, surprise, disgust, trust, and anticipation.

Emotion recognition is a unification of neuroscience and cognitive science, especially where the input is in the form of linguistics [1, 3]. The models of emotions majorly include Ekman's basic emotion [4], Plutchik wheel of emotions [5], circumplex model of emotion [6], etc. These models have provided multiple analogs to emotion classes. The existing research work majorly incorporates the identification of basic emotions of Ekman's or few more [7, 8].

Thus, the central aim of the research work proposed incorporates:

- Curation of lexicon dictionary, ESentiEmo consisting of 17,812 lexicons in English.
- Labeling of lexicon dictionary, ESentiEmo with 22 fine-grained emotions and sentiment (positive, negative, and neutral) classes.
- Employing the various classifiers and an ensemble-based approach to determine the finest model to attain the best accuracy.
- The proposed lexicon dictionary may help in analyzing the state of anxiety, depression, suicide, etc.

The above section has given an overview of introduction, Sect. 2 discusses literature review and Sect. 3 elaborates about lexicon dictionary formation and annotation. Sect. 4 details lexicon dictionary analysis, Sect. 5 extracts the ensemble approach employed, Sect. 6 refines results, and Sect. 7 shows the conclusion with further future work scope.

## 2 Literature Review

Sentiment analysis, also known as opinion mining, involves determining the sentiment expressed in a piece of text. The main types of sentiment analysis are [9]:

- **Document-Level Sentiment Analysis:** Analyzes the sentiment of an entire document or piece of text. The goal is to classify the overall sentiment of the document as positive, negative, or neutral.
- **Sentence-Level Sentiment Analysis:** Examines the sentiment of individual sentences. Each sentence is classified as expressing positive, negative, or neutral sentiment.

- **Aspect-Based Sentiment Analysis (ABSA):** Identifies and extracts opinions on specific aspects or features of an entity within the text. For example, in a product review, the analysis might identify sentiments about the battery life, screen quality, and price of a smartphone.

Emotion detection is a specialized subset of sentiment analysis that focuses on identifying and categorizing emotions expressed in textual data. Unlike traditional sentiment analysis, which typically classifies text as positive, negative, or neutral, emotion detection aims to recognize specific emotional states [1, 10–13].

#### a. Applications of Emotion Detection

- Customer Feedback Analysis: Understanding customer emotions in reviews and feedback to improve products and services.
- Social Media Monitoring: Tracking public sentiment and emotions on platforms like Twitter, Facebook, and Instagram to gauge reactions to events, products, or policies.
- Mental Health Monitoring: Analyzing text from social media posts, chat messages, or journal entries to identify signs of emotional distress or well-being.
- Human-Computer Interaction: Enhancing user experience by making systems responsive to the user's emotional state.
- Marketing and Advertising: Tailoring content and campaigns based on the emotional responses of target audiences.

#### b. Challenges in Emotion Detection

- Ambiguity and Subjectivity: Emotions are often subjective and context-dependent, making them challenging to detect accurately.
- Sarcasm and Irony: These can mislead models as the literal meaning of the words used may not reflect the true emotion.
- Multilingual and Multicultural Variations: Emotions can be expressed differently across languages and cultures, requiring models to be adaptable and culturally aware.
- Data Scarcity: Labeled datasets for specific emotions can be limited, particularly for less commonly studied emotions.

#### c. Tools and Libraries

- NLTK (Natural Language Toolkit): Provides basic tools for text processing and emotion lexicons.
- SpaCy: A powerful NLP library that can be used with emotion detection models.
- Transformers Library (by Hugging Face): Includes pre-trained models like BERT and GPT-3 that can be fine-tuned for emotion detection tasks.

Emotion detection is a rapidly evolving field within NLP, with ongoing research and development aimed at improving the accuracy and applicability of emotion detection systems across various domains.

Emotion detection dictionaries, also known as emotion lexicons, are collections of words and phrases that are annotated with their corresponding emotional associations. These dictionaries are used to identify and quantify emotions expressed in text by matching words in the text with entries in the lexicon. Here are some of the most widely used emotion detection dictionaries.

a. **NRC Emotion Lexicon (EmoLex) [14]**

The NRC Emotion Lexicon is one of the most comprehensive and widely used emotion lexicons. It includes a large number of words annotated with eight basic emotions and sentiments:

*Emotions:* Anger, Anticipation, Disgust, Fear, Joy, Sadness, Surprise, Trust.

*Sentiments:* Positive, Negative.

b. **WordNet-Affect [15]**

WordNet-Affect is an extension of the WordNet database, specifically annotated with affective concepts. It includes words tagged with various emotional categories:

*Emotional Categories:* Joy, sadness, anger, fear, disgust, surprise, etc.

*Other Affect Dimensions:* Emotional valence, arousal, and dominance.

iii. **LIWC (Linguistic Inquiry and Word Count) [16]**

LIWC is a text analysis software that includes an extensive dictionary with categories for various psychological constructs, including emotions:

*Emotional Categories:* Positive emotion, negative emotion, anxiety, anger, sadness, etc.

d. **ANEW (Affective Norms for English Words) [17]**

ANEW provides normative emotional ratings for a large set of English words. Words are rated on:

*Valence:* Pleasantness of the word (happy-sad dimension).

*Arousal:* Intensity of emotion provoked by the word (calm-excited dimension).

*Dominance:* Control or influence associated with the word (controlled-in-control dimension).

v. **SentiWordNet [18, 19]**

SentiWordNet is an extension of the WordNet database where each synset (set of cognitive synonyms) is annotated with sentiment scores:

*Scores:* Positive, negative, and objective.

#### f. General Inquirer [20, 21]

The General Inquirer lexicon includes categories for various psychological and sociological constructs, including emotions:

*Categories:* Positive, negative, pleasure, arousal, dominance, etc.

The research work [7] has formed a fine-grained suicide notes corpus in English consisting of 15 labels of emotions, particularly (hopefulness, forgiveness, peacefulness, pride, sorrow, hopelessness, guilt, fear, instructions, information, happiness, love, anger, thankfulness, love, blame, abuse) only.

The FED dataset is curated in research work [22] comprising of original and fake emotion images. It comprises of 12 emotion classes, namely, sadness, surprise, anger, fear, disgust, happiness, fake surprise, fake anger, fake fear, fake sadness, fake disgust, and fake happiness.

The research work [1] aims at detecting depression by means of analysis of the social media posts of the person [2, 23]. It forms a dictionary consisting of 2499 words annotated with 10 non-depressive and 13 depressive emotions. Thus, the dictionary is formed with the major task as depression detection.

The research work [8] is based on rule-based emotion detection and analysis at sentence level integrating cognitive-based emotion theory. The created extended emotion sentiment lexicon, EESL consists of eight emotion class only.

Whereas the research work [24, 25] utilized lexicons of sentiments for the evaluation of the sentence polarity equipped with linguistics, namely, text, slang, or emojis [9].

This determines the requirement of a fine-grained lexicon dictionary annotated with an enormous number of fine-grained emotions and sentiments.

### 3 ESentiEmo: Lexicon Dictionary Formation and Annotation

The proposed research work aims at construction of an English language-based sentiment and emotion (ESentiEmo) lexicons consisting of 17812 lexicons. Our research work annotated this lexical dictionary with the further 11 positive emotions (Powerful, Energetic, Reactive, Anticipate, Confident, Trust, Happy, Proud, Caring, Generous, Serene), 11 negative emotions (Fear/Threatened, Powerless, Ignorant, Surprise, Confused, Jealous, Sad, Embarrass, Hurt, Disgust, Anger) along with sentiments (positive, negative and neutral). Thus, the lexicon dictionary consists of 22 unique positive and negative emotions in totality.

The mentioned dictionary is annotated by three proficient English language annotators. The annotators do not have any health issues and they were also eager to contribute to annotation process. The process of annotation is observed immensely and the annotators were not allowed to interact among themselves.

The measure used for finding the agreement of annotation among annotators is Cohen's Kappa coefficient [1]:

$$k = (P_o - P_e) / (1 - P_e) \quad (1)$$

where,

$P_o$  = The number of annotations in which the raters agree/total number of annotations

$P_e$  = Probability of chance agreement.

The proposed annotation of lexicon-based dictionary has acquired an average agreement of 89.2% predicting the good annotation quality. The concept of majority voting is pre-owned to evaluate the concluding annotation of the dictionary annotated by three annotators.

The sample of annotated lexicon dictionary is provided in Table 1.

## 4 ESentEmo: Lexicon Dictionary Analysis

Fig. 1 and Table 2 represent the count of lexicons belonging to various emotion classes. Fig. 2 and Table 3 represent the count of lexicons belonging to positive, negative, and neutral sentiments. In these figures, we have used red color to represent the negative emotions and sentiments, green color for positive emotions and sentiments, and blue color for neutral sentiments.

## 5 Ensemble Approach Employed

Ensemble learning coupled with the learning of numerous models of neural networks, diminishes the divergence of predictions and generalization error. The benefits and strengths of different neural networks used in ensemble learning improve the predictive accuracy and reduce the model prediction dispersion [1]. The suggested model has utilized three deep learning base models for training, namely, CNN (Convolution Neural Network), SVM (Support Vector Machine), and LSTM (Long Short-Term Memory) and one ensemble bagging model.

The word embedding, fastText and vector representation has been used in the proposed framework [26]. It is an extension of Word2Vec word embedding. It provides an improvement over Out of Vocabulary (OOV) words and morphology. It captures syntactic and semantic relations among words.

The selected bagging ensemble approach [1] is employed with majority voting. The final prediction is the prediction that is voted from the majority of classifiers.

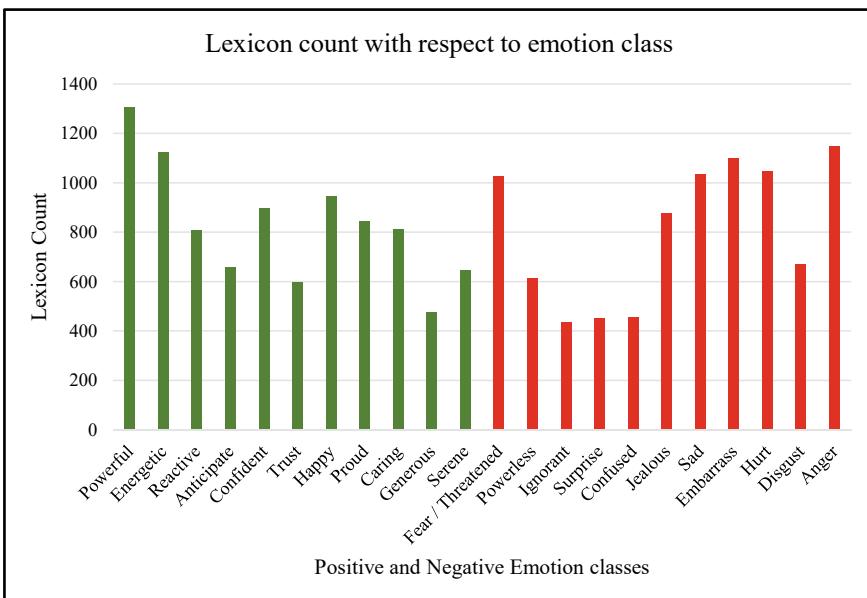
**Table 1** Sample of annotated words with 22 critical positive and negative emotion class along with sentiment class

Annotated lexicon sample	Sentiment class	Emotion class with lexicon count	Emotion class type
Authoritative, capable, compelling, dominant, forceful, influential, mighty, persuasive, robust	Positive	Powerful (1305)	Positive
Strong, tireless, vigorous, spirited, active, dynamic, industrious, powerful, sprightly, spry, animated	Positive	Energetic (1124)	Positive
Active, aware, compassionate, conscious, receptive, Susceptible, awake, impressionable, influenceable, passionate, perceptive, persuadable, sharp, warm, warmhearted	Positive	Reactive (806)	Positive
Assume, await, count on, forecast, foresee, prepare for, see, foretell, promise oneself	Positive	Anticipate (658)	Positive
Assured, convinced, positive, sure, optimistic, counting on, expectant, having faith in, secure	Positive	Confident (899)	Positive
Expectation, faith, assurance, certainty, reliance, sureness, entrustment	Positive	Trust (598)	Positive
Cheerful, delighted, elated, glad, joyful, joyous, jubilant, merry, overjoyed, thrilled	Positive	Happy (947)	Positive
Appreciative, great, honored, illustrious, dignified, eminent, rewarded	Positive	Proud (846)	Positive
Friendly, loving, sympathetic, warm, warmhearted	Positive	Caring (813)	Positive
Benevolent, charitable, considerate, fair, helpful, honest, hospitable, unselfish, willing, thoughtful	Positive	Generous (476)	Positive
Poised, sedate, tranquil, composed, cool, easygoing, peaceful, quiet, smooth, laid-back, placid	Negative	Serene (647)	Negative
Angst, anxiety, doubt, dread, horror, panic, suspicion, terror, worry exposed, vulnerable, imperiled, jeopardized, warned, in danger, unprotected, unsafe	Negative	Fear/threatened (1026)	Negative
Defenseless, disenfranchised, helpless, impotent, incapable, paralyzed, passive, vulnerable	Negative	Powerless (613)	Negative
Illiterate, innocent, naive, oblivious, obtuse, uneducated, uninformed	Negative	Ignorant (434)	Negative
Amazement, astonishment, bewilderment, consternation, curiosity, disappointment, jolt, miracle, shock, wonder	Negative	Surprise (450)	Negative
Puzzled, baffled, bewildered, disorganized, distracted, muddled, perturbed, befuddled, perplexed, dazed	Negative	Confused (456)	Negative
Apprehensive, envious, intolerant, possessive, protective, resentful, skeptical, begrudging, doubting	Negative	Jealous (878)	Negative
Dismal, heartbroken, melancholy, mournful, sorrowful, sorry, unhappy, cheerless, distressed	Negative	Sad (1035)	Negative

(continued)

**Table 1** (continued)

Annotated lexicon sample	Sentiment class	Emotion class with lexicon count	Emotion class type
Give a bad time, hang up, abash, dumbfound, give a hard time	Negative	Embarrass (1100)	Negative
Aching, bleeding, burned, crushed, bruised, cut, disfigured, hit, lacerated, contused, damaged, harmed, impaired	Negative	Hurt (1047)	Negative
Antipathy, aversion, dislike, distaste, hatred, repulsion, revulsion, hatefulness, detestation	Negative	Disgust (669)	Negative
Acrimony, animosity, annoyance, enmity, violence, irritation, outrage, hatred	Negative	Anger (1147)	Negative

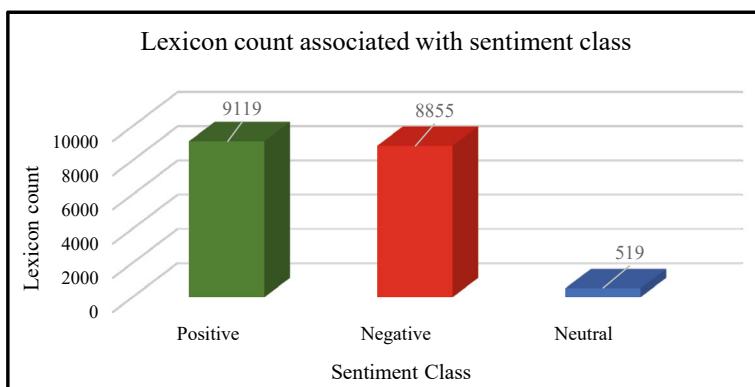
**Fig. 1** Depicts the count of lexicons belonging to fine-grained 22 emotion classes

## 6 Results

Using the 10-fold cross-validation, the best-acquired accuracy is 81.02% within the bagging ensemble approach coupled with max voting (Table 4).

**Table 2** The count of lexicons associated with fine-grained 22 emotion classes

Emotion class	Emotion class type	Number of lexicons associated	Emotion class	Emotion class type	Number of lexicons associated
Powerful	Positive	1305	Fear/threatened	Negative	1026
Energetic	Positive	1124	Powerless	Negative	613
Reactive	Positive	806	Ignorant	Negative	434
Anticipate	Positive	658	Surprise	Negative	450
Confident	Positive	899	Confused	Negative	456
Trust	Positive	598	Jealous	Negative	878
Happy	Positive	947	Sad	Negative	1035
Proud	Positive	846	Embarrass	Negative	1100
Caring	Positive	813	Hurt	Negative	1047
Generous	Positive	476	Disgust	Negative	669
Serene	Positive	647	Anger	Negative	1147

**Fig. 2** The lexicon count associated with sentiment classes**Table 3** The lexicon count associated with sentiment classes

Sentiment class	Lexicon count
Positive	9119
Negative	8855
Neutral	519

**Table 4** The base models and ensemble model performance

Models	F1-score	Recall	Precision	Accuracy
CNN	0.74	0.72	0.73	0.74
SVM	0.74	0.73	0.75	0.77
LSTM	0.73	0.71	0.72	0.75
Bagging with max voting	0.77	0.76	0.78	<b>0.81</b>

## 7 Conclusion and Future Work

Investigating the social media expression can help in determining the appropriate emotion and sentiment of the person, which can be further utilized in determining the state of mental well-being of the person. Thus, the proposed 22 fine-grained emotions and sentiments (positive, negative, and neutral) annotated-based lexicon dictionary can assist in achieving this.

The research work in the future can be extended by annotating the same or much more enhanced emotions for emojis and slangs.

## References

- Gupta S, Singh A, Ranjan J (2023) Multimodal, multiview and multitasking depression detection framework endorsed with auxiliary sentiment polarity and emotion detection. *Int J Syst Assur Eng Manag* 1–16
- Gupta S, Singh A, Ranjan J (2022) Online document content and emoji-based classification understanding from normal to pandemic COVID-19. *Int J Perform Eng* 18(10)
- Gupta S, Garg O, Mehrotra R, Singh A (2021) Social media anatomy of text and emoji in expressions. In: Smart computing techniques and applications: proceedings of the fourth international conference on smart computing and informatics, vol 2. Springer Singapore, pp 41–49
- Ekman P (1999) Basic emotions. In: Handbook of cognition and emotion, vol 98, no 45–60, p 16
- Plutchik R (1980) A general psychoevolutionary theory of emotion. In: Theories of emotion. Academic Press, pp 3–33
- Russell JA (1980) A circumplex model of affect. *J Pers Soc Psychol* 39(6):1161
- Ghosh S, Ekbal A, Bhattacharyya P (2020, May) Cease, a corpus of emotion annotated suicide notes in English. In: Proceedings of the 12th language resources and evaluation conference, pp 1618–1626
- Asghar MZ, Khan A, Bibi A et al (2017) Sentence-level emotion detection framework using rule-based classification. *Cogn Comput* 9:868–894. <https://doi.org/10.1007/s12559-017-9503-9>
- Gupta S, Singh A, Kumar V (2023) Emoji, text, and sentiment polarity detection using natural language processing. *Information* 14(4):222
- Khare SK, Blanes-Vidal V, Nadimi ES, Acharya UR (2023) Emotion recognition and artificial intelligence: a systematic review (2014–2023) and research recommendations. *Inf Fusion* 102019
- Sharma CM, Damani D, Chariar VM (2023) Review and content analysis of textual expressions as a marker for depressive and anxiety disorders (DAD) detection using machine learning. *Discov Artif Intell* 3(1):38

12. Safari F, Chalechale A (2023) Emotion and personality analysis and detection using natural language processing, advances, challenges and future scope. *Artif Intell Rev* 56(Suppl 3):3273–3297
13. Kumari N, Bhatia R (2023) Deep learning based efficient emotion recognition technique for facial images. *Int J Syst Assur Eng Manag* 14(4):1421–1436
14. Mohammad SM, Turney PD (2013) NRC emotion lexicon, vol 2. National Research Council, Canada, p 234
15. Strapparava C, Valitutti A (2004, May) Wordnet affect: an affective extension of WordNet. In: International conference on language resources and evaluation, vol 4, no 1083–1086, p 40
16. Pennebaker JW, Francis LE, Booth RJ (2001) LIWC: Linguistic inquiry and word count
17. Bradley MM, Lang PJ (1999) ANEW (Affective norms for English words). NIMH Center for Emotion and Attention, University of Florida, Gainesville, FL
18. Esuli A, Sebastiani F (2006) SentiWordNet: a publicly available lexical resource for opinion mining. In: Proceedings of the fifth international conference on language resources and evaluation (LREC'06). European Language Resources Association (ELRA), pp 417–422
19. Li Y, Kazemeini A, Mehta Y, Cambria E (2022) Multitask learning for emotion and personality traits detection. *Neurocomputing* 493:340–350
20. Stone PJ, Dunphy DC, Smith MS, Ogilvie DM (1966) The general inquirer: a computer approach to content analysis. MIT Press, Cambridge, MA
21. Cohen J (1960) A coefficient of agreement for nominal scales. *Educ Psychol Measur* 20(1):37–46
22. Annadurai S, Arock M, Vadivel A (2023) Real and fake emotion detection using enhanced boosted support vector machine algorithm. *Multimed Tools Appl* 82(1):1333–1353
23. Gupta S, Singh A, Ranjan J (2020) Sentiment analysis: usage of text and emoji for expressing sentiments. In: Advances in data and information sciences: proceedings of ICDIS 2019. Springer Singapore, pp 477–486
24. Gupta S, Bisht S, Gupta S (2021) Sentiment analysis of an online sentiment with text and slang using lexicon approach. In: Smart computing techniques and applications: proceedings of the fourth international conference on smart computing and informatics, vol 2. Springer Singapore, pp 95–105
25. Gupta S, Singh A, Ranjan J (2021) Emoji score and polarity evaluation using CLDR short name and expression sentiment. In: Proceedings of the 12th international conference on soft computing and pattern recognition (SoCPaR 2020), vol 12. Springer International Publishing, pp 1009–1016
26. Santos I, Nedjah N, de Macedo Mourelle L (2017, November) Sentiment analysis using convolutional neural network with fastText embeddings. In: 2017 IEEE Latin American conference on computational intelligence (LA-CCI). IEEE, pp 1–5

# Netflix Analysis Using Tableau and ML



E. Elakiya, Leki Chom Thungon, Benoy Joseph, and Manas Kamal Das

**Abstract** The domains where problems are faced in the existing system are Data Volume and Variety, Data Quality and Cleansing, Complex Data Relationships, and Real-time Analysis. Data Volume and Variety of Netflix generates massive amounts of data daily, including user behavior, content preferences, and streaming patterns. Analyzing and making sense of this vast and diverse dataset can be challenging. Data Quality and Cleansing ensures the accuracy, completeness, and consistency of the data that is crucial for meaningful analysis. Data cleaning and preprocessing tasks are time-consuming and require expertise. Complex Data Relationships of Netflix data contains complex relationships between users, content, genres, and other variables. Understanding and visualizing these relationships can be difficult without proper tools and techniques. Real-time Analysis of Netflix operates in a fast-paced environment where real-time analysis is essential. Traditional analytical methods may not be efficient in processing and analyzing data in real time. The paper gives various Netflix analyses using Tableau.

**Keywords** Netflix · Tableau · Statistical analysis · Visualization · Machine learning

---

E. Elakiya (✉) · L. C. Thungon · B. Joseph · M. K. Das  
School of Computer Science and Engineering, VIT Chennai, Chennai, India  
e-mail: [elakiya.e@vit.ac.in](mailto:elakiya.e@vit.ac.in)

L. C. Thungon  
e-mail: [lekichom.thungon@vit.ac.in](mailto:lekichom.thungon@vit.ac.in)

B. Joseph  
e-mail: [benoy.joseph2020@vitstudent.ac.in](mailto:benoy.joseph2020@vitstudent.ac.in)

M. K. Das  
e-mail: [manaskamal.das2022@vitstudent.ac.in](mailto:manaskamal.das2022@vitstudent.ac.in)

## 1 Introduction

Netflix is the most subscribed service provider of films and television shows from various genres, and it is available internationally in multiple languages. The existing analysis of online video streams is using Traditional Business Intelligence Tools, SQL and Data Warehousing and Statistical Analysis Tools. Many organizations use traditional business intelligence tools to analyze Netflix data. These tools provide basic visualization capabilities but may lack advanced features and interactivity. SQL queries and data warehousing techniques are commonly used to extract, transform, and load Netflix data for analysis. These methods offer data aggregation and filtering capabilities. Statistical analysis tools such as R and Python provide advanced statistical modeling and analysis capabilities. They can be used to perform in-depth analyses of Netflix data.

Tableau offers a user-friendly interface with drag-and-drop functionality, enabling users to create interactive visualizations easily. This helps in exploring Netflix data visually and gaining insights quickly. Tableau supports real-time data streaming, enabling users to analyze streaming data from Netflix as it arrives. This feature allows for immediate insights and faster decision-making. Tableau allows users to blend and integrate data from multiple sources, facilitating a comprehensive analysis of Netflix data. It enables combining Netflix data with external data sources to gain deeper insights. Tableau offers storytelling capabilities, allowing users to create interactive dashboards and present data-driven narratives. This feature helps communicate insights effectively and engage stakeholders. Tableau provides cloud-based and mobile solutions, allowing users to access and analyze Netflix data from anywhere, anytime. This enhances collaboration and enables remote data analysis.

A comprehensive data analysis of Netflix is performed using Tableau. The analysis focuses on various aspects, including country, genre, movie or show title, release year, IMDb score, number of upvotes, duration, number of seasons, and content type (movie or show). The paper seeks to utilize Tableau's robust capabilities to extract insights on Netflix content and user preferences, detect patterns, and present a graphical depiction of the data. The paper addresses existing problems in Netflix data analysis, such as data collection and integration challenges, data quality and consistency issues, and limited visualization capabilities. It explores both existing methods employed to solve these problems and proposes new solutions utilizing Tableau for enhanced data analysis. Tableau's visualization capabilities play a crucial role in the paper. Utilizing Tableau's functionalities, researchers can develop interactive and dynamic visualizations to facilitate the exploration and interpretation of Netflix data. Researchers can utilize filters, parameters, and drill-down functionality to analyze movies/shows based on specific criteria such as genre, IMDb score, or release year.

Additionally, Tableau Integration with external tools like MySQL allows for advanced data enrichment techniques. Overall, the paper aims to leverage Tableau's data analysis and visualization capabilities to gain valuable insights into Netflix data, exploring movie/show attributes, user preferences, and content trends. By addressing

existing problems and proposing new solutions, the paper contributes to enhancing the analysis of Netflix data using Tableau.

### A. Motivation

This paper is motivated by the goal of performing an in-depth analysis of Netflix's content ecosystem and leveraging sophisticated data visualization techniques for improved decision-making. By examining various dimensions of Netflix content, such as regional preferences, genre trends, and the attributes of successful movies and shows, we aim to derive actionable insights that can guide strategic decisions. Through the application of Tableau's advanced visualization capabilities, we will develop interactive and visually compelling representations of complex datasets, facilitating stakeholders' ability to identify trends, patterns, and relationships with enhanced clarity. This methodology is intended to enhance communication and support data-driven decision-making within Netflix. Additionally, the paper seeks to illustrate the effectiveness of Tableau in processing and analyzing large datasets, uncovering hidden patterns, and delivering actionable insights, thereby highlighting its utility as a critical tool in the entertainment industry.

The paper starts with a background study in Sect. 2. The flowchart analysis of the proposed method for Netflix using Tableau is given in Sect. 3. Section 4 explains the results of various analyses and investigations. Finally, we conclude our paper with Sect. 5.

## 2 Background Study

Tripati et al. [1] introduce several hypotheses regarding the integration of big data-driven solutions. It emphasizes the increasing focus on machine learning and human intelligence in these solutions, with over 30% of technological leaders seeking intelligent software for decision-making. The hypotheses posit positive associations between big data-driven solutions and information quality, user behavior-based recommendations, customer lifetime value (CLV), strategic decision-making, and business performance. The study underscores the significance of data visualization tools, geographical information, and continuous recommendations in shaping marketing strategies and decision-making processes. Overall, the research suggests that effective utilization of big data-driven solutions positively impacts various facets of organizational success.

The paper [2] underscores the pivotal role of Data Science in enhancing business models through analytics and deep learning. It outlines applications such as internet search, digital advertising, and recommender systems, emphasizing their positive impact on user experience and marketing effectiveness. Big Data's influence in financial services, communications, and retail is discussed, highlighting its predictive capabilities and contribution to decision-making. Big Data Analytics is presented as a critical process for cost reduction, better decision-making, and the creation of customer-centric goods and services. The subsequent sections delve into emerging

trends in Data Science and Big Data Analytics, including the rise of Augmented Analytics, and offer a practical use case in Diabetes Prevention. Finally, a taxonomy is provided, comparing terms like Data Science, Data Engineer, Data Analytics, Business Intelligence, and Business Analytics, elucidating their distinct roles in the analytical process.

The paper [3] points out that the integration of data analytics and consumer insights offers businesses a powerful means to enhance decision-making, deliver personalized customer experiences, and gain a competitive advantage. Successful case studies from industry leaders like Netflix, Amazon, Starbucks, and Target underscore the effectiveness of leveraging data-driven strategies in marketing campaigns. While challenges such as data quality, privacy concerns, and skill requirements exist, addressing them through strategic investments and ethical considerations can pave the way for sustainable success. The impact of targeted marketing campaigns and personalized experiences is evident in heightened customer satisfaction, increased engagement, conversion rates, and overall revenue growth. Ultimately, the ethical integration of data analytics and consumer insights remains a pivotal element in building lasting customer loyalty and achieving long-term business success.

The paper [4] describes the versatility and user-friendly interface of Tableau, a powerful data visualization tool, highlighting its key products and stages of data processing. Tableau enables easy handling through drag-and-drop functionality, facilitating the creation of reports and dashboards that can be shared across organizations securely. The tool's speed, scalability, and integration capabilities make it a preferred choice for data analysis. It then transitions to the application of Tableau in cleaning and analyzing data obtained from Kaggle and IMDb, emphasizing the integration of auto-suggestions and the development of a recommender system based on item-based collaborative filtering. The recommender system incorporates user ratings, movie IDs, and a survey form to enhance personalization, demonstrating Tableau's effectiveness in extracting valuable insights from diverse datasets.

The paper [5] employs OLAP for complex data analysis, connecting Tableau Desktop to the PostgreSQL database. Visualized through a smart dashboard, business intelligence insights reveal fluctuations in monthly sales, prompting suggestions for marketing innovations. Product category segmentation emphasizes the success of bed bath table products, while payment type segmentation highlights credit cards as the preferred method. Annual profit segmentation indicates a steady increase, providing a reference for future sales strategies. Rating segmentation underscores the need for improving low customer satisfaction scores, and customer city segmentation suggests expanding marketing efforts to attract interest beyond Sao Paulo and Rio De Janeiro.

The paper [6] aims to analyze Netflix data, particularly on movies and shows, with a focus on attributes such as release year, genre, and ratings from TMDB and IMDb databases. Special attention is given to gauging the popularity of content among Netflix viewers. By leveraging datasets for visualization, the authors present a comprehensive overview of the most popular shows and movies, offering valuable insights for the platform. The findings indicate that comedy emerges as the most favored genre among audiences. The research suggests that such analyses can inform

content recommendations to enhance viewership on the platform, showcasing the significance of dataset preprocessing [7] and visualization tools in this process.

The paper [8] delves into the usage of social media platforms, particularly during quarantine/lockdown, aiming to answer questions about the hours spent on social media, the popularity of specific apps, and age group preferences. Utilizing public source data from Kaggle, the analysis focuses on around 600 responses from individuals spanning different age groups. The data encompasses variables such as sleep patterns, exercise routines, skills acquired during lockdown, and time spent on various media. After thorough data cleaning to ensure accuracy, the study proceeds with data visualization, creating graphs and interactive dashboards using Tableau. The Chi-square independence test in R is employed to assess associations between categorical variables, providing insights into the relationships within the dataset.

The paper [9] underscores Tableau's widespread popularity with over 57,000 accounts across various industries, emphasizing its reputation as a potent data visualization tool. While celebrated for its simplicity and ability to create interactive visualizations, concerns are raised regarding manual effort required for report refreshing and limitations in importing existing visualizations. Scholarly discussions delve into Tableau's core components, such as Tableau Desktop, Server, and Public, elucidating their functionalities and applications, especially in the context of Big Data projects and AI applications. It also highlights challenges users face, including issues with conditional formatting, table presentations, and static parameters, along with the perceived higher cost compared to alternative BI tools. Overall it provides a comprehensive view of Tableau's strengths, challenges, and its pivotal role in shaping visual analytical processes.

The paper [10] finds out that the adoption of Tableau and SQL in a company has empowered employees to gain clearer insights into their data, facilitating more informed decision-making. This combination allows for effective program execution by anticipating and responding to various circumstances. Decision structures, evaluating multiple expressions as true or false, guide the selection of actions. Business Data Analytics (BDA) involving data gathering, analysis, and interpretation is recognized for producing business value and actionable insights across functional divisions. Tableau's role in creating visual narratives is highlighted, providing context and showcasing the impact of decisions on outcomes. Additionally, SQL's efficient statement execution is crucial for preventing performance issues, contributing to a more streamlined decision-supporting platform.

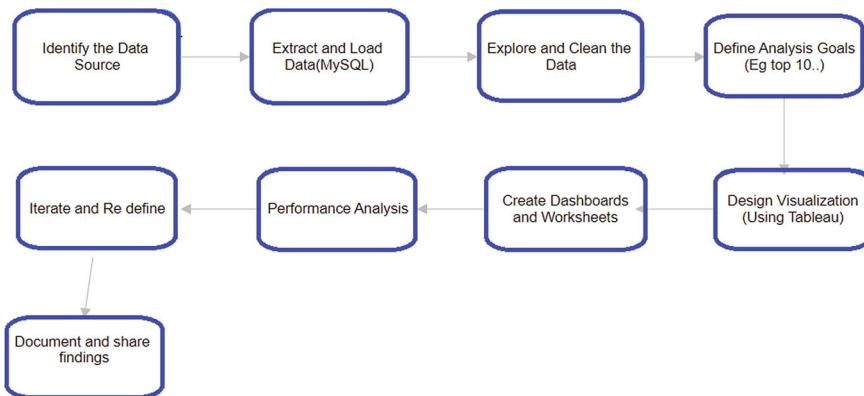
The paper [11] underlines the pivotal role of data visualization tools in effective data analysis and communication. The comprehensive overview delves into various visualization tools, elucidating their features, strengths, and weaknesses. Different types of visualizations, including bar charts, line graphs, and heatmaps, are explored with insights into their effective use in diverse situations. Popular tools such as Tableau, Power BI, and Python libraries like Matplotlib are evaluated, providing real-world examples of their applications. Best practices for creating impactful visu-

alizations, such as choosing suitable color schemes and designing for accessibility, are discussed. It concludes by delving into future trends, including augmented reality and machine learning integration [12], which promise to further enhance the capabilities of data visualization tools, underscoring their importance in data analysis and decision-making.

### 3 Proposed Methodology

In this section, we explain about the proposed flowchart for the Netflix analysis as shown in Fig. 1.

1. ***Interactive Visualizations:*** Tableau offers a user-friendly interface with drag-and-drop functionality, enabling users to create interactive visualizations easily. This helps in exploring Netflix data visually and gaining insights quickly.
2. ***Real-Time Data Streaming:*** Tableau supports real-time data streaming, enabling users to analyze streaming data from Netflix as it arrives. This feature allows for immediate insights and faster decision-making.
3. ***Data Blending and Data Integration:*** Tableau allows users to blend and integrate data from multiple sources, facilitating a comprehensive analysis of Netflix data. It enables combining Netflix data with external data sources to gain deeper insights.
4. ***Storytelling and Dashboard Creation:*** Tableau offers storytelling capabilities, allowing users to create interactive dashboards and present data-driven narratives. This feature helps communicate insights effectively and engage stakeholders.
5. ***Cloud and Mobile Accessibility:*** Tableau provides cloud-based and mobile solutions, allowing users to access and analyze Netflix data from anywhere, anytime. This enhances collaboration and enables remote data analysis.



**Fig. 1** Flowchart of Netflix analysis problem

## A. Model Description

Netflix needs machine learning analysis to better understand audience preferences, optimize content strategies, enhance personalization, identify new market opportunities, improve operational efficiency, and leverage big data. By using machine learning, Netflix can handle vast amounts of data to uncover hidden patterns and insights, giving the company a competitive advantage in the streaming industry and enabling it to deliver a more engaging and satisfying viewer experience. We have taken Netflix dataset preprocessed and performed seven distinct machine learning algorithms and the descriptions are mentioned below.

### a. *Logistic Regression*

Logistic regression stands as one of the most widely utilized algorithms in supervised learning, employing a predetermined set of independent factors to predict categorical dependent variables.

### b. *Decision Tree*

Decision trees, internal nodes represent feature tests while leaves denote class labels, with the connections between nodes illustrating the relationships between features and class labels [13].

### c. *Random Forest*

Random forest, a supervised learning method, is versatile, catering to both classification and regression tasks. It incorporates multiple decision trees, providing enhanced reliability and accuracy. Despite its ability to handle missing data and combat overfitting, its utilization of several decision trees may lead to slower predictions compared to a single decision tree.

### d. *XGBoost*

XGBoost Classifier merges gradient boosting with decision trees, yielding high-performance predictions. Renowned for its efficient parallel processing, adept management of missing data, and provision of feature-importance insights, it's favored for classification tasks due to its adaptability, regularization, and robustness.

### e. *KNN*

KNN (K-nearest neighbors), a supervised learning technique, predicts output data points based on labeled input data. It's celebrated for its simplicity and flexibility, applicable across various problem domains. By assessing the similarity of data points to nearby neighbors, KNN excels in handling real-world data, devoid of assumptions about the underlying dataset [14] machine learning.

### f. *Naive Bayes*

Naive Bayes, rooted in Bayes' Theorem, is a probabilistic model ideal for vast datasets, leveraging conditional probabilities to predict the likelihood of hypotheses given observed data. Based on the Naive Bayes assumption, it presumes independence among features, facilitating straightforward model construction [15].

### g. SVM

SVM (support vector machine) navigates an N-dimensional space to identify a hyperplane effectively segregating data points into distinct groups. The dimensions and characteristics of this hyperplane align with the dataset's features, assuming a linear shape in scenarios.

### h. Evaluation Methods

$$\text{Accuracy} = \frac{\text{Number of correct Predictions}}{\text{Total Number of Predictions}} \quad (1)$$

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \quad (2)$$

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \quad (3)$$

## 4 Result

During the work, several analyses and investigations were conducted while working on the solution. These analyses focused on various aspects of Netflix data and aimed to extract meaningful insights and findings. Some of the key analyses and investigations are.

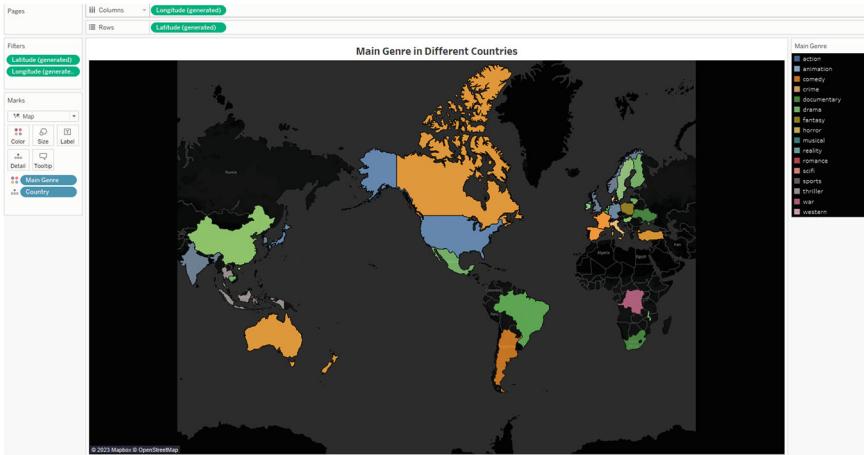
### 1. Country Analysis

The dataset was analyzed based on country preferences. The number of movies and shows available in different countries was investigated. This analysis aimed to identify top-performing countries and their preferred genres. Fig. 2 displays the dominant genre in each nation. For instance, action is a popular genre in the USA. Likewise, main genre in Brazil is drama. War genre is only popular in Democratic Republic of the Congo. Likewise, fantasy genre in Poland.

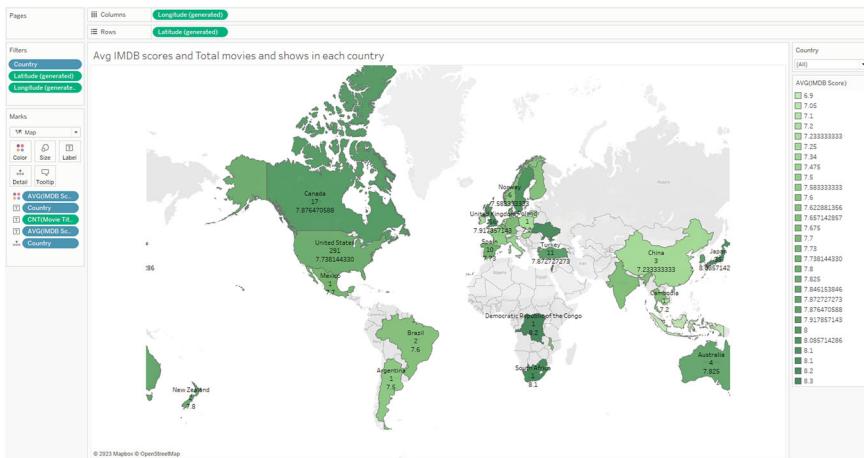
### 2. Movie and Show Analysis

In-depth analysis of movies and shows on Netflix was conducted. Factors such as release year, IMDb score, number of upvotes, duration, number of seasons, and content type were investigated to understand their impact on the success and popularity of movies and shows. This analysis aimed to identify highly rated movies and shows, trends in movie durations, and preferred show formats. The average IMDb ratings for films and TV shows in each nation are displayed in Fig. 3. Japan has the highest average IMDb rating. Having the lowest overall IMDb rating is Indonesia.

The top 10 films and TV shows with the most votes are displayed in this bubble chart. Breaking Bad is in the lead. Death Note is the second. Avatar: The Last Airbender comes in third (Fig. 4).



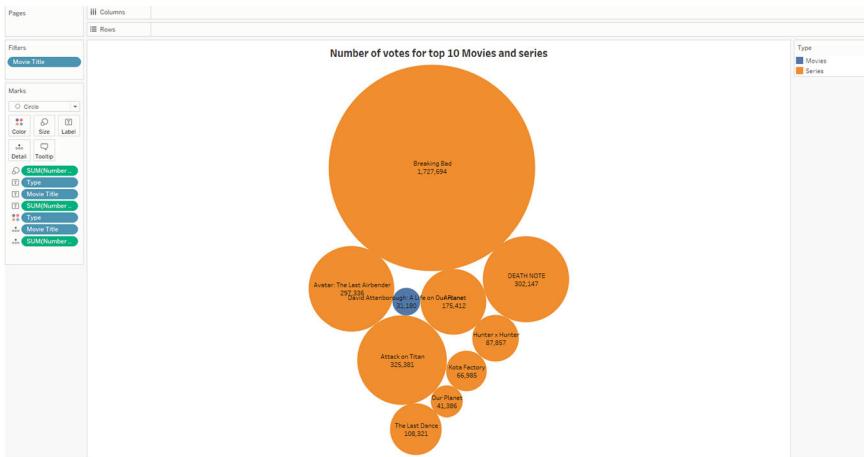
**Fig. 2** Dominate genre in different countries



**Fig. 3** Average IMDB ratings for movies and TV shows

### 3. Visualization and Dashboard Creation

Tableau was used to create interactive visualizations and dashboards to represent the analysis findings. Visualizations included bar charts, heatmaps, and other types of graphs to present data patterns and trends effectively. Dashboards were designed to provide an intuitive and interactive platform for stakeholders to explore the data and derive insights. Figure 5 shows the heatmap of different films from India, Japan, and the USA along with their genre and IMDb rating. The tree map displays the typical number of TV seasons for each genre in Fig. 6.



**Fig. 4** Voting-based bubble chart for movies and TV shows

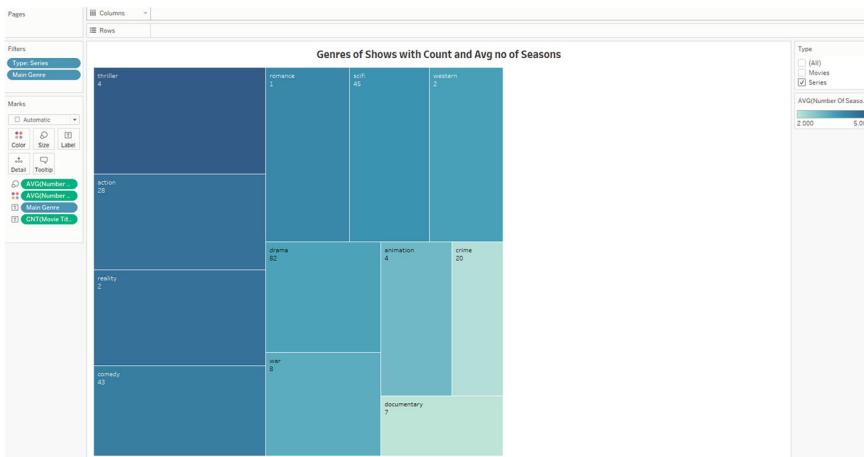


**Fig. 5** Heatmap for different movies in different countries

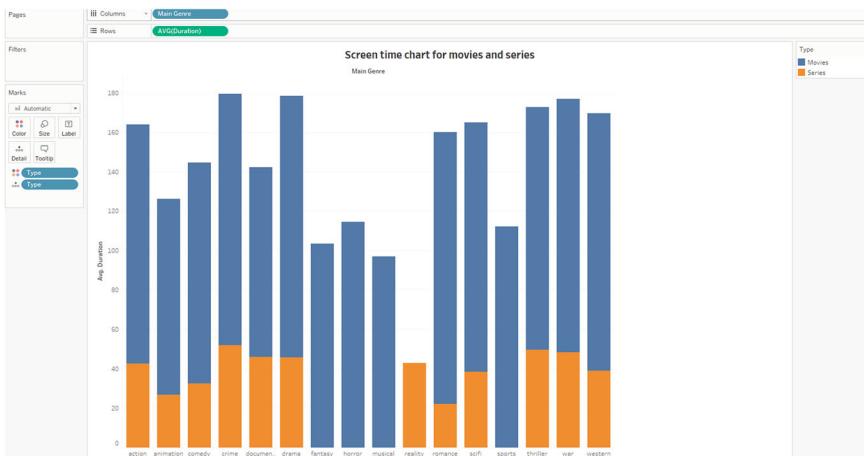
#### 4. Comparative Analysis

Comparative analysis was conducted to compare different variables and identify relationships between them. For example, the relationship between IMDb scores and the number of upvotes or the correlation between content duration and user ratings. These comparative analyses helped identify key factors influencing content popularity and user preferences. The average screen time for each genre is displayed in Fig. 7. The percentage of genre by country is shown in Fig. 8.

Lastly, the top 10 films and TV shows with the most IMDb score are displayed in the Fig. 9.



**Fig. 6** Tree map for genre of shows with count and average number of seasons

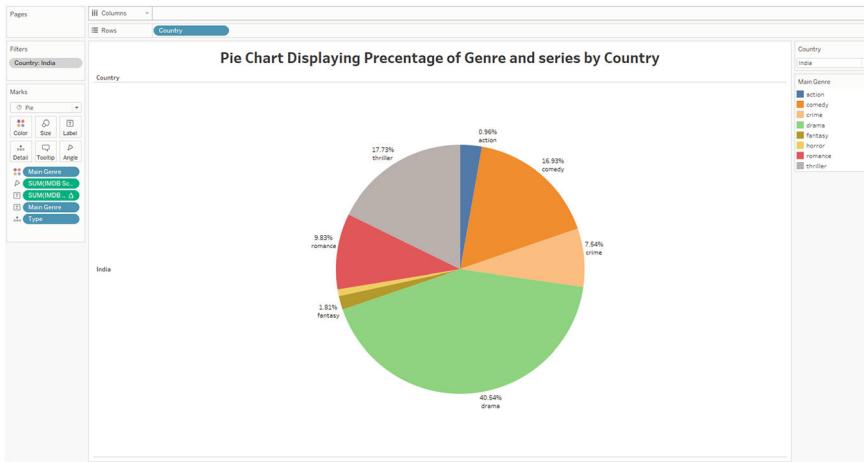


**Fig. 7** Average screen time for each genre of movies and series

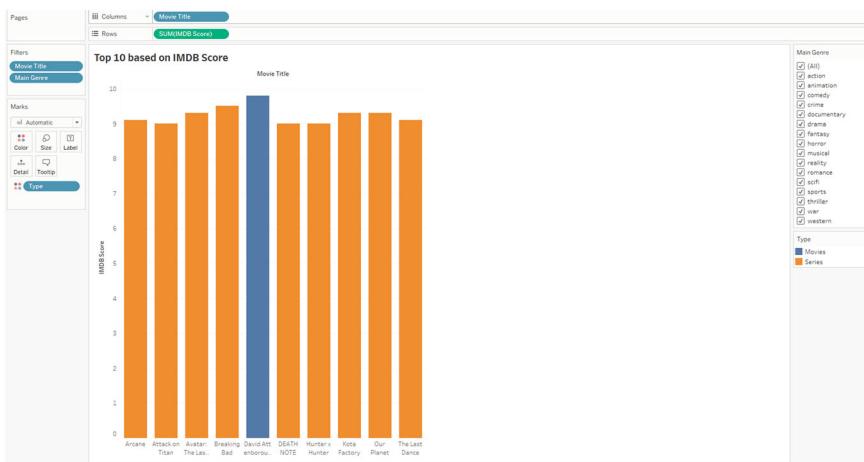
Throughout the paper, statistical techniques, data mining, and visualization methods were employed to explore, analyze, and interpret the Netflix data. The investigations aimed to uncover insights that could inform decision-making processes, identify content opportunities, and contribute to a better understanding of Netflix's audience and content landscape.

#### A. ML Visualization Results

This section explains the various evaluation metrics for 70–30 and 80–20 (Tables 1 and 2).



**Fig. 8** Percentage of genre and series by country



**Fig. 9** Top 10 films and TV shows based on IMDb score

## 1. For 70–30 train-test split

- (a) Accuracy Rate and Precision Rate

See Figs. 10 and 11.

- (b) Recall Rate and F1 Score Rates

See Figs. 12 and 13.

## 2. For 80–20 train-test split

- a. Accuracy Rate and Precision Rate

**Table 1** 70–30 train-test split testing results

Serial Number	Model name	70–30 train-test split testing results			
		Accuracy	Precision	Recall	F1
1	Decision tree	100.00	100.00	100.00	100.00
2	Random forest	100.00	100.00	100.00	100.00
3	Logistic regression	100.00	100.00	100.00	100.00
4	Support vector machine	98.85	98.66	98.86	98.76
5	K-nearest neighbor	97.70	97.45	97.58	97.5
6	Naïve Bayes	100.00	100.00	100.00	100.00
7	XGBoost	100.00	100.00	100.00	100.00

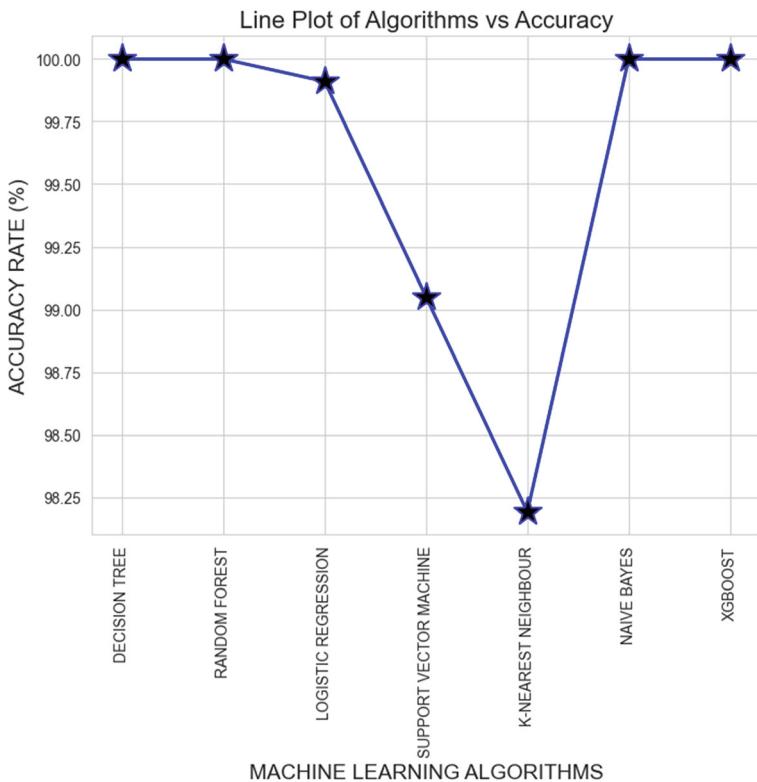
**Table 2** 80–20 train-test split testing results

Serial Number	Model name	80–20 train-test split testing results			
		Accuracy	Precision	Recall	F1
1	Decision tree	100.00	100.00	100.00	100.00
2	Random forest	100.00	100.00	100.00	100.00
3	Logistic regression	100.00	100.00	100.00	100.00
4	Support vector machine	99.91	99.93	99.88	99.91
5	K-nearest neighbor	99.05	98.79	99.16	98.97
6	Naïve Bayes	98.19	97.76	98.33	98.03
7	XGBoost	100.00	100.00	100.00	100.00

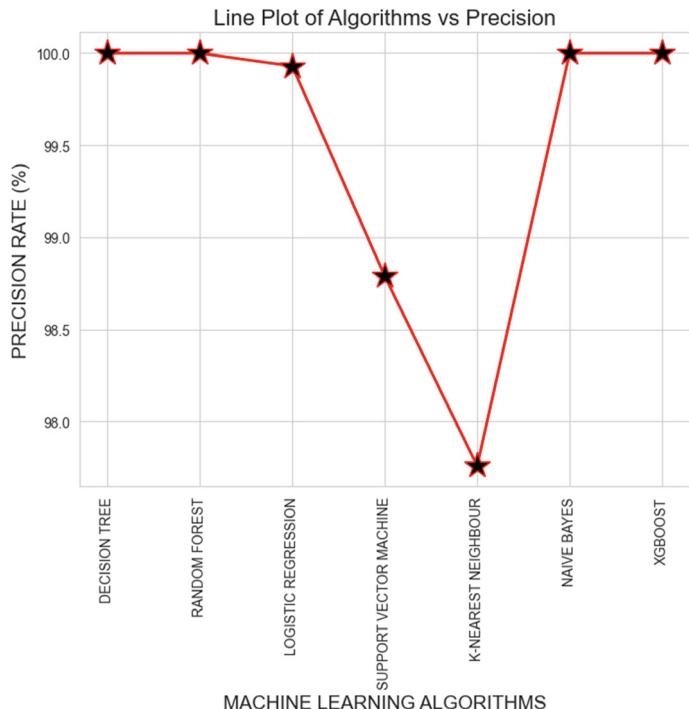
See Figs. 14 and 15.

b. Recall Rate and F1 Score Rates

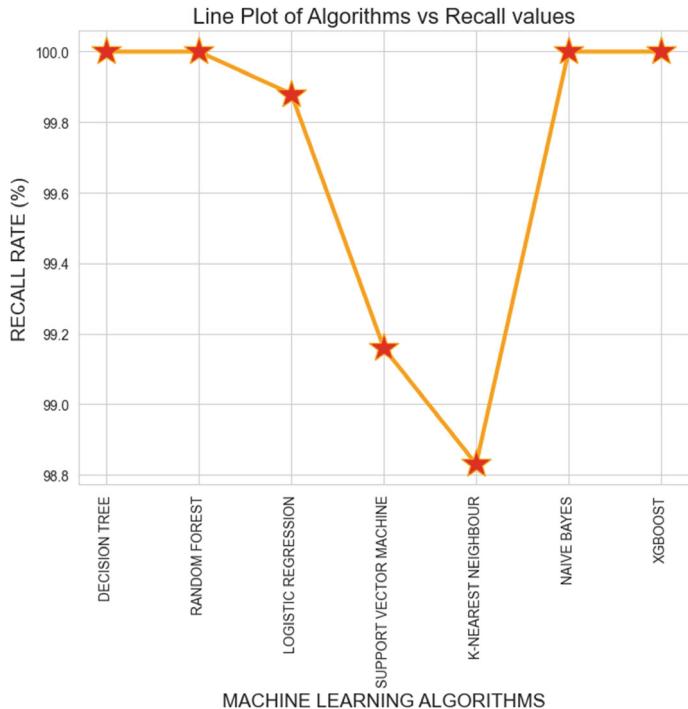
See Figs. 16 and 17.



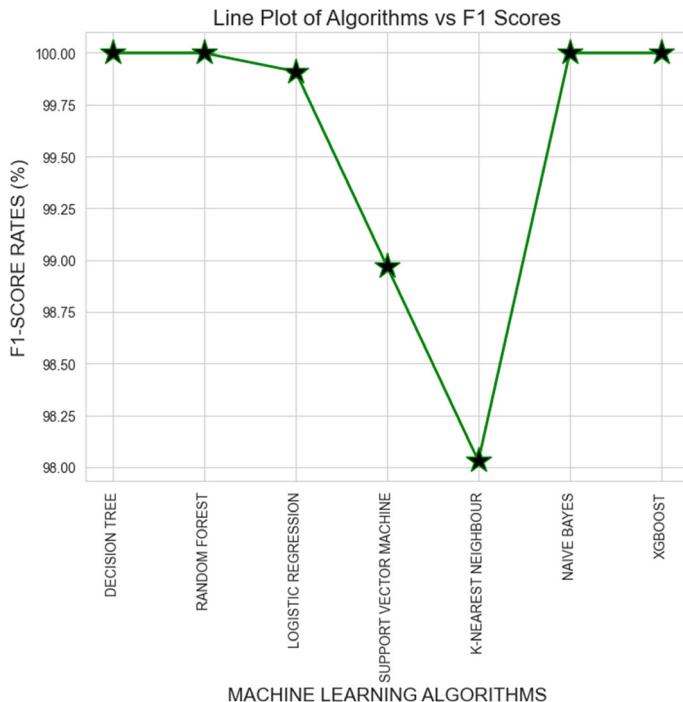
**Fig. 10** Accuracy rate of various ML algorithms



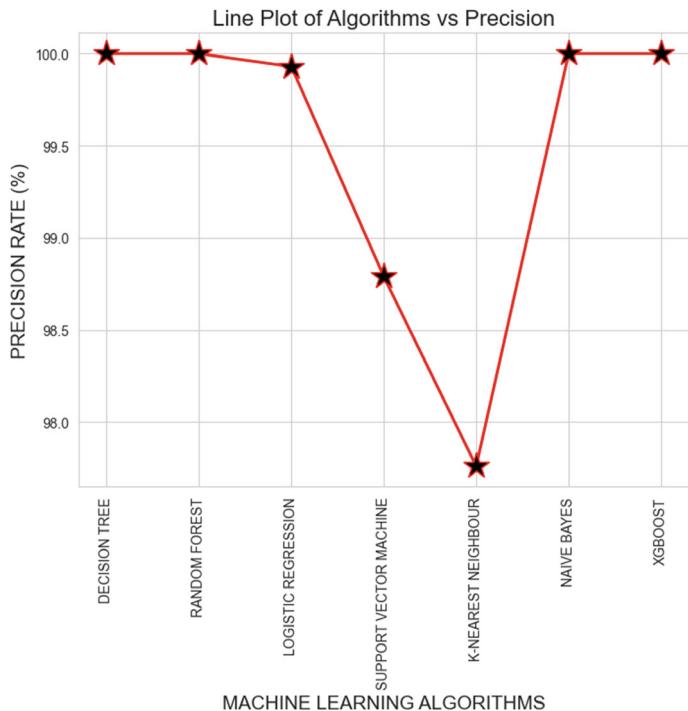
**Fig. 11** Precision rate of various ML algorithms



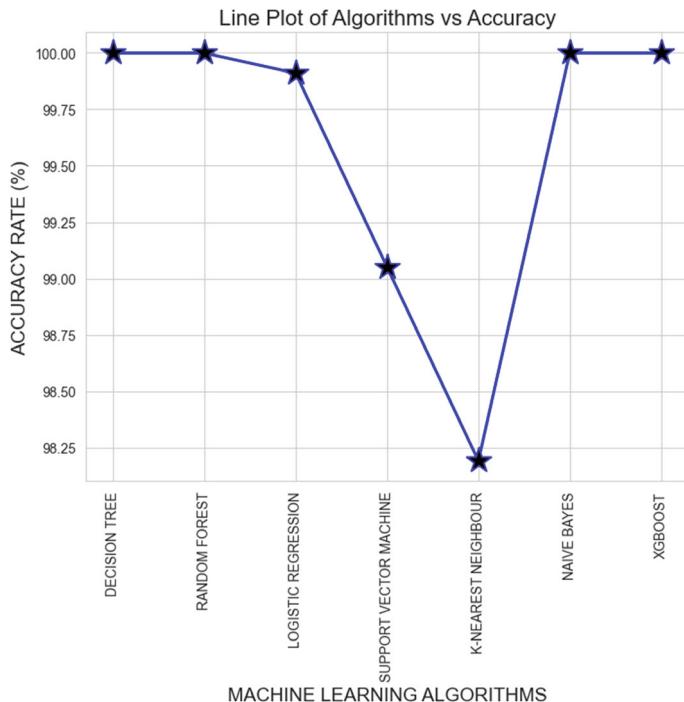
**Fig. 12** Comparison of recall rate of various ML algorithms



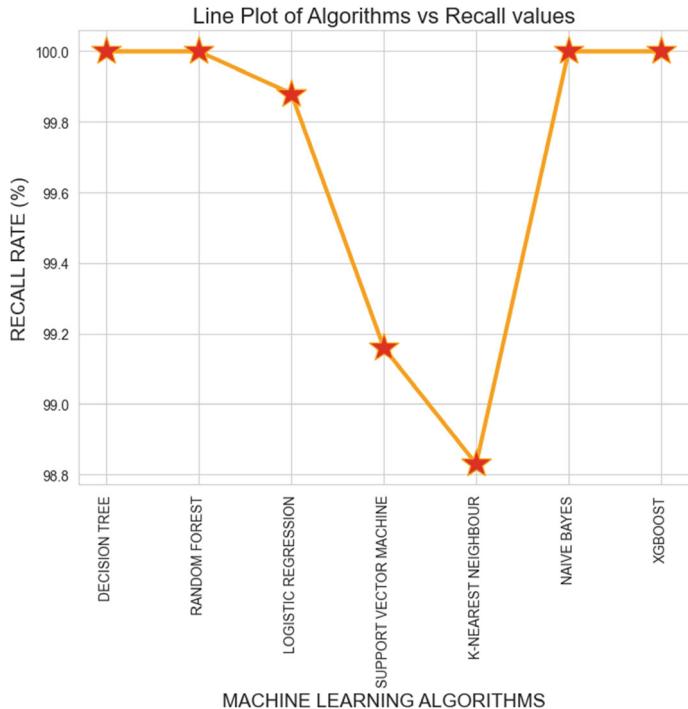
**Fig. 13** Comparison of F1 score rates



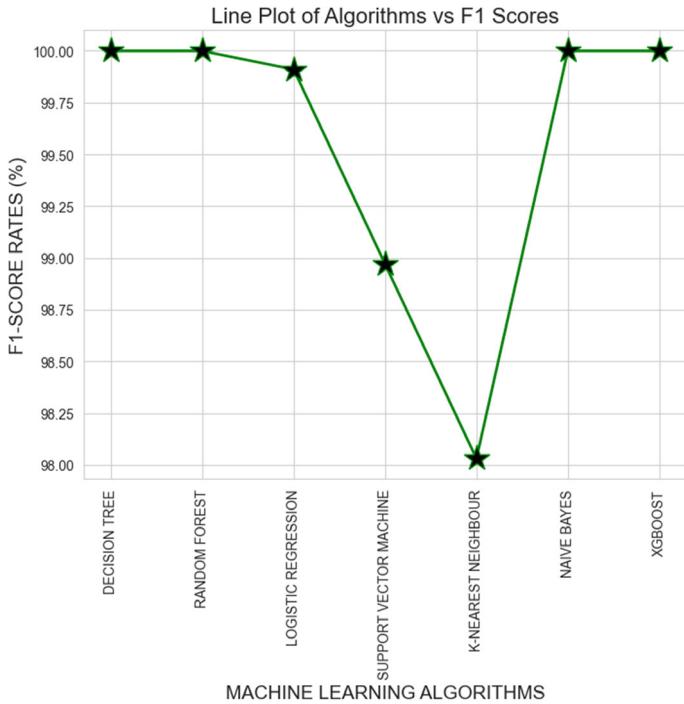
**Fig. 14** Accuracy rate of various ML algorithms



**Fig. 15** Precision rate of various ML algorithms



**Fig. 16** Comparison of recall rate of various ML algorithms



**Fig. 17** Comparison of F1 score rate

## 5 Conclusion

The paper aimed to perform a comprehensive data analysis of Netflix using Tableau, focusing on various aspects such as country preferences, genre trends, movie and show characteristics, and content type. By leveraging Tableau's visualization capabilities, interactive dashboards were created to present the analyzed data effectively, allowing stakeholders to explore and interpret the findings. The insights gained from the analysis can inform content strategy decisions, content acquisition, and production investments. They can also enhance user experience by optimizing content recommendations, improving the user interface, and tailoring marketing campaigns to align with user preferences. By using machine learning, Netflix can handle vast amounts of data to uncover hidden patterns and insights, giving the company a competitive advantage in the streaming industry and enabling it to deliver a more engaging and satisfying viewer experience. The performance of the decision tree, random forest, logistic regression, and XGBoost classifier remained consistent across both the train-test splits. Overall, the paper has provided valuable insights into user preferences, content popularity, and trends within the Netflix platform. The findings and recommendations can drive strategic planning, content selection, user experience

enhancements, and overall business performance on Netflix. By leveraging data analysis and visualization techniques, this paper has contributed to an improved understanding of the Netflix ecosystem and its impact on user satisfaction and business success.

## References

1. Tripathi A, Bagga T, Sharma S, Kumar Vishnoi S (2021) Big data-driven marketing enabled business performance: a conceptual framework of information, strategy and customer lifetime value. In: 2021 11th international conference on cloud computing, data science and engineering (confluence). IEEE, Noida, India, pp 315–320. <https://doi.org/10.1109/Confluence51648.2021.9377156>. [https://doi.org/10.1007/978-981-15-6648-6\\_10](https://doi.org/10.1007/978-981-15-6648-6_10)
2. Goyal D, Goyal R, Rekha G, Malik S, Tyagi AK (2020) Emerging trends and challenges in data science and big data analytics. In: 2020 International conference on emerging trends in information technology and engineering (ic-ETITE). IEEE, Vellore, India, pp 1–8. <https://doi.org/10.1109/ic-ETITE47903.2020.9316>
3. Wu H (2023) Leveraging data analytics and consumer insights for targeted marketing campaigns and personalized customer experiences. *J World Econ* 2(3):3
4. Shah H, Jadhav V, Gharte T, Wattamwar S, Naik V (2022) Evaluation of different OTT platforms with data analytics techniques for recommending personalized content to the users, vol 9, no 2
5. Angrainy TD, Sari AR. Implementation of extract, transform, load on data warehouse and business intelligence using pentaho and tableau to analyse sales performance of Olist Store, vol 7, no 2
6. Devashree, Goel H, Sharma N, Mangla M (2022) Analysis and visualization of Netflix shows. In: 2022 4th international conference on artificial intelligence and speech technology (AIST). IEEE, Delhi, India, pp 1–6. <https://doi.org/10.1109/AIST55798.2022.10065331>
7. Elakiya E, Rajkumar N (2017) Designing preprocessing framework (ERT) for text mining application. In: 2017 International conference on IoT and application. IEEE, pp 1–8. <https://doi.org/10.1109/ICIOTA.2017.8073613>
8. Goncalves Almeida A (2023, May) Use of social media during the Covid-19 pandemic of 2020. In: Honors program theses project. [https://vc.bridgew.edu/honors\\_proj/611](https://vc.bridgew.edu/honors_proj/611)
9. Kumar A, Ali AS, Jamnadas H, Sharma V (2019) Big data visualisation—an update until today. In: 2019 IEEE Asia-Pacific conference on computer science and data engineering (CSDE), pp 1–8. <https://doi.org/10.1109/CSDE48274.2019.9162356>
10. Janani S, Kumar MS, Bilagi AS, Chatterjee PS (2023) Effectiveness of tableau and SQL software in analytics in business decisions. *Int J Eng Manag Res* 13(1):1. <https://doi.org/10.31033/ijemr.13.1.15>
11. Addepalli L, Sindhuja S, Gaurav L, Ali W (2023) A comprehensive review of data visualization tools: features, strengths, and weaknesses, vol 10, pp 10–20. <https://doi.org/10.22362/ijcert/2023/v10/i01/v10i0102>
12. Elakiya E et al (2023) Text feedback classification using machine learning techniques. In: 2023 2nd international conference on edge computing and applications (ICECAA). IEEE <https://doi.org/10.1109/ICECAA58104.2023.10212398>
13. Yadav P (2018, November 14) Decision tree in machine learning. <https://towardsdatascience.com/decision-tree-in-machine-learning-e380942a4c96>
14. Lateef Z (2020, May 14) KNN algorithm: a practical implementation of KNN algorithm in R. <https://www.edureka.co/blog/knn-algorithm-in-r/>
15. Gandhi R (2018, May 5) Naïve Bayes classification. <https://towardsdatascience.com/naive-bayes-classifier-81d512f50a7c>

# Acoustic Monitoring of Biodiversity



Aniket Kumar, Swati Kale, Amey Jojare, and Siddesh Sabade

**Abstract** Sound is considered as one of the distinguishing features for the classification of animals. Acoustic monitoring is one of the effective tools for ecological research and biodiversity assessment. Deep Learning has emerged as one of the effective techniques to deal with huge and complex datasets. It has a great ability to understand patterns in the data. It is used in various fields like image processing, robotics, etc. including bioacoustics. Despite advancement in the Deep Learning on bioacoustics classification, attaining high accuracy remains a significant challenge. In this paper, a combination of data augmentation and feature extraction techniques is proposed. The performance of DL models is analyzed in the context of data augmentation and feature extraction. Experimental results demonstrate significant improvements in classification accuracy compared to previous methods. The Deep Learning model achieves 93.71% accuracy on the ESC-50 datasets, representing significant improvements over previous methodologies.

**Keywords** Environmental sound classification · MEL spectrogram · MFCC · Gabor · Convolutional neural network (CNN) · Data augmentation · Hyperparameter optimization · Model evaluation

---

A. Kumar (✉) · A. Jojare · S. Sabade  
JSPM's Rajarshi Shah College of Engineering, Pune, India  
e-mail: [aniketkr08903@gmail.com](mailto:aniketkr08903@gmail.com)

A. Jojare  
e-mail: [ameyjojare@gmail.com](mailto:ameyjojare@gmail.com)

S. Sabade  
e-mail: [siddeshsabade@gmail.com](mailto:siddeshsabade@gmail.com)

S. Kale  
Savitribai Phule Pune University, Pune, India  
e-mail: [swadip.06@gmail.com](mailto:swadip.06@gmail.com)

## 1 Introduction

In years, the use of advanced machine learning techniques and deep learning has greatly improved the field of bioacoustics resulting in progress in identifying and categorizing animal vocalizations. Multiple studies have investigated various techniques within forest monitoring systems aimed at safeguarding forest reserves. Previous research has explored diverse sound classification methods for identifying various species and potential forest hazards like illegal logging, poaching, and wildfires [1–3]. Bioacoustics research has been considerably advanced by recent machine learning developments, which have improved the capacity to identify and classify animal vocalizations. Numerous methods for audio signal processing and machine learning have been used to improve the accuracy of the model. These include the use of traditional machine learning algorithms, including Gaussian mixture model (GMM) [4], K-nearest neighbors (KNN) [3], support vector machine (SVM) [5]. Audio processing techniques including time stretch, resampling, and resizing have shown significant increase in the model accuracy [6]. Various feature extraction techniques have been employed including mel spectrogram, mel frequency cepstral coefficient (MFCC), and gammatone spectrogram [7]. Across different geographical regions, biodiversity varies significantly. For instance, distinct species inhabit specific locations, each adapting to its unique environment. Consequently, the acoustic landscapes also vary, influenced by factors such as terrain, vegetation, and climate [8]. Deep Learning (DL) models perform much better than ML models in terms of feature extraction. Deep learning uses layers to transform data in complex ways, allowing neural networks to understand data at different levels. This makes it great for solving tough problems like recognizing speech, finding objects, and understanding images. In most of the audio classification approach, audio data are converted into image and then image is classified using convolutional neural networks [7]. In this paper, performance and accuracy of sound classification using traditional machine learning and deep learning techniques are investigated. The model accuracy is further increased by using various methods including hyperparameter tuning, feature extraction, layer architecture, and data augmentation. The rest of this paper is structured in the following way. Recent related works are introduced in Sect. 2. A comprehensive explanation of the suggested methodology is given in Sect. 3. Section 4 compares and analyzes the results obtained. The paper is finally concluded in Sect. 5.

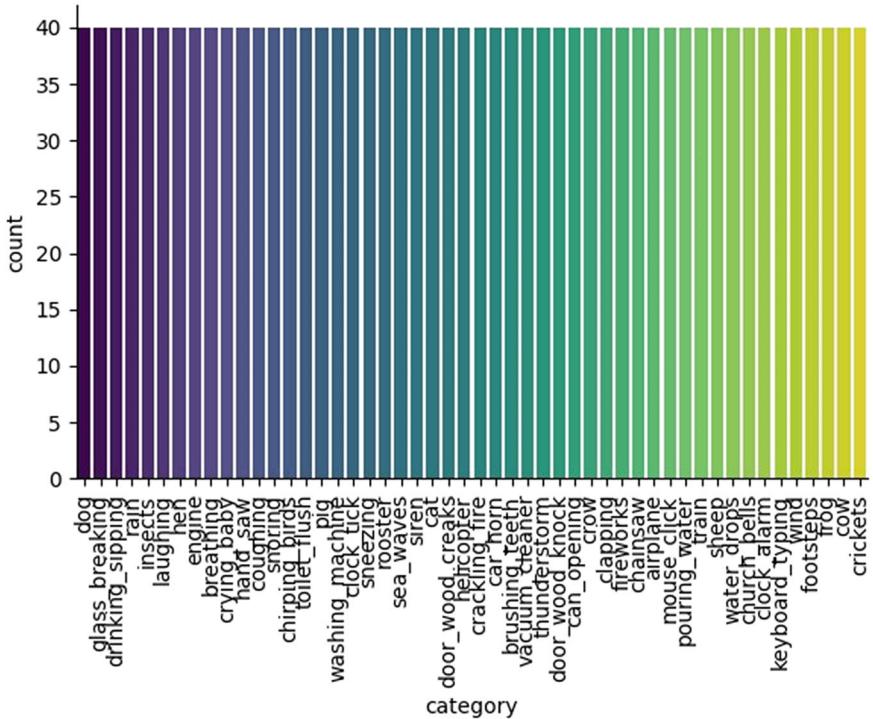
## 2 Related Work

In this section, emphasis is given to bioacoustic or environment sound classification (ESC) studies that utilize machine learning algorithms and deep learning techniques for the classification and detection of animal sound. Iosif Mporas et al. [3] detected illegal logging activity in forest using audio recordings. Various Machine learning algorithms used, including, KNN, SVM, and 3 layer multilayer perceptron (MLP).

On the basis of result, SVM outperforms other classifiers. Late fusion of postprocessed recognition output of three classifiers (SVM, MLP, and decision tree) resulted in achieving an impressive 94.42% accuracy. Boddapati et al. [7] discovered that powerful neural networks like GoogLeNet and AlexNet can classify both images and sounds. An audio is first converted into a spectral image and then this spectral image is given as input to image classification neural networks (GoogLeNet and AlexNet). GoogLeNet performed the best, achieving 73% accuracy on the ESC-50 dataset and 91% accuracy on the ESC-10 dataset. Lorène Jeantet et al. [8] demonstrated the efficacy of integrating contextual information into classifiers to enhance performance. Their approach involved combining the outcomes of a baseline CNN model with geographical prior information. This fusion yielded a remarkable F1 score of 87.78%, a substantial improvement compared to the 61.02% achieved by the baseline model alone. Pickzak et al. [9] demonstrated that employing a convolutional neural network (CNN) performs better than manual feature extraction technique. They used a 2D CNN to understand the patterns in log mel spectrograms, which are pictorial representations of sound. This approach gave much better results than older methods like KNN, SVM, and Random Forest, especially when tested on the ESC-50 dataset.

Zhang et al. [10] investigated that frame level attention mechanism for CNN and RNN layers gives much better performance. It was found that sound clips contain various irrelevant frames which reduces model accuracy. Attention mechanism applied to the model to focus on the semantically relevant parts which resulted in increase in accuracy 93.7% for ESC-10 and 86.1% for ESC-50 dataset which is better than previous works. Boqing Zhu et al. [11] developed WaveMsNet, a new type of network. By using multi-scale convolution and a two-phase approach, they achieved remarkable accuracy rates of 93.75% and 79.10% on the ESC-10 and ESC-50 datasets, respectively. Their method combines waveform and spectrogram features in one model, leading to better results.

The reviewed literature highlights the evolution and effectiveness of various machine learning and deep learning approaches in the domain of audio classification. From traditional classifiers like SVM and KNN to advanced CNN architectures and novel networks like WaveMsNet, each study demonstrates significant strides in improving accuracy and robustness. Our understanding of these advancements has motivated us to propose a CNN-based approach that leverages 2D convolution, multi-feature extraction, and extensive data augmentation techniques. This comprehensive methodology ensures that our model can effectively learn and generalize from complex and varied audio data, ultimately leading to improved classification accuracy.



**Fig. 1** Distribution of classes in the ESC-50 dataset

### 3 Methodology

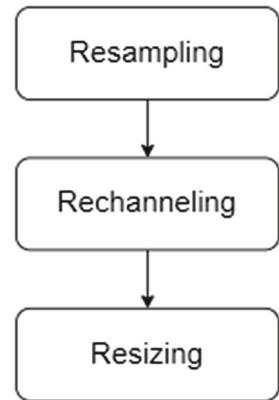
#### 3.1 Dataset Collection

The ESC-50 dataset [12] is a publicly available resource for environmental sound classification, encompassing 50 equally balanced classes. The distribution of these classes is depicted in Fig. 1. This dataset includes 2,000 audio recordings that have been divided into five categories: exterior/urban noises, human non-speech, natural soundscapes and water sounds, animals, and interior/domestic sounds.

#### 3.2 Dataset Preprocessing

Data play a pivotal role in determining the performance of a model. Data preprocessing, a crucial technique in data science, involves the identification, correction, or removal of inaccurate data from a dataset. This process encompasses various steps, including data cleaning, data reduction, and data transformation. When working with

**Fig. 2** Data preprocessing pipeline



audio signals, raw data cannot be directly fed into machine learning or deep learning models. Therefore, it's essential to perform data transformation to convert it into a more suitable format. The data preprocessing pipeline employed in this research paper, as depicted in Fig. 2, follows several key steps.

Firstly, the sampling rate of the audio data is adjusted. Sampling rate refers to the number of times the audio signal is sampled per second. In this study, all audio signals are resampled to 20 kHz to ensure uniformity. Next, the audio channels are standardized. Each channel represents the amplitude of the audio produced by the source at a specific moment in time. To simplify processing, all audio signals are converted to mono format. If an audio clip contains multiple channels, the average of both channels is taken. Audio resizing is another essential step in the preprocessing pipeline. This involves adjusting the duration of all audio clips to a fixed length. Each audio signal is resized to 4000 ms using a combination of padding and trimming techniques. Padding involves adding silence (represented by 0) equally at the beginning and end of the audio clip. For trimming, the starting point is selected randomly to avoid bias.

### 3.3 *Data Augmentation*

Data augmentation is a vital pre-processing technique in machine learning and deep learning, particularly for tasks like image and audio classification. By introducing variations in the dataset, models are exposed to a broader spectrum of scenarios and contexts, thereby enabling them to generalize better and exhibit robust performance across unseen data. Additionally, augmenting the dataset mitigates the risk of overfitting, where models show high performance over training dataset but perform poorly over test dataset. In order to increase the number of data, this paper uses a combination of augmentation techniques such as time scaling, time shift, and pitch

**Table 1** Data augmentation

Augmentation techniques	Value range	Description
Time stretch (TS)	[0.5, 1.5]	Time is speed up and down
Time shift (TShift)	[0, 0.8]	Time is shifted randomly
Pitch shift (PShift)	[-1.5, 1.5]	Pitch is shifted by considering n_step in between -1.5 and 1.5
Mixed	TS: [0.5, 1.5] TShift: [0, 0.8] PShift: [-1.5, 1.5]	Combination of TS, TShift, and PShift

shift. This caused an increase in the ESC-50 dataset size and improved model accuracy. Through augmentation, the original ESC-50 dataset expanded to 8000 audio samples. 80% of the augmented samples were allocated for training the model, with the remaining 20% reserved for validating the model's performance during hyper-parameter optimization. Table 1 depicts various data augmentation techniques used in this paper along with the range of the value chosen randomly for generating an augmented dataset.

### 3.4 Feature Extraction

Feature extraction is a process of transforming raw data such as image, audio, etc. into numerical form that are compatible with machine learning algorithms. Spectrogram is the visual representation of audio signal. It shows signal strength at various frequencies over time. Mathematically, Spectrograms represent the squared magnitudes of the short-time Fourier transform (STFT).

$$\text{STFT}\{x(t)\}(\tau, w) = \int_{-\infty}^{\infty} x(t)w(t - \tau)e^{-iwt} dt \quad (1)$$

where,  $x(t)$  is the signal that is to be transformed,  $w(\tau)$  is the window function,  $\tau$  represents time axis,  $w$  represents frequency axis, and  $X(\tau, w)$  is the Fourier transform of the signal  $x(t)w(\tau)$ .

$$S\{x(t)\}(\tau, w) \equiv |X(\tau, w)|^2 \quad (2)$$

where,  $S$  represents spectrogram of the signal  $x(t)$  along the time axis,  $\tau$  and frequency axis,  $w$ .

A 3D representation of features constructed from Mel Spectrogram, MFCCs, and Chromagram. Gabor Filter is a digital filter that helps in texture analysis, edge detection, and feature extraction. It is a bandpass filter. Mathematically, it is a function of various parameters including kernel size, wavelength, standard deviation, angle,

**Table 2** Feature extraction

Feature	Description
Mel spectrogram	Audio signal converted to mel spectrogram using STFT
MFCCs	Cosine transformation of mel spectrogram results in MFCCs
Chromagram	This is obtained by using STFT. These values represent the intensity of 12 pitch classes
Gabor	Filter bank consisting of 12 filters applied to the spectrogram

aspect ratio, and phase offset. Gabor filter bank consisting of 12 filters has been generated by varying angle in range  $[\pi/4, 3\pi/4]$  with  $\pi/4$  interval size and wavelength in range  $[\pi/8, \pi/2]$  with interval size  $\pi/8$ . These 12 filters are applied to the spectrogram obtained from each audio signal to generate features. Table 2 shows different methods used in this paper to extract features.

### 3.5 Neural Network Architecture

The Convolutional Neural Network (CNN) specializes in feature extraction and learning. These layers perform convolution operations on the data such as images using filters. Filters are applied to the image data to extract essential features and analyze the pattern. The pooling layer serves to downsample the output generated by each preceding layer. In the Neural Network, the Max Pooling operation was employed with stride values of (2, 2). The activation function plays a pivotal role in transforming the weighted input of a neuron into a specific range of output values. In the hidden layers, we've implemented the Rectified Linear Unit (ReLU) activation function. ReLU introduces nonlinearity into the network, empowering it to effectively learn complex relationships within the data. The BatchNormalization layer normalizes the input of the layer, ensuring that the layer receives a consistent distribution of outputs from the preceding layer. Figure 3 illustrates the layers comprising the convolutional neural network. The L1 layers function as the normalization layer, receiving spectrograms of size (128, 157, 3) generated during the feature extraction process. Normalization layer in CNNs stabilizes learning by maintaining consistent input ranges.

In the L2 layers, kernels sized (5, 5) with 64 filters and strides as (2, 2) are used. Each filter's output contributes to a 3D output, where the height of 3D output corresponds to the number of filters used. In layers L3 and L4, kernels of size (4, 4) and (1, 1) are used with respective strides of (2, 2) and (1, 1), each using 128 filters. MaxPooling with a pool size of (2, 2) is utilized for downsampling. Additionally, a dropout rate of 50% is incorporated to prevent overfitting. In the L5 layers, kernel sized (1, 1) with 128 filters and strides as (1, 1) are used. The L5 layers extract low-level features from the spectrogram. Two L5 layers are used consecutively to

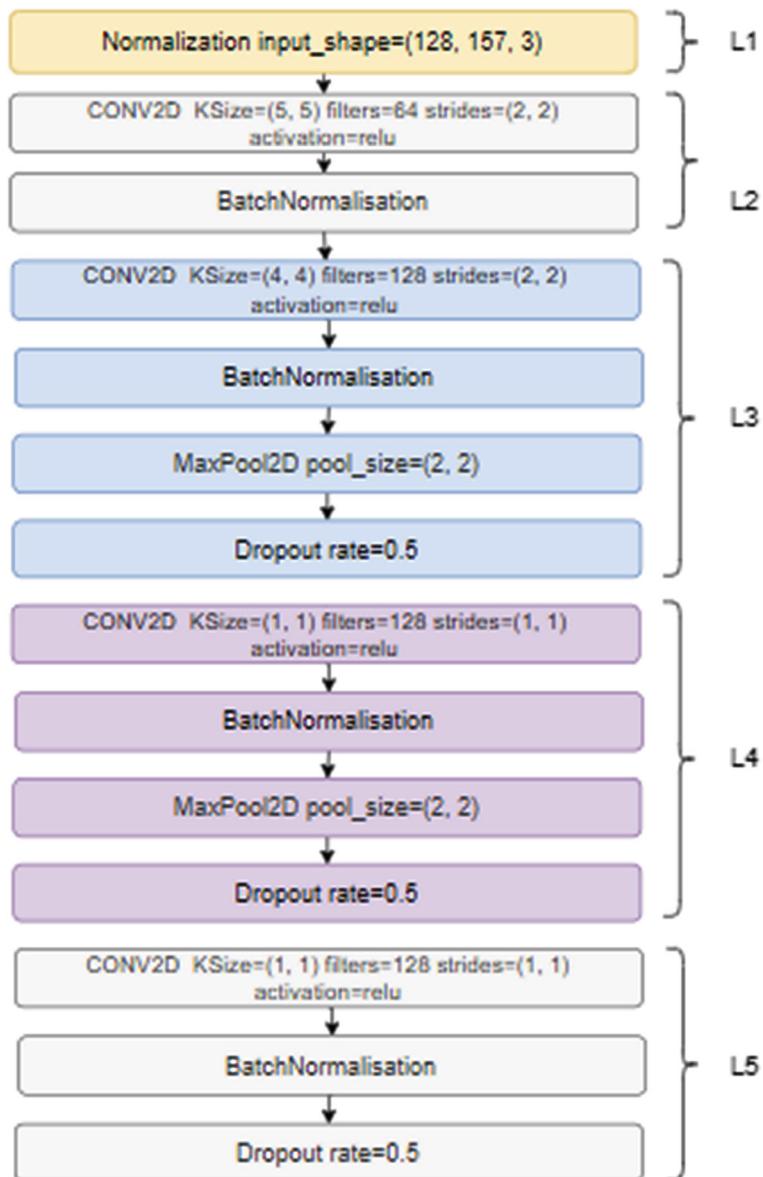
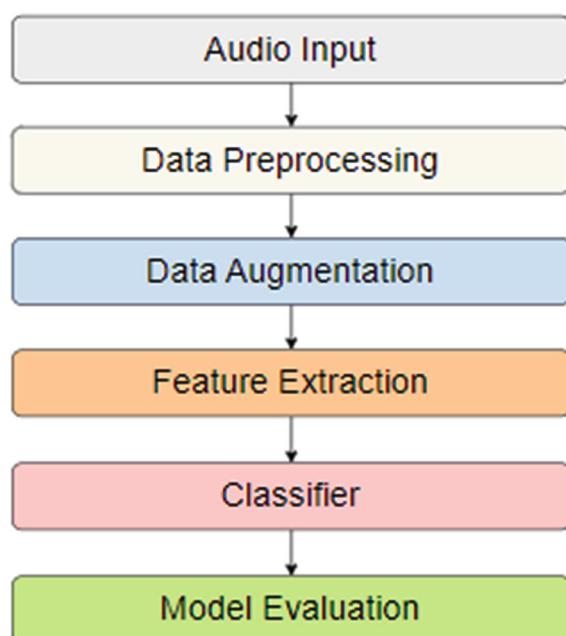


Fig. 3 CNN layer architecture

**Fig. 4** Experimental process

help the network in learning and analyzing patterns more effectively. We've added a dropout rate of 50% between layers to prevent overfitting. The second L5 layers act as the final output for the feature map.

### 3.6 Experiment Setup

The experimental process comprises five major steps, as depicted in Fig. 4.

Initially, the audio input undergoes preprocessing in the data preprocessing layer, where it is resampled, rechanneled, and resized. The output of the data preprocessing layer is then passed into the data augmentation layer, where four augmentation techniques are applied using the librosa library, resulting in an expanded dataset and improved model performance. In the feature extraction layer, four techniques, including mel spectrogram, MFCCs, Chromagram, and Gabor filter, are utilized. The feature map produced by the feature extraction layer serves as input to the classifier. A fully connected dense layer is utilized as the classifier, consisting of three layers. The first layer contains 1024 units with ReLU activation, while the second layer contains 512 units, also with ReLU activation. To mitigate overfitting, we've introduced a dropout rate of 40% between layers. The final layer serves as the output layer, uses softmax activation function and sparse categorical cross entropy as loss function. This activation function distributes probabilities across each output node, making it well-suited for multiclass classification tasks. Sparse categorical cross-entropy

is ideal for multiclass classification because it represents target labels as integers, efficient in terms of memory and computation compared to other loss functions like categorical cross entropy that uses one-hot encoded vectors to represent target labels. The process began with loading the CSV file that contains metadata and paths to the audio files, utilizing the Python pandas library. The CSV data were then fed into a data preprocessing pipeline, where preprocessing techniques such as resampling, rechanneling, and resizing were applied to convert the audio files into tensors. Next, these tensors were passed through a data augmentation pipeline. Various augmentation techniques, including time stretch, pitch shift, and time shift, were applied as detailed in Table 1. This step aimed to enhance the dataset's diversity and robustness. To compare unaugmented techniques with augmented techniques, data were passed through all the stages except the data augmentation pipeline. Following augmentation, the augmented audio data were fed into a feature extractor to obtain multiple essential features like mel spectrograms, MFCCs, and Gabor Mix features, as listed in Table 2. These features are crucial for capturing the distinct characteristics of the audio signals. The extracted features were then stored in memory for subsequent model training and evaluation. The data were split into training and testing sets in a 4:1 ratio, ensuring uniform class representation in both sets through stratified sampling and random shuffling with a random state of 42. For hyperparameter optimization, the Optuna framework was utilized to identify the best hyperparameters, including learning rate, number of epochs, optimizer, and batch size. Optuna conducted a series of trials to determine the optimal values for these hyperparameters. With the optimal hyperparameters, the CNN model was trained on the training data. An early stopping callback with a patience value of 15 and monitored metric as accuracy was used to avoid overfitting. After training, the model was evaluated on the test data using accuracy metrics, assessing its performance and the effectiveness of the augmentation and feature extraction techniques. Finally, the results obtained from different combinations of data augmentation and feature extraction techniques were recorded and compared to identify the best-performing configurations.

## 4 Result

The experiments are conducted on the Kaggle platform, utilizing the GPU P100 accelerator to accelerate the training of the deep learning model. In the initial experiment with the ESC-50 dataset, our proposed convolutional neural network achieved an accuracy of 58%. However, after incorporating various data augmentation and feature extraction techniques, we were able to significantly enhance the model's performance, achieving an impressive accuracy of 93.71%. This represents a substantial improvement over the previous method. Table 3 illustrates the best accuracy obtained from both augmented and unaugmented datasets, highlighting the efficacy of our approach in elevating model performance.

Hyperparameters play a crucial role as hidden variables within deep learning models, directly impacting their performance and accuracy. Within the model, four

**Table 3** Comparison of different methods on ESC-50 dataset

Method	Accuracy	
	Without augmentation (%)	With augmentation (%)
Mel spectrogram	52	84.8
MFCCs	58	<b>93.71</b>
Gabor mix	55	90.51

**Table 4** Best hyperparameter configuration for different features

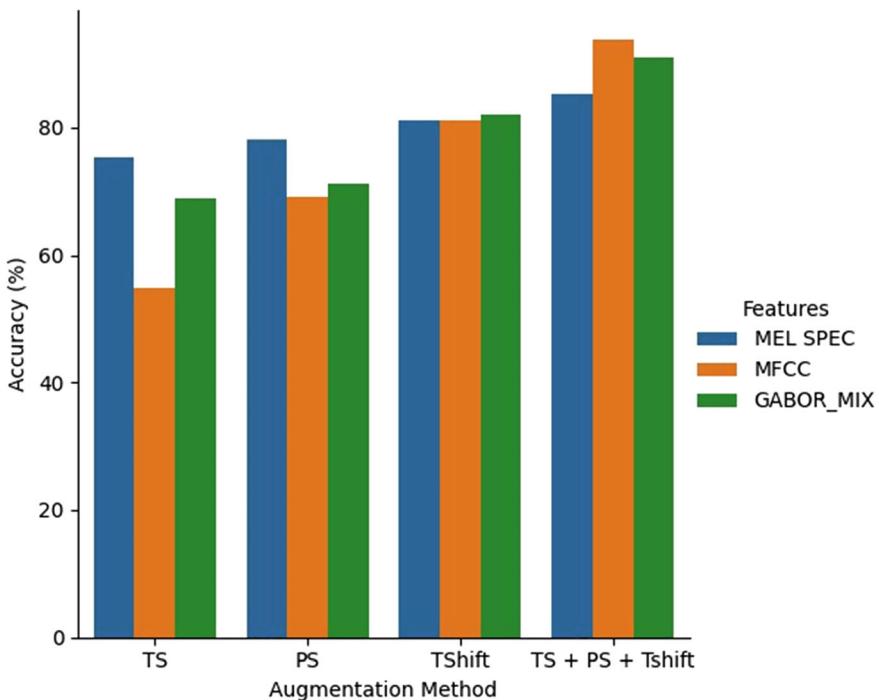
Feature	Learning rate	Number of epochs	Optimizer	Batch size
Mel spectrogram	0.06128	111	SGD	16
MFCCs	0.05185	102	SGD	32
Gabor	0.04265	100	SGD	32

critical hyperparameters are carefully considered: learning rate, number of epochs, optimizer selection, and batch size. To identify the most effective hyperparameter configurations, the Optuna hyperparameter optimization framework was used. Table 4 summarizes the suggested hyperparameter values obtained from a total of five trials, helping us in fine-tuning the model for optimal performance. In order to increase model performance, various augmentation techniques have been employed. Accuracy of the model over various augmentation techniques is depicted in the Fig. 5.

The MFCC feature, augmented with techniques like time stretch (ts), pitch shift (ps), and time shift (tshift), achieves a remarkable accuracy score of 93.71%. The analysis reveals that the SGD optimizer consistently outperforms Adam. One reason for this is the dataset size. SGD tends to converge with fewer samples compared to Adam, especially when dealing with complex datasets.

## 5 Conclusions

In this paper, we conducted a comprehensive comparison of Convolutional Neural Network (CNN) performance across various feature extraction and augmentation techniques. Unlike previous works that primarily rely on single-feature extraction techniques, we employ a combination of mel spectrogram, MFCCs, and Gabor filters, enhancing the richness of the features and improving classification performance. We also utilize multiple data augmentation techniques, including time stretch, pitch shift, and time shift, to expand the dataset and enhance model robustness. We propose a CNN model architecture that combines traditional convolutional neural networks (CNNs) with advanced feature extraction methods. This approach is designed to effectively capture spectral features of audio signals, offering a more sophisticated analysis compared to previous models like PiczakCNN [9]. Our study employs the



**Fig. 5** Model accuracy of various augmentation techniques

Optuna framework for hyperparameter optimization, ensuring that the model is fine-tuned for optimal performance. This level of hyperparameter tuning is often not detailed in existing studies, which typically use default settings or manual tuning. The effectiveness of the proposed CNN model was evaluated against existing approaches. Notably, the findings indicate that augmenting the MFCC feature with time stretch, time shift, and pitch shift techniques yielded an impressive accuracy of 93.71% for environmental sound classification.

In conclusion, our work contributes to advancing the field of environmental sound classification by presenting novel techniques and achieving state-of-the-art results. We believe that our findings will inspire further exploration and innovation in this area, ultimately benefiting applications ranging from wildlife monitoring to urban noise detection.

## References

1. Andreadis A, Giambene G, Zambon R (2021) Monitoring illegal tree cutting through ultra-low-power smart IoT devices. *Sensors* 21(22):7593
2. Zhang C et al (2023) Classification of complicated urban forest acoustic scenes with deep learning models. *Forests* 14(2):206
3. Mporas I et al (2020) Illegal logging detection based on acoustic surveillance of forest. *Appl Sci* 10(20):7379
4. Olteanu E et al (2018) Forest monitoring system through sound recognition. In: 2018 International conference on communications (COMM). IEEE
5. Bansal A, Garg NK (2022) Environmental sound classification: a descriptive review of the literature. *Intell Syst Appl* 16:200115
6. Paranyaya T et al (2024) A comparative study of preprocessing and model compression techniques in deep learning for forest sound classification. *Sensors* 24(4):1149
7. Boddapati V et al (2017) Classifying environmental sounds using image recognition networks. *Procedia Comput Sci* 112:2048–2056
8. Jeantet L, Dufourq E (2023) Improving deep learning acoustic classifiers with contextual information for wildlife monitoring. *Eco Inform* 77:102256
9. Piczak KJ (2015) Environmental sound classification with convolutional neural networks. In: 2015 IEEE 25th international workshop on machine learning for signal processing (MLSP). IEEE
10. Zhang Z et al (2021) Attention based convolutional recurrent neural network for environmental sound classification. *Neurocomputing* 453:896–903
11. Zhu B et al (2018) Learning environmental sounds with multi-scale convolutional neural network. In: 2018 international joint conference on neural networks (IJCNN). IEEE
12. Piczak KJ. ESC-50: dataset for environmental sound classification. <https://github.com/karolpiczak/ESC-50>

# Design and Implementation of Real-Time Environment Tracking System Using Internet of Things (IoT)



Ajeet Singh, Sanjay Singh, and Rupali Mahajan

**Abstract** Critical levels of rapid urbanization and skyrocketing population increase have been attained. Since air quality has a significant impact on human health and well-being, its monitoring has become an important task. This paper introduces the real-time Environment Tracking System (ETS) utilizing IoT that makes use of sensors DHT22, MQ135 GPS Module NODEMCU, and ThingSpeak. IoT-based ETS, which makes use of developments in embedded and sensor technologies, offers a cost-effective and adaptable substitute for conventional fixed monitoring systems. In order to effectively build IoT-based ETS architectures, the study performs an extensive assessment of enabling technologies, including sensors, communication protocols, and data processing algorithms. In addition, the discourse explores the importance of temperature and humidity in addition to air quality, taking into account their combined impacts on environmental circumstances. This research attempts to provide insights into the complex trade-offs involved in deploying IoT-based ETS by shedding light on the benefits and drawbacks of various technologies. In the end, it aims to provide a complete solution to resolve environmental issues.

**Keywords** Internet of things · DHT22 · MQ135 · GPS · NODEMCU · ThingSpeak

---

A. Singh (✉) · S. Singh · R. Mahajan

Department of ECE, Hindustan College of Science and Technology Mathura, Mathura, India  
e-mail: [ajeetsinghkain@gmail.com](mailto:ajeetsinghkain@gmail.com)

S. Singh  
e-mail: [sanjaysanju1001@gmail.com](mailto:sanjaysanju1001@gmail.com)

R. Mahajan  
e-mail: [rupali\\_mahajan61@rediffmail.com](mailto:rupali_mahajan61@rediffmail.com)

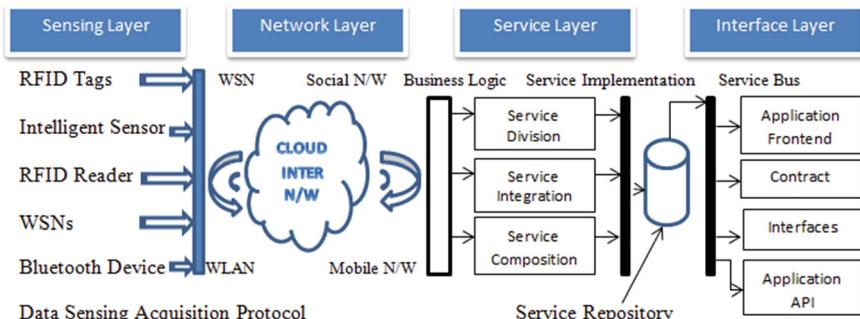
## 1 Introduction

The Internet of Things (IoT) is a revolutionary concept where physical objects like devices, vehicles, and buildings are embedded with electronics, software, sensors, and network connectivity [1]. This allows them to collect and exchange data, enabling remote sensing and control via existing networks. IoT integrates the physical and digital worlds, enhancing efficiency and accuracy across various fields [2].

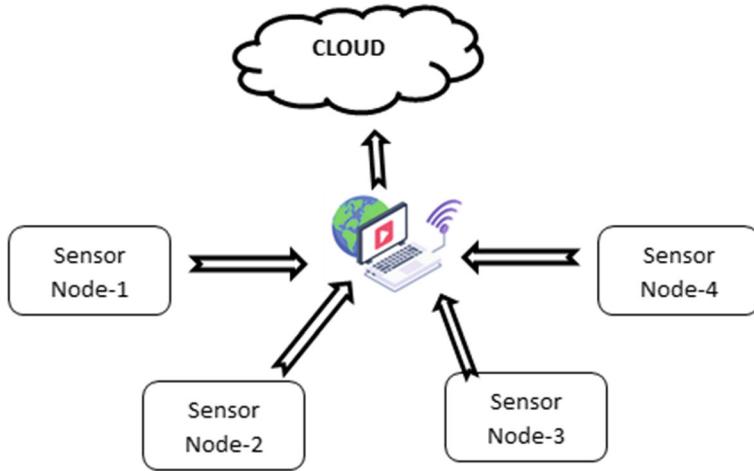
Figure 1 shows the layered architecture of IoT [3]. The architecture layer consists of four essential layers forming a system's foundation. The Sensing Layer gathers data from sensors, IoT devices, or user inputs. The Network Layer manages data transmission and routing, ensuring seamless communication across the system. The Service Layer processes, manipulates, and analyzes data, housing core logic, and algorithms for decision-making and value-added services. Finally, the Interface Layer enables interaction between the system and users or external entities through user interfaces, APIs, or integration points [4].

One of the defining characteristics of IoT is its ability to function autonomously [5], without the need for human intervention. While IoT technologies are still in their nascent stages, preliminary applications have emerged in sectors such as healthcare [6–8], transportation [9, 10], and automotive industries [11, 12]. However, the development of IoT system presents numerous challenges [13], in terms of infrastructure [14, 15], communication protocols [16, 17], interface design [18], and standardization efforts [19]. Despite these challenges, ongoing advancements continue to push the boundaries of IoT integration, paving the way for a more connected and intelligent future.

In the dynamics of daily life, the logistic environment undergoes rapid changes, often leading to health concerns due to prolonged exposure to certain conditions [20]. Monitoring these environmental changes is crucial for mitigating their potential health risks [21]. Internet of Things (IoT) technology presents a broad scope in this area. Using IoT system, monitoring, controlling, and accessing information within logistic environments become feasible. Amongst the essential parameters for environment monitoring, fire detection [22], temperature and humidity measurement



**Fig. 1** Architecture layer of IoT [3]



**Fig. 2** Sensor node system model [25]

stand out as foundational variables. To address this need, a cost-effective system has been designed that not only tracks environmental conditions but also provides insights into the specific location where these conditions are observed [23]. This integration of IoT with localization enhances the efficacy of monitoring logistic environments, ensuring timely interventions and safeguarding health and safety standards [24].

The network system architecture depicted in Fig. 2 provides an overview of the model's infrastructure. Users have the flexibility to deploy nodes arbitrarily within the environment. For the sake of simplicity and robustness, a star topology has been adopted. In this configuration, the failure of one device does not impact others, and data collisions are minimized [25].

The existing system for environment monitoring uses sensors (DHT22 and MQ135) for data collection and microcontroller (NODEMCU-ESP8266) for data processing and cloud (ThingSpeak) for data storage and visualization for the end user. But there is no mechanism for getting location information of the multiple sensor nodes using GPS Module, which helps in initiating action at the particular location where there is abnormal rise in temperature, humidity, and air quality.

## 2 Literature Review

Saima Zafar et al. proposed IoT-based environmental monitoring system [26]. This system employs Arduino and cloud services to monitor temperature and humidity levels in real time. The sensed data are uploaded to cloud storage to present the result to the end user.

Karthigaeni et al. have also introduced an Internet of Things-based wireless weather monitoring system using Blynk server [27]. The proposed system mainly captures about four environment parameters namely temperature, soil moisture, humidity, and vibration.

Hassan et al. have proposed an IoT-based Environment Monitoring System [25]. This model monitors the temperature, humidity, and harmful gases present in indoor and outdoor environment. Monitored data are stored on Web Server so that user can access anywhere in the world through internet connection. Users can also configure notifications to alert them of critical changes in sensor data. proposed system is low cost, accurate and user friendly.

Ghule et al. have modeled Web-Based Environment Monitoring System Using IOT [28]. The system is designed to gather data from various sensors and transmit their values directly to ThingSpeak, for data storage and process.

Shinde et al. have proposed Environment Monitoring System through Internet of Things (IOT) [29]. This model introduces an Internet of Things (IoT) system architecture tailored for monitoring indoor and outdoor environments using two sensor nodes and gateway. Gateway collects data from sensor node and publish on cloud so that it can be analyzed and appropriate decision can be made. Source of power supply for the system is solar panel.

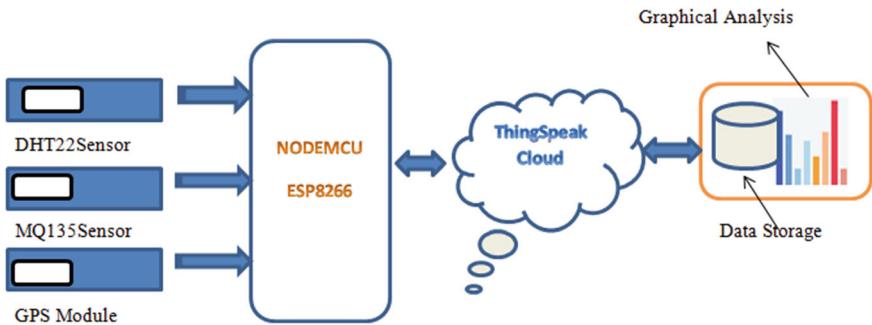
Ram et al. have made IoT-based data logger system for weather monitoring using wireless sensor networks [30]. The system deals with monitoring and controlling the environmental conditions like temperature, relative humidity, light intensity, and CO<sub>2</sub> level with sensors and sends the information to the web page and then plot the sensor data as graphical statistics. The data updated from the implemented system can be accessible on the internet from anywhere in the world.

### 3 Design Methodology and Implementation

To track the data continuously for different indices of environment like temperature, humidity, air quality, design methodology of the environment tracking system is divided into the following sections.

#### i. Hardware Block Diagram

Figure 3 illustrates the logical data flow model of the proposed system, within this framework, ThingSpeak service allocates specific channels to each node, with each channel possessing its unique API Key. This API Key plays a pivotal role in structuring and managing data within the channels, thereby ensuring the integrity of the database. Consequently, these segregated databases can be seamlessly visualized within the ThingSpeak platform or effortlessly exported to external services for comprehensive data analysis.



**Fig. 3** Hardware block diagram for the environmental tracking system

### ii. Sensor Nodes

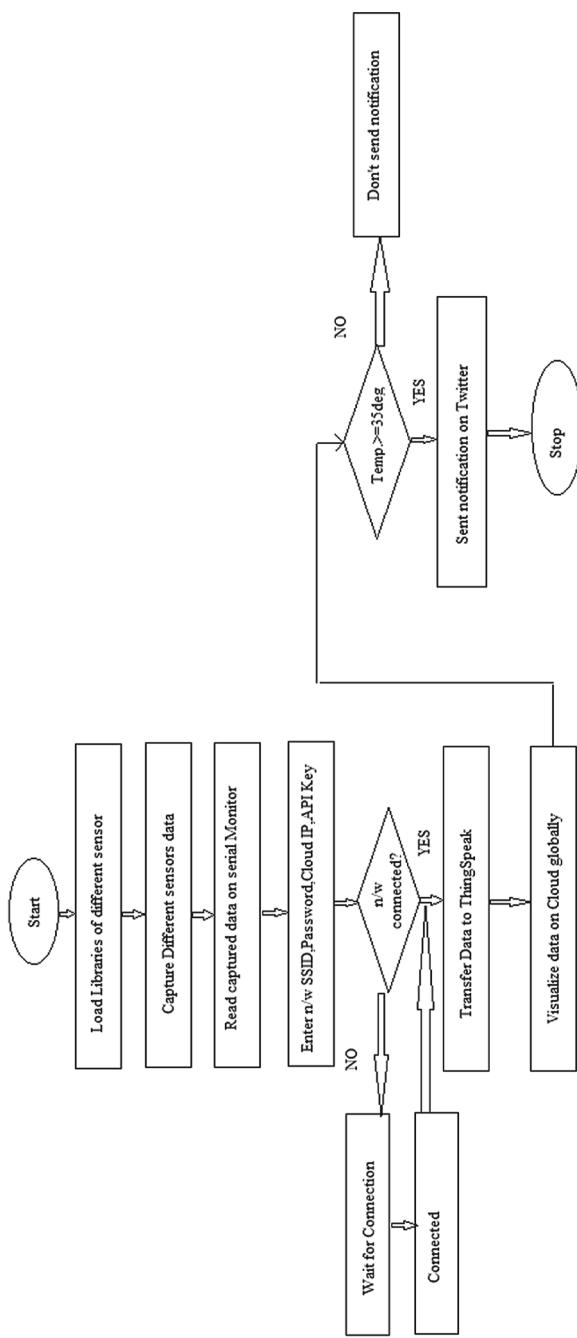
The proposed system integrates various types of sensors like DHT22, MQ135, GPS Module for temperature and humidity measurement, and gas detection and location tracking. By leveraging this array of sensors, the system can continuously capture and analyze data related to these critical environmental factors.

### iii. NODEMCU (ESP8266)

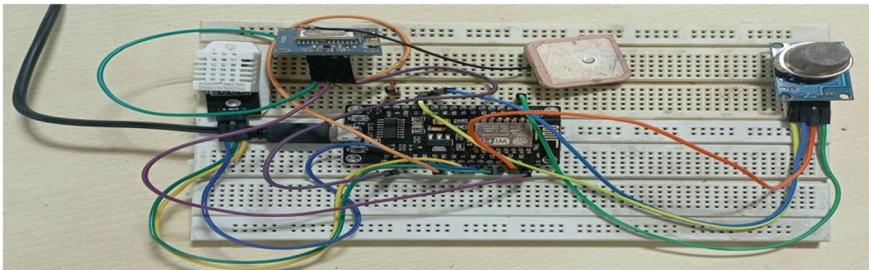
At the core of the entire device lies the ESP8266, serving as its central processing unit. This versatile microcontroller assumes multiple crucial responsibilities within the system. Firstly, it oversees the collection of sensor data, ensuring accurate and timely readings from the connected sensors. Subsequently, the ESP8266 is tasked with formatting the gathered data, preparing it for transmission to the sensor gateway. The ESP8266 also takes charge of sending the formatted data to the sensor gateway, facilitating the flow of information within the network.

### iv. Data Collection with ThingSpeak

To collect data using ThingSpeak, set up channels to organize and store data, with each channel configured for specific measurement parameters. Generate API keys for secure communication between IoT devices and ThingSpeak, ensuring only authorized devices can send data. Configure IoT devices to transmit data to ThingSpeak channels via HTTP or MQTT, connecting to Wi-Fi and sending data in the required format. As sensors collect data, it's sent in real-time to ThingSpeak for immediate visualization and analysis. Users can set triggers for notifications when conditions, like temperature exceeding 35 °C, are met. ThingSpeak's MATLAB tools allow for custom algorithms and advanced visualizations, and data can be exported in formats like CSV or JSON for further analysis. Regular monitoring ensures efficient and reliable data collection. Figure 4 represents the algorithm followed by model.



**Fig. 4** Flowchart of the proposed system



**Fig. 5** Connection diagram of the proposed system

## 4 Implementation of the Proposed System

To implement the proposed system using DHT22, MQ135, GPS Module, NODEMCU ESP8266, and ThingSpeak with tweet notifications, start by connecting the DHT22 and MQ135 sensors to the NODEMCU ESP8266, ensuring proper wiring and power supply. Configure the Arduino IDE with the necessary libraries for the sensors and ESP8266. Write code to read data from the sensors and connect the ESP8266 to Wi-Fi. Send the sensor data to ThingSpeak using its API for real-time visualization and analysis. For tweet notifications, use the Twitter API or a service like IFTTT, writing code to send tweets based on sensor thresholds. Ensure proper authentication with Twitter's API. This setup continuously monitors temperature, humidity, location, and air quality, uploads data to ThingSpeak, and sends tweet alerts for significant changes, enabling real-time monitoring and notifications (Fig. 5).

## 5 Results and Discussion

By using DHT22, GPS Module, and MQ135 sensors connected to a NODEMCU ESP8266 microcontroller, the system continuously gathers crucial environmental data such as temperature, humidity, location (latitude and longitude), and air quality. This setup ensures accurate and reliable data collection for effective monitoring. Integrating with ThingSpeak, a cloud-based IoT platform, allows seamless real-time transmission of sensor data for storage, visualization, and analysis. Through the ThingSpeak API, users can remotely monitor environmental conditions and analyze trends over time, leveraging GPS data to track the system's precise location (Tables 1 and 2).

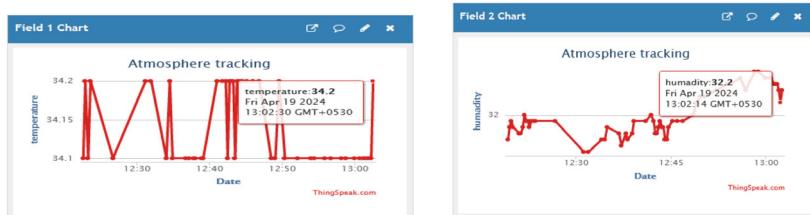
**Table 1** Summary of data on ThingSpeak for DHT22 and MQ135

Date/time	Entry	Temperature	Humidity	Gas index
2024-04-18T05: 01:30 + 00:00	128	31.8	32	13,462
2024-04-19T05: 38:29 + 00:00	150	33.8	32.3	22,065
2024-04-19T05: 48:14 + 00:00	158	33.9	31.9	19,787
2024-04-19T06: 45:19 + 00:00	191	34	32	28,631
2024-04-19T07: 12:44 + 00:00	229	34.2	31.7	22,307
2024-04-19T07: 31:12 + 00:00	253	34.1	32.5	23,298
2024-04-19T07: 31:27 + 00:00	254	34.1	32.5	23,553
2024-04-19T07: 31:43 + 00:00	255	34.1	32.5	23,553
2024-04-19T07: 31:59 + 00:00	256	34.1	32.4	23,298
2024-04-19T07: 32:14 + 00:00	257	34.1	32.2	25,690
2024-04-19T07: 32:30 + 00:00	258	34.2	32.4	25,970

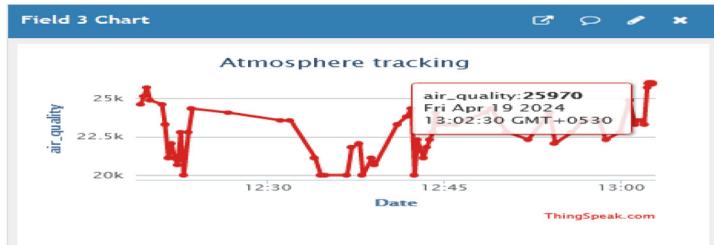
**Table 2** Summary of data on ThingSpeak for GPS Module

Date/time	Entry	Latitude	Longitude
2024-03-19T17: 24:45 + 00:00	1	27.18003	77.95294
2024-03-19T17: 25:57 + 00:00	2	27.18002	77.95295
2024-03-19T17: 26:35 + 00:00	3	27.18002	77.95296

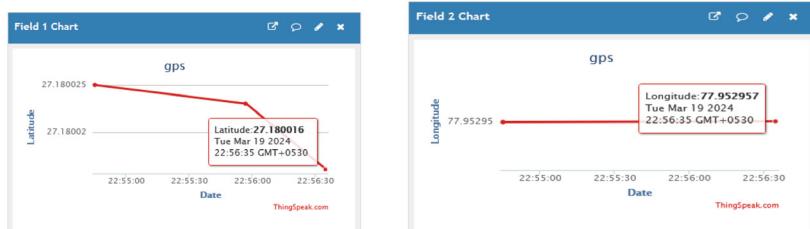
Figure 6a–c represents the Graphical analysis of all sensors DHT22, MQ135, GPS Module data that are used in the system.



(a) Temperature and Humidity Analysis using DHT22 Sensor



(b) Air Quality Analysis using MQ135 Sensor



(c) Location of sensor node in terms of latitude and longitude

**Fig. 6** **a** Temperature and humidity analysis using DHT22 sensor. **b** Air quality analysis using MQ135 sensor. **c** Location of sensor node in terms of latitude and longitude

## 6 Conclusions

The proposed Environment Tracking System uses affordable components to monitor temperature, humidity, location, and air quality in real time, suitable for both indoor and outdoor use. Data from sensors are sent via Wi-Fi to the ThingSpeak cloud platform for real-time access and graphical analysis. The system includes a notification feature that alerts users if any parameter exceeds a preset threshold, such as 35 °C for temperature. When this threshold is reached, a notification, like a tweet, is triggered. This system combines low-cost hardware, wireless connectivity, cloud storage, and threshold-based notifications, making it versatile, affordable, and effective for various environments.

## References

1. Madakam S, Ramaswamy R, Tripathi S (2015) Internet of things (IoT): a literature review. *J Comput Commun* 3(5):164–173
2. Gokhale P, Bhat O, Bhat S (2018) Introduction to IOT. *Int Adv Res J Sci, Eng Technol* 5(1):41–44
3. Pukkasenung P, Lilakiatsakun W (2021) Improved generic layer model for IoT architecture. *J Inf Sci Technol* 11(1):18–29
4. JabraeilJamali MA, Bahrami B, Heidari A, Allahverdizadeh P, Norouzi F, JabraeilJamali MA, Bahrami B, Heidari A, Allahverdizadeh P, Norouzi F (2020) IoT architecture. In: Towards the internet of things: architectures, security, and applications, pp 9–31
5. Braten AE, Kraemer FA, Palma D (2020) Autonomous IoT device management systems: structured review and generalized cognitive model. *IEEE Internet Things J* 8(6):4275–4290
6. Kashani MH, Madanipour M, Nikravan M, Asghari P, Mahdipour E (2021) A systematic review of IoT in healthcare: applications, techniques, and trends. *J Netw Comput Appl* 15(192):103164
7. Rejeb A, Rejeb K, Treiblmaier H, Appolloni A, Alghamdi S, Alhasawi Y, Iranmanesh M (2023) The internet of things (IoT) in healthcare: taking stock and moving forward. *Internet Things* 1(22):100721
8. Mathew PS, Pillai AS, Palade V (2018) Applications of IoT in healthcare. In: Cognitive computing for big data systems OverIoT: frameworks, tools and applications, pp 263–288
9. Humayun M, Jhanjhi NZ, Hamid B, Ahmed G (2020) Emerging smart logistics and transportation using IoT and blockchain. *IEEE Internet Things Mag* 3(2):58–62
10. Rey A, Panetti E, Maglio R, Ferretti M (2021) Determinants in adopting the internet of things in the transport and logistics industry. *J Bus Res* 1(131):584–590
11. Sekar RA, Prabakaran T, Sudhakar A, Kumar RS (2022, May 19) Industrial automation using IoT. In: AIP Conference proceedings, vol 2393, no 1. AIP Publishing
12. Breivold HP, Sandström K (2015, December 11) Internet of things for industrial automation—challenges and technical solutions. In: 2015 IEEE international conference on data science and data intensive systems. IEEE, pp 532–539
13. Karale A (2021) The challenges of IoT addressing security, ethics, privacy, and laws. *Internet Things* 15:100420
14. Moudgil V, Hewage K, Hussain SA, Sadiq R (2023) Integration of IoT in building energy infrastructure: a critical review on challenges and solutions. *Renew Sustain Energy Rev* 1(174):113121
15. Verma A, Prakash S, Srivastava V, Kumar A, Mukhopadhyay SC (2019) Sensing, controlling, and IoT infrastructure in smart building: a review. *IEEE Sens J* 19(20):9036–9046
16. Sabin CC (2020) A survey on architecture, protocols and challenges in IoT. *Wirel Pers Commun* 112(3):1383–1429
17. Dizdarević J, Carpio F, Jukan A, Masip-Bruin X (2019) A survey of communication protocols for internet of things and related challenges of fog and cloud computing integration. *ACM Comput Surv (CSUR)* 51(6):1–29
18. Gupta BB, Quamara M (2020) An overview of internet of things (IoT): architectural aspects, challenges, and protocols. *Concurr Comput: Pract Exp* 32(21):e4946
19. Al-Qaseemi SA, Almulhim HA, Almulhim MF, Chaudhry SR (2016, December 6) IoT architecture challenges and issues: lack of standardization. In: 2016 Future technologies conference (FTC). IEEE, pp 731–738
20. Haghi M, Neubert S, Geissler A, Fleischer H, Stoll N, Stoll R, Thurow K (2020) A flexible and pervasive IoT-based healthcare platform for physiological and environmental parameters monitoring. *IEEE Internet Things J* 7(6):5628–5647
21. Wolkoff P, Azuma K, Carrer P (2021) Health, work performance, and risk of infection in office-like environments: the role of indoor temperature, air humidity, and ventilation. *Int J Hyg Environ Health* 1(233):113709
22. Sungheetha A, Sharma R (2020) Real time monitoring and fire detection using internet of things and cloud based drones. *J Soft Comput Parad (JSCP)* 2(03):168–174

23. Ramnath S, Javali A, Narang B, Mishra P, Routray SK (2017, May 19) IoT based localization and tracking. In: 2017 International conference on IoT and application (ICIOT). IEEE, pp 1–4
24. Bhoi SK, Panda SK, Jena KK, Sahoo KS, Jhanjhi NZ, Masud M, Aljahdali S (2022) IoT-EMS: an internet of things based environment monitoring system in volunteer computing environment. *Intell Autom Soft Comput* 32(3):1493–1507
25. Hassan MN, Islam MR, Faisal F, Semantha FH, Siddique AH, Hasan M (2020) An IoT based environment monitoring system. In: Third international conference on intelligent sustainable systems [ICISS 2020], IEEE Xplore Part Number: CFP20M19-ART. ISBN: 978-1-7281-7089-3
26. Zafar S, Miraj G, Baloch R, Murtaza D, Arshad K (2018) An IoT based real-time environmental monitoring system using Arduino and cloud service. *Eng, Technol Appl Sci Res* 8(4):3238–3242
27. Karthigaeni K, Nithyalakshmi R (2020) Internet of things based wireless weather monitoring system using Blynk server. *Int Res J Mod Eng Technol Sci* 2(9):811–816
28. Ghule P, Kamble M (2019) Web based environment monitoring system using IOT. In: 3rd international conference on trends in electronics and informatics (ICOEI 2019). IEEE Xplore Part Number: CFP19J32-ART. ISBN: 978-1-5386-9439-8. [ieeexplore.ieee.org](http://ieeexplore.ieee.org)
29. Shinde VR, Tasgaonkar PP, Garg RD (2018) Environment monitoring system through internet of things (IOT). In: 2018 International conference on information, communication, engineering and technology (ICICET). <https://doi.org/10.1109/icicet.2018.8533835>
30. Ram KSS, Gupta ANPS (2016) IoT based data logger system for weather monitoring using wireless sensor networks. *Int J Eng Trends Technol* 32(2):71–75

# Automatic Weed Detection Using CNN



**Shubham Kumar Gupta, Sarthak Agarwal, Yash Garg, and Dilkeshwar Pandey**

**Abstract** Weeds are becoming a serious threat to the agricultural sector, which is acknowledged as the foundation of the Indian economy but is currently experiencing production issues. Plants that grow in inappropriate places are known as weeds, and they compete with crops for vital resources like water, light, nutrients, and space. This competition lowers crop yields and uses machinery inefficiently, which lowers agricultural productivity as a whole. Traditional weed control techniques include applying herbicide widely across the field or removing weeds by hand, which takes a lot of work. The latter approach, on the other hand, is considered ineffective since it pollutes the environment and offers little assistance in controlling weeds. There are financial and environmental issues associated with the widespread use of agricultural chemicals, such as fertilizers and herbicides. As a result, farmers are looking for alternatives more and more to reduce their reliance on chemicals in farming operations. Creative weed management strategies are becoming more and more necessary in response to these difficulties. The main goal is to distinguish between crops and weeds to provide a focused and effective weed management strategy. The agricultural industry may be able to increase productivity while lowering its impact on the environment and relying less on chemical solutions by implementing cutting-edge technologies for accurate weed identification and targeted eradication. The transition in weed control techniques towards technology-based and sustainable approaches is indicative of a wider movement in the agriculture sector to investigate environmentally friendly substitutes for a future that is more robust and fruitful.

**Keywords** Weed detection · Weed management · Agriculture sector · Convolution neural network · AlexNet · GoogLeNet

---

S. K. Gupta (✉) · S. Agarwal · Y. Garg · D. Pandey

Department of Computer Science and Engineering, KIET Group of Institutions, Ghaziabad, UP, India

e-mail: [skg2002007@gmail.com](mailto:skg2002007@gmail.com)

D. Pandey

e-mail: [dilkeshwar.pandey@kiet.edu](mailto:dilkeshwar.pandey@kiet.edu)

## 1 Introduction

The cornerstone of the Indian economy undeniably rests upon agriculture, a sector that sustains livelihoods for nearly half of the country's population. Given its paramount importance, ensuring the efficiency and productivity of agricultural practices becomes imperative. Thus, there arises a critical need to embrace cutting-edge cultivation techniques that not only optimize resources but also maximize crop yields. One of the primary challenges faced by farmers in this endeavor is the meticulous task of discerning weeds from the cultivated crop during the rinsing process. This seemingly mundane yet crucial aspect can significantly impact crop quality and quantity, underscoring the significance of innovative solutions and technologies in modern agricultural practices. Among a group of cultivated crops, weeds are extraneous plants that compete with the desired plants for nutrients, light, water, and space. The weeds can absorb the nutrients needed for crop growth. The yield may significantly decrease or be delayed in such a scenario. Therefore, it is necessary to prevent weed growth as much as possible. Furthermore, weeds will likely grow faster than crops. This is because the weed's seed or root is already in the ground and is just waiting for the right circumstances to sprout. This necessitates routine and frequent weed removal. When done by hand, this is a labor-and time-intensive process [1]. Identifying crops and weeds manually is a time-consuming task, requiring considerable labor to complete. The process involves distinguishing between desirable crops and unwanted weeds, a task that has become increasingly challenging in recent times. Traditionally, techniques for agricultural weed identification focused primarily on recognizing the weed species itself. However, as agricultural practices evolve and weed populations become more diverse and widespread, the complexity of accurately identifying and distinguishing weeds from crops has intensified. This heightened difficulty necessitates the development of more sophisticated and efficient methods for weed identification in plants, ensuring optimal management and maintenance of agricultural fields.

## 2 Literature Review

There has been a lot of work done to classify crops and weeds. Classification of crops and weeds has been a lengthy process. Sethia et al. [2] concluded that only a few percentage of fertilizers are reached to the root of plants which is very less effective. Authors of [3] have identified that most research is targeted towards unsupervised learning.

Authors of [4] identified three classes: apple scab, carpetweed, and crabgrass (weeds) by using the histogram based on color indices and tested with methods viz CNN with an accuracy of 93%, respectively. Other models like GoogLeNet are also available, AlexNet has also been tested and is very accurate with a high f1 score of more than 95% for the detection of weeds. In addition, research has been

carried out with the implementation of CNN for weed detection in unsupervised training data collection [3]. Research has been carried out on the detection of broad leaf weed in pasture using CNN models with an accuracy of 90%. The authors present CNN models for the classification of 16 plant species, including weeds, with a precision of 94%. In the case of weed species with an accuracy of 80%, similar work has been proposed to predict the growth stage [5]. The authors investigated the use of CNNs and obtained more than 90% accuracy with an average between all images above 85% to detect carpetweed and grass weeds in the soil. In this paper, traditional machine learning algorithms and deep learning models have been compared for the classification of seeds. By performing background segmentation, a good accuracy of 93.8% was achieved. For 16 different plant species with high precision, the authors have shown that CNNs are very effective in learning useful feature representations. Various approaches and systems for the classification of crops and weeds have been suggested to be introduced into the literature. The authors have tried to solve the problem using the CNN model Detection of weeds. Agriculture has always been vital to human existence [6]. Agriculture has begun to mechanize and digitize throughout the past century, and more specifically over the last 15 years. As a result of this development and automation, labor flow has become virtually entirely standardized. The data will be used for the prediction of the weed from the crop in the Convolutional Neural Networks (CNNs) and deep learning base model to find out the unwanted weeds and then suggest some herbicides. A machine vision technique may detect crops for weed management. Its characteristics, such as size, shape, spectral reflection, and texture, have detected weeds in agricultural fields. In this document, they have demonstrated the detection of weed by its size. Crop and weed detection using texture and size characteristics, as well as the automatic spraying of herbicides." They've been developing an image processing algorithm for crop discovery and weed management. Computer vision application for detection of undesirable weeds from one area which has an impact on agriculture. An Image is used to achieve the region of interest, and further processed through the technique using neural network.

### 3 Methodology

#### 3.1 Trend in Recent Year

Deep learning algorithms have been helpful in recent years for effectively analyzing text, picture, and spectrum data. Artificial intelligence uses a variety of deep learning methods to make it easier to identify weeds in photos. These algorithms are effective in analyzing data and identifying distinctive characteristics. Each digital image can be recognized as a 2D array of values, where each value corresponds to a greyscale code between 0 and 255. The convolutional, pooling, and dense layers get these pixel values after which they are fed [7]. Throughout this process, weights are adjusted in

accordance with how much the output and true label differ from one another. The methodologies employed in this investigation will be covered in the parts that follow.

### 3.2 Deep Neural Networks

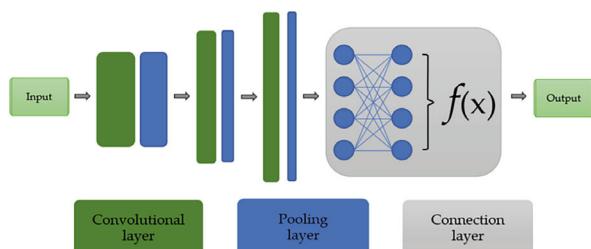
Weed detection is the primary goal of the suggested methodology. The convolutional neural network is proposed for weed detection. Figure 1 depicts the architecture of the suggested methodology. We tried to use CNN with a few conv2d layers, dropout, max\_pooling, and dense layers. Deep Learning (DL) is a type of machine learning algorithm characterized by sequential layers [8]. Unlike traditional machine learning methods that necessitate manual feature extraction, DL automatically selects features. A popular DL model known as Convolutional Neural Network (CNN) efficiently extracts features from input data, particularly in image analysis tasks. CNN's layered architecture allows it to identify and classify elements/pixels with minimal preprocessing. Typically, a CNN model consists of four main layers: convolutional, activation function, pooling, and fully connected layers (FCN) for classification purposes [9].

### 3.3 Procedure

In this part we will understand the characterization process for weeds into all the classes that we considered, we performed image processing on the dataset; images in the dataset are in RGB color code and have various dimensions (width and heights) [4]. AlexNet and GoogLeNet models use three input channels corresponding to red, green, and blue color codes, input dimensions for GoogLeNet are  $(224 \times 224)$  and AlexNet are  $(227 \times 227)$  [7].

We performed image processing in two steps. In the first step, all images are resized to conform to the input layer dimensions of AlexNet and GooleNet, and in the second step original image is duplicated three times for input channels (Red, Green, and Blue). We have used a transfer learning model to extract important information

**Fig. 1** The basic structure of CNN models



from the dataset images by identifying key details. Our models involve numerous convolutional neural networks (CNNs) stacked over each other. We have used two pre-trained models AlexNet and GoogLeNet, we have replaced the bottom layers of the model with three fully connected layers which helps in uniting data extracted by previous layers. We used a softmax layer to convert a vector of real values into probability distribution with k-potential outcomes and we used a softmax layer to normalize the output. Figure 2 below lists the hyper-parameters that were utilized during training [10].

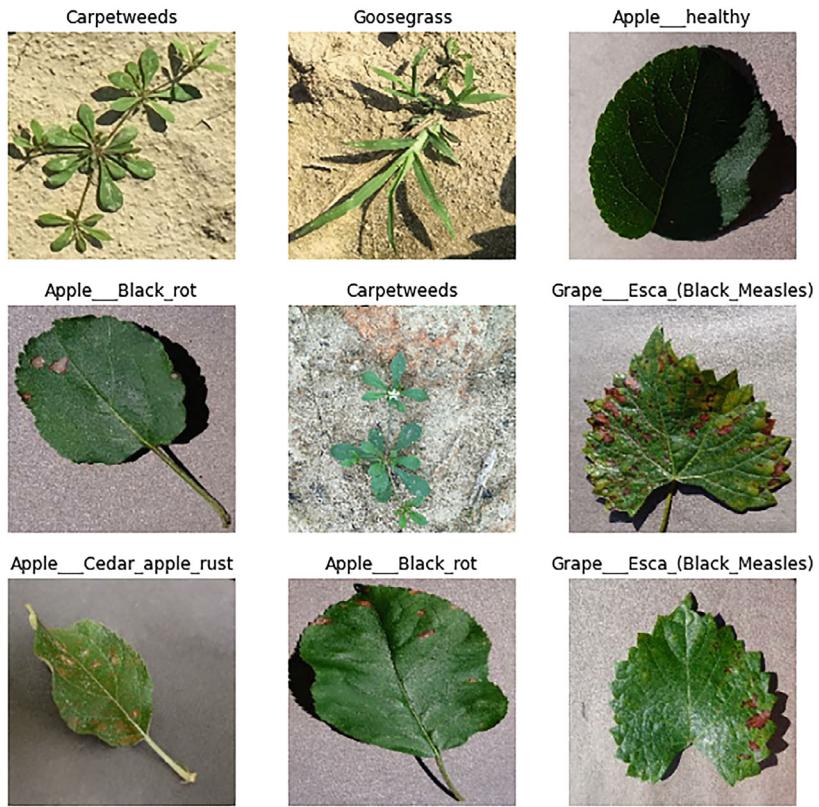
## 4 Dataset

For our experimental setup, we utilized a dataset sourced from the Kaggle website, consisting of approximately 6268 plant images, which are all in PNG format [11]. A training set and a validate set were created from this dataset, with a validate size ratio of 0.2 (Fig. 2).

Tiwari et al. [12] divided the dataset for training and testing of the model in which 20% part was used in training and the remaining used for testing the model.

Species	Training	Testing
Apple_Apple_scab	630	126
Apple_Black_rot	621	124
Apple_Cedar_apple_rust	275	55
Apple_healthy	786	157
Carpetweeds	763	152
Crabgrass	111	22
Goosegrass	216	43
Grape_Black_rot	1000	200
Grape_Esca_(Black_Measles)	1000	200
Grape_healthy	423	85
Tomato_Bacterial_Spot	96	19
Tomato_Early_Blight	46	9
Tomato_Healthy	73	15
Tomato_Leaf_mold	44	9
Tomato_Septorial_Leaf_Spot	82	16
Tomato_Yellow_Leaf_Curl_Virus	102	20

**Fig. 2** Different classes used in our model



**Fig. 3** Sample pictures from dataset

Subsequently, the training set consists of 5015 images, while the test set comprises 1253 images, totaling the initial 6268 images in the dataset. Our model was trained using the training set with 5015 images over 30 epochs, resulting in a comprehensive training process encompassing the entirety of the available image samples (Fig. 3).

## 5 Performance Analysis

To assess the efficacy and efficiency of various CNN designs in weed identification in agricultural contexts, a performance analysis of automatic weed detection using CNNs was carried out [13]. The primary focus of the test was on metrics such as accuracy, precision, recall, F1-score, and computing efficiency.

## 6 Result

The research paper concludes with a thorough investigation of machine learning-based automatic weed detection, emphasizing the effectiveness of various models on a particular dataset. Initial test accuracy of only 84% was obtained using an AlexNet CNN model and 90% was obtained using GoogLeNet , which produced less-than-ideal results. There was no overfitting or underfitting observed in our model [7]. Figure 5 displays the confusion matrices for weed identification using the chosen deep learning models. Figure 4 presents a comparison of all the models' precision, recall, and F1 scores using the best-performing set, trained over 30 epochs, and a 16-batch size. The obtained results demonstrate that the model can accurately and confidently identify weeds in crops (Fig. 5).

Classification Report:					
	precision	recall	f1-score	support	
Apple__Apple_scab	0.93	0.95	0.94	95	
Apple__Black_rot	0.95	0.98	0.96	94	
Apple__Cedar_apple_rust	0.90	0.90	0.90	42	
Apple__healthy	0.96	0.95	0.95	119	
Carpetweeds	1.00	0.97	0.98	89	
Crabgrass	0.78	0.78	0.78	18	
Goosegrass	0.83	0.91	0.87	33	
Grape__Black_rot	0.94	0.97	0.96	153	
Grape__Esca_(Black_Measles)	0.97	0.93	0.95	151	
Grape__healthy	1.00	0.98	0.99	64	
Tomato_Bacterial_Spot	0.79	0.73	0.76	15	
Tomato_Early_Blight	0.60	0.38	0.46	8	
Tomato_Healthy	0.90	0.75	0.82	12	
Tomato_Leaf_mold	0.88	0.88	0.88	8	
Tomato_Septoria_leaf_Spot	0.80	0.62	0.70	13	
Tomato_Yellow_Leaf_Curl_Virus	0.64	0.88	0.74	16	
accuracy			0.93		930
macro avg	0.87	0.85	0.85	930	
weighted avg	0.93	0.93	0.93	930	

Accuracy: 0.932258064516129

**Fig. 4** Accuracy of each class

Confusion Matrix:

[	[	90	0	1	3	0	0	0	0	0	0	0	0	0	0	0	0	1]
[	[	1	92	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0]
[	[	1	3	38	0	0	0	0	0	0	0	0	0	0	0	0	0	0]
[	[	4	1	1	113	0	0	0	0	0	0	0	0	0	0	0	0	0]
[	[	0	0	0	0	86	1	2	0	0	0	0	0	0	0	0	0	0]
[	[	0	0	0	0	0	14	4	0	0	0	0	0	0	0	0	0	0]
[	[	0	0	0	0	0	3	30	0	0	0	0	0	0	0	0	0	0]
[	[	0	0	0	0	0	0	0	149	4	0	0	0	0	0	0	0	0]
[	[	0	0	0	0	0	0	0	10	140	0	0	0	0	0	0	0	1]
[	[	0	0	0	1	0	0	0	0	0	63	0	0	0	0	0	0	0]
[	[	0	0	1	0	0	0	0	0	0	0	11	1	0	0	0	0	2]
[	[	0	0	0	0	0	0	0	0	0	0	1	3	1	0	2	1	1]
[	[	0	1	1	0	0	0	0	0	0	0	0	0	9	0	0	0	1]
[	[	0	0	0	0	0	0	0	0	0	0	1	0	0	7	0	0	0]
[	[	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	8	2]
[	[	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	14]

**Fig. 5** Confusion Matrix of our model

## 7 Conclusion

Weed detection using convolutional neural networks is a promising technique that facilitates agricultural operations automation. This study illustrated CNN models' applicability in the field of weed detection. Two machine learning models (AlexNet, GoogLeNet) have been used in this work to identify weeds that are present in the field. Real-time crop and weed detection based on the CNN models' decisions is one area of possible future investigation.

Our model's accuracy in the experiment was 93.2%. We concluded that our suggested approach might more accurately and swiftly predict weeds than the manual method. This demonstrates the great potential of deep learning in the agricultural sector. You will be able to identify weeds much more quickly by employing this strategy.

### 1. Accuracy:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

where:

$TP$  = True Positives (correctly identified weeds)

$TN$  = True Negatives (correctly identified non-weeds)

$FP$  = False Positives (incorrectly identified as weeds)

$FN$  = False Negatives (missed identification of weeds)

### 2. Precision:

$$\text{Precision} = \frac{TP}{TP + FP}$$

### 3. Recall:

$$\text{Recall} = \frac{TP}{TP + FN}$$

### 4. F1-score:

$$F1 - score = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

We have created a user-friendly web interface, especially for farmers as part of our creative project. Farmers are empowered by this interface, which makes it easy for them to choose photos from a gallery and upload them to the platform. After the photos are uploaded, the interface's built-in algorithms examine them and, astonishingly, identify any weeds. Farmers greatly benefit from this feature, which makes it possible for them to quickly locate and eradicate weed infestations in their fields. To further promote agricultural efficiency, we've included a helpful resource: a link to comprehensive guidelines on practical weed removal techniques. This resource gives farmers the skills and information they need to successfully manage weed growth, maximizing crop yield and guaranteeing the success of their farming endeavors.

## 8 Future Scope

Convolutional Neural Network (CNN) models, such as AlexNet and GoogLeNet, provide great promise for automatic weed detection in a variety of agricultural and environmental management applications.

Precision Agriculture: Using CNN models to identify weeds can improve methods of precision agriculture. Farmers can administer targeted herbicide treatments, lim-

iting chemical usage and environmental impact while maximizing crop production, by properly recognizing and localizing weeds within crops.

**Crop management:** By differentiating between undesirable weeds and crops, weed detection CNN models can help monitor the health of crops. By using this data, crop management practices can be optimized by timely interventions like selective harvesting and irrigation modifications.

**Environmental Conservation:** By reducing the environmental impact of pesticide use, accurate weed detection using CNN models promotes sustainable agriculture methods.

All things considered, the future of automatic weed detection with CNN models such as AlexNet and GoogLeNet resides in its many applications in scientific research, environmental management, and agriculture, providing creative answers for effective and sustainable weed management techniques.

## References

1. Johnson R, Mohan T, Paul S (2020) Weed detection and removal based on image processing. *Int J Recent Technol Eng (IJRTE)* 8(6):347–352
2. Sethia G, Guragol HKS, Sandhya S, Shruthi J, Rashmi N (2020) Automated computer vision based weed removal bot. In: 2020 IEEE international conference on electronics, computing and communication technologies (CONECCT). IEEE, pp 1–6
3. Hasan ASMM, Sohel F, Diepeveen D, Laga H, Jones MGK (2021) A survey of deep learning techniques for weed detection from images. *Comput Electron Agric* 184:106067
4. Zhangnan W, Chen Y, Zhao B, Kang X, Ding Y (2021) Review of weed detection methods based on computer vision. *Sensors* 21(11):3647
5. Umamaheswari S, Arjun R, Meganathan D (2018) Weed detection in farm crops using parallel image processing. In: 2018 conference on information and communication technology (CICT). IEEE, pp 1–4
6. Aravind R, Daman M, Kariyappa BS (2015) Design and development of automatic weed detection and smart herbicide sprayer robot. In: 2015 IEEE recent advances in intelligent computational systems (RAICS). IEEE, pp. 257–261
7. Subeesh A, Bhole S, Singh K, Chandel NS, Rajwade YA, Rao KVR, Kumar SP, Jat D (2022) Deep convolutional neural network models for weed detection in polyhouse grown bell peppers. *Artif Intell Agric* 6:47–54
8. Muhammad Hamza Asad and Abdul Bais (2020) Weed detection in canola fields using maximum likelihood classification and deep convolutional neural network. *Inf Process Agric* 7(4):535–545
9. Hema MS, Abhilash V, Tharun V, Reddy DM (2022) Weed detection using convolutional neural network. *BOHR Int J Intell Instrum Comput* 1(1):46–49
10. García-Navarrete OL, Correa-Guimaraes A, Navas-Gracia LM (2024) Application of convolutional neural networks in weed detection and identification: a systematic review. *Agriculture* 14(4):568
11. Haq MA (2022) Cnn based automated weed detection system using uav imagery. *Comput. Syst. Sci. Eng.* 42(2)

12. Tiwari O, Goyal V, Kumar P, Vij S (2019) An experimental set up for utilizing convolutional neural network in automated weed detection. In: 2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU). IEEE, pp. 1–6
13. Islam N, Rashid MM, Wibowo S, Xu C-Y, Morshed A, Wasimi SA, Moore S, Rahman SM (2021) Early weed detection using image processing and machine learning techniques in an Australian Chilli farm. Agriculture 11(5):387

# Height Measurement of Pose Bent Knees by Using Pose Estimation of MediaPipe



Nguyen Phan Kien, Hoang Van Thao, Duc-Tan Tran,  
and Vijender Kumar Solanki

**Abstract** Height is an important human parameter used in many fields such as healthcare, beauty, sports, etc. There are many different non-contact height measurement methods in the world, but most of these methods only support measuring a person's height in a standing straight posture, at a fixed distance. Therefore, in this study, we developed a system to support measuring human height in a bent-knee posture at different distances by extracting the coordinates of human skeletal landmarks using MediaPipe Pose Estimation. Moreover, our system also uses a multiple linear regression model to improve the accuracy of human height estimation. From the experimental results, our model has a relatively low error of about 2.46 cm (~1.44%). In the future, the research will be extended to support measuring human height in different postures such as sitting, walking, lying down, etc.

**Keywords** Height measurement · Non-contact measurement · Multiple linear regression · MediaPipe · Yolov8 · 2D space

## 1 Introduction

MediaPipe Pose Estimation [1] is a powerful open-source library developed by Google. It is a great tool for estimating human pose from images or real-time video. It has been used in various studies such as evaluating push-up exercises [2], or a system to support recognition of exercise postures [3], etc. These studies have

---

N. P. Kien (✉) · H. Van Thao

School of Electrical and Electronic Engineering, Hanoi University of Science and Technology,  
Hanoi, Vietnam

e-mail: [Kien.nguyễnphan@hust.edu.vn](mailto:Kien.nguyễnphan@hust.edu.vn)

D.-T. Tran

Phenikaa University, Hanoi, Vietnam

V. K. Solanki

Stanley College of Engineering and Technology For Woman, Hyderabad, TS, India

leveraged MediaPipe for tasks like motion analysis, and posture adjustment for therapeutic purposes. However, it has not been widely applied to support human height measurement. Therefore, in this study, we will take advantage of the excellent ability of MediaPipe Pose Estimation in extracting coordinates of body landmarks to build a non-contact height measurement support system.

Typically, height measurement requires the person being measured to stand in an ideal upright posture. This makes it difficult to support measuring human height in a bent-knee posture. Therefore, in this study, we propose a new non-contact height measurement method to support non-contact height measurement in a bent-knee posture at different distances. We use the available tools of MediaPipe pose estimation and a smartphone camera to extract the skeletal coordinates of the human body, with the support of a reference object such as a blackboard of fixed size. The coordinates are represented as  $X, Y$  in 2D space. From these coordinates, we will apply geometric formulas to calculate the length of each corresponding skeletal segment, and from there estimate the height. Moreover, to improve the accuracy of human height estimation, we have applied machine learning models to train on the dataset we collected.

This research focuses on non-contact human height measurement in a bent-knee posture in 2D space, aiming to build an algorithm to support non-contact human height measurement from body skeletal coordinates. From there, we can build a system to support measuring human height in different postures.

## 2 Literature Review

Estimating height from images is an important research area with many applications such as healthcare, beauty, human recognition, motion tracking, sports analysis, and augmented reality. Previous studies have used various methods to estimate body height.

Some studies have used the length of a body part such as the face [4], knee [5], and arm [6] to estimate height. Ye-Peng Guan proposed a method using a single-face image to estimate body height [4]. This method requires extracting some facial features such as eyes, lips, chin to estimate height according to statistical data and the golden ratio on the face. This method achieved quite accurate results; however, the height measurement results of this method are affected by the distance and standing posture of the person being measured. Moreover, the results of this method were only tested on four samples, so the accuracy of the method is still not entirely clear.

In another study based on the height of adult women in Thailand, Nopphanath Chumpatth and colleagues used knee height to estimate height [5]. This method achieved relatively good results with an error of about 2.8 cm. However, the study only focused on a dataset of adult women in Thailand aged 18–59 years old. Therefore, the results of the study cannot be applied to other age groups and genders.

The method of extracting body coordinates is a common method used to recognize and measure human height from images or videos. Dong-seok Lee and colleagues

proposed a method for measuring height using Mask R-CNN to detect the human body region [7]. The highest point of the head and the lowest point of the foot will be used to estimate the body height. However, this method also has certain limitations. Factors such as the standing posture of the person being measured, shooting angle, distance, and low light can affect the accuracy of the method.

The methods presented above only focus on supporting height measurement of people in a standing straight posture, at a fixed distance, and studying on a small group of subjects. Therefore, these methods have difficulty in supporting height measurement of people in a bent-knee posture at different distances.

### 3 Methodology

In order to measure human height at different distances from a single image, we need a reference object with a known size. Therefore, in this study, we use a black rectangular object with a known actual size of width  $w_{\text{real}} = 20 \text{ cm}$  and height  $h_{\text{real}} = 30 \text{ cm}$  to serve for estimating human height. First, we need to determine the bounding box around our reference object. To do this, we use the YOLOv8 model [8] to detect the reference object in the image dataset that we collected. The YOLOv8 model has been proven to be a high-performance object detection model for fast and accurate object recognition. After detecting the reference object, we extract its width and height in the image frame as  $w(\text{pixels})$  and  $h(\text{pixels})$ , respectively. From there, we calculate the factor  $k = \frac{h_{\text{real}}}{h}$  (cm/pixel), which will be used to compute the length of each human body skeleton segment in the subsequent steps.

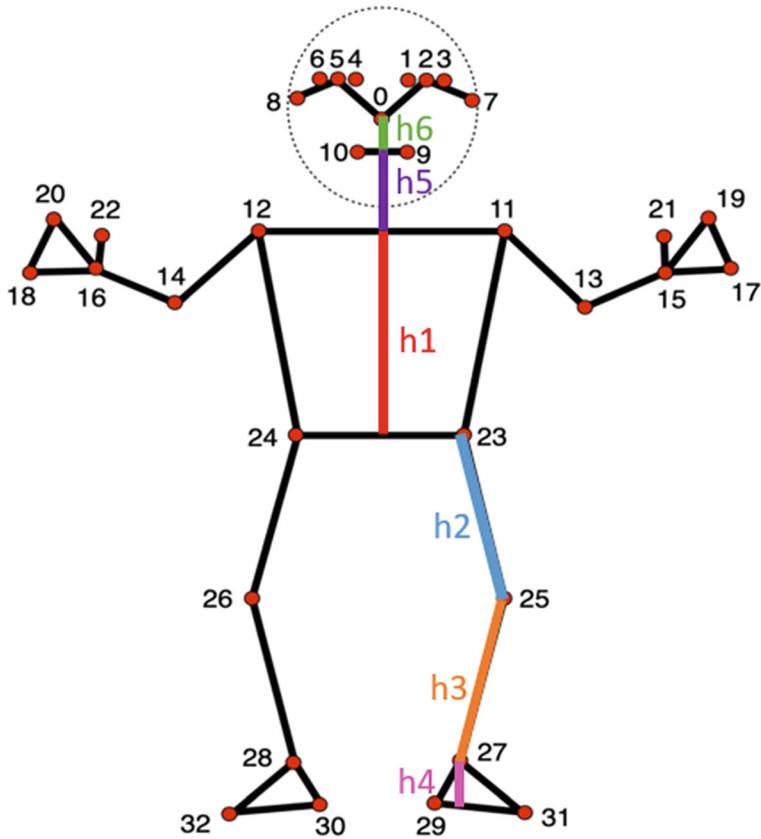
Other height measurement methods only support measuring human height in a standing upright position. Therefore, these methods seem infeasible for measuring height in a bent knee pose. To estimate human height in a bent knee pose, we will divide the human body into different segments based on the skeleton, including:  $h_1$  is the distance from the shoulders to the hips,  $h_2$  is the distance from the hips to the knees,  $h_3$  is the distance from the knees to the ankles,  $h_4$  is the distance from the ankles to the soles of the feet,  $h_5$  is the distance from the midpoint of the shoulders to the midpoint of the mouth, and finally  $h_6$  is the distance from the midpoint of the mouth to the nose. In this project, our calculations are based on the normalized  $X$  and  $Y$  coordinates obtained from MediaPipe (Fig. 1).

We use geometric formulas to calculate the length of the skeleton segments in 2D space.

The distance between two points  $A(x_A, y_A)$  và  $B(x_B, y_B)$  is:

$$AB = \sqrt{(x_A - x_B)^2 + (y_A - y_B)^2} \quad (1)$$

The coordinate of the midpoint between two points  $A$  and  $B$  is:



**Fig. 1** Map of landmarks and skeletons according to MediaPipe

$$M\left(\frac{x_A + x_B}{2}; \frac{y_A + y_B}{2}\right) \quad (2)$$

The equation of a straight line  $\Delta_1$  with a normal vector  $\vec{n}$  is:

$$\begin{aligned} \vec{n} &= (a; b) = (y_A - y_B; x_B - x_A) \\ (y_A - y_B)(x - x_A) + (x_B - x_A)(y - y_A) &= 0 \\ \text{or } ax + by + c &= 0 \end{aligned} \quad (3)$$

The distance from a point  $D$  to a straight line  $\Delta_1$  is given by:

$$d(D, \Delta_1) = \frac{|ax_d + by_d + c|}{|\vec{n}|} \quad (4)$$

From these 2D formulas, we can calculate the distances between landmark points and the lengths of the skeletal frame in the human body.

First, the distance  $h_1$ , which is the distance from the shoulder to the hip, can be calculated as the distance from the midpoint of the shoulders to the midpoint of the hips. Applying formula (2), we can find the coordinates of the midpoint between the shoulders as  $(\frac{X_{11}+X_{12}}{2}; \frac{Y_{11}+Y_{12}}{2})$ . Similarly, the coordinates of the midpoint between the hips are  $(\frac{X_{23}+X_{24}}{2}; \frac{Y_{23}+Y_{24}}{2})$ . Then, by applying formula (1), we can calculate the distance from the shoulder to the hip as follows:

$$h_1 = k \times \sqrt{\left(\frac{X_{23} + X_{24}}{2} - \frac{X_{11} + X_{12}}{2}\right)^2 + \left(\frac{Y_{23} + Y_{24}}{2} - \frac{Y_{11} + Y_{12}}{2}\right)^2} \quad (5)$$

The skeletal segment  $h_2$  is the distance from the hip to the knee, calculated as the distance between points 23 and 25. Applying formula (1), we can compute the length  $h_2$  as follows:

$$h_2 = k \times \sqrt{(X_{25} - X_{23})^2 + (Y_{25} - Y_{23})^2} \quad (6)$$

Similar to the way we calculated the distance from the hip to the knee, we can compute  $h_3$ , which is the distance from the knee to the ankle. The distance  $h_3$  is calculated as the distance between points 27 and 25, as follows:

$$h_3 = k \times \sqrt{(X_{27} - X_{25})^2 + (Y_{27} - Y_{25})^2} \quad (7)$$

Applying formula (4), we calculated the distance from the ankle (landmark 28) to the sole of the left foot as follows:

$$h_4 = k \times \frac{|(Y_{29} - Y_{31})x_{27} + (X_{31} - X_{29})Y_{27} + (Y_{31} - Y_{29})X_{29} - (X_{31} - X_{29})Y_{29}|}{\sqrt{(Y_{29} - Y_{31})^2 + (X_{31} - X_{29})^2}} \quad (8)$$

The distance  $h_5$  is the distance from the midpoint of the shoulders to the midpoint of the mouth. Applying formula (2), we can calculate the coordinates of the midpoint between the shoulders as  $(\frac{X_{11}+X_{12}}{2}; \frac{Y_{11}+Y_{12}}{2})$ . Similarly, the coordinates of the midpoint of the mouth are  $(\frac{x_9+x_{10}}{2}; \frac{y_9+y_{10}}{2})$ . By applying formula (1), we can compute the distance from the midpoint of the shoulders to the midpoint of the mouth using the following formula:

$$h_5 = k \times \sqrt{\left(\frac{X_{11} + X_{12}}{2} - \frac{X_9 + X_{10}}{2}\right)^2 + \left(\frac{Y_{11} + Y_{12}}{2} - \frac{Y_9 + Y_{10}}{2}\right)^2} \quad (9)$$

Finally,  $h_6$  is the distance from the midpoint of the mouth to the nose. Applying formula (1), we can calculate the distance from the midpoint of the mouth to the nose

using the following formula:

$$h_6 = k \times \sqrt{\left( X_0 - \frac{X_9 + X_{10}}{2} \right)^2 + \left( Y_0 - \frac{Y_9 + Y_{10}}{2} \right)^2} \quad (10)$$

Moreover, to improve the effectiveness of human height prediction, in this project, we apply machine learning models to train the model on the dataset we collected. We use the multiple linear regression model—a method used to model the relationship between a dependent variable and multiple independent variables [9]. This model has proven to be effective in predicting output results when the independent variables are known. The multiple linear regression equation has the form [10]:

$$Y = \beta_0 + \beta_1 * X_1 + \beta_2 * X_2 + \dots + \beta_n * X_n + \varepsilon \quad (11)$$

where:

$Y$ : dependent variable

$X_i$ : independent variable

$\beta_i$ : parameter

$\varepsilon$ : error.

After calculating the length of each skeleton segment, we apply the multiple linear regression equation to predict human height. The equation takes the form:

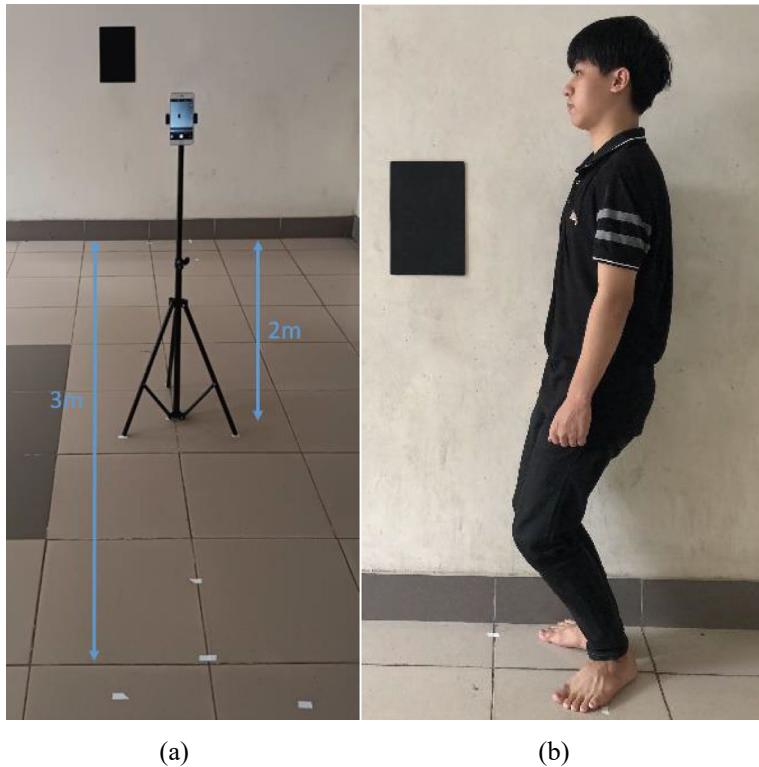
$$h = \beta_0 + \beta_1 h_1 + \beta_2 h_2 + \beta_3 h_3 + \beta_4 h_4 + \beta_5 h_5 + \beta_6 h_6 + \varepsilon \quad (12)$$

where  $h$  is the predicted height, and  $h_1, h_2, \dots, h_6$  are the lengths of the skeleton segments as presented above.

## 4 Experiment Setup

Our system includes a smartphone camera to capture images. The camera is mounted on a tripod, ensuring it is perpendicular to the ground and at a height of 115 cm from the ground. We capture the subject at two different distances of 2 and 3 m. For the measurement subject, they are asked to stand in a bent-knee pose, with their legs forming a 45° angle, keeping their back straight, eyes looking forward, and maintaining this pose throughout the measurement process.

Next, we will use MediaPipe to extract the  $X, Y$  coordinates of the landmarks on the human body and the YOLov8 model that we have trained to detect the reference object, as shown in Fig. 2. We will use the coordinates of these landmarks to calculate the length of each corresponding skeletal segment. Then, we remove outlier values by using the mean and standard deviation (SD) [11]. Values outside the mean range



**Fig. 2** Experiment description. **a** Set up tripod and camera. **b** Subject's experimental posture

will be removed to increase the accuracy of the model. Valid values will be used to train and test the model (Fig. 3).

## 5 Results

After training the model on our dataset with 128 samples for training and 33 samples for testing the model's accuracy, we derived Eq. (13) to estimate height from the lengths of the skeleton segments. The results are presented in Table 1. From the results in Table 1, we note that this method has a maximum error of 6 cm with an average error of 2.46 cm (~1.44%) across the 33 test data samples.

The errors in this method are attributed to lack of camera calibration, as well as inaccuracies in extracting coordinates from MediaPipe, and varying lighting conditions during data collection.

$$h = 0.364h_1 + 0.811h_2 + 0.587h_3 + 0.584h_4$$

**Fig. 3** Extract the landmarks and detect the reference object



**Table 1** The evaluation results of the 33 test subjects

33 Samples	Actual height (cm)	Predicted height (cm)	Error (cm)	Error rate (%)
Average	170.73	171.51	2.46	1.44
Maximum	180	182.65	6.00	3.53

$$+ 0.354h_5 - 0.211h_6 + 73.779 \quad (13)$$

## 6 Conclusion and Product Orientation

Our proposed method is to extract human skeletal coordinates from MediaPipe and use a multiple linear regression model to measure human height in a bent knee pose from a single image. From the experimental results, the method shows quite good accuracy with an average error of 2.46 cm (~1.44%) over 33 test data samples. These results indicate that this is a promising method to estimate human height in various poses. It can provide an effective height measurement method in various situations. In the future, we will try to expand our dataset to be able to estimate height for other poses such as sitting, walking, lying down, etc.

## References

1. Kulkarni S, Deshmukh S, Fernandes F, Patil A, Jabade V (2023) PoseAnalyser: a survey on human pose estimation. SN Comput Sci 4(2):136. <https://doi.org/10.1007/s42979-022-01567-2>
2. Nguyen PK, Nguyen AT, Doan TB, Trung PN, Thi ND (2023) Lecture notes in networks and systems, pp 581–589. [https://doi.org/10.1007/978-3-031-27524-1\\_55](https://doi.org/10.1007/978-3-031-27524-1_55)
3. Supanich W, Kulkarnieetham S, Sukphokha P, Wisarnsart P (2023) Machine learning-based exercise posture recognition system using MediaPipe pose estimation framework. In: 2023 9th international conference on advanced computing and communication systems (ICACCS), Coimbatore, India, pp 2003–2007. <https://doi.org/10.1109/ICACCS57279.2023.10112726>
4. Guan Y-P (2009) Unsupervised human height estimation from a single image. J Biomed Sci Eng
5. Chumpathat N, Rangsin R, Changbumrung S, Soonthornworasiri N, Durongritchai V, Kwanbunjan K (2015) Use of knee height for the estimation of body height in Thai adult women. Original Article
6. Jarzem PF, Gledhill RB (1993) Predicting height from arm measurements. J Pediatr Orthop 13(6):761–765
7. Lee D-S, Kim J-S, Jeong SC, Kwon S-K (2020) Human height estimation by color deep learning and depth 3D conversion. Appl Sci 10(16)
8. Jiang P, Ergu D, Liu F, Cai Y, Ma B (2022) A review of YOLO algorithm developments. Procedia Comput Sci 199:1066–1073. <https://doi.org/10.1016/j.procs.2022.01.135>
9. Olive DJ (2017) Multiple linear regression. Springer
10. Uyanik GK, Güler N (2013) A study on multiple linear regression analysis. Procedia - Soc Behav Sci 106:234–240. <https://doi.org/10.1016/j.sbspro.2013.12.027>
11. Yang J, Rahardja S, Fränti P (2019) Outlier detection: how to threshold outlier scores? In: Proceedings of the international conference on artificial intelligence, information processing and cloud computing (AIIPCC '19), Sanya, China. Association for Computing Machinery, New York, NY. <https://doi.org/10.1145/3371425.3371427>

# Non-contact Height Measurement in 2D with the Pose of 45° Side Standing



Nguyen Phan Kien, Dong Quoc Dat, Duc-Tan Tran,  
and Vijender Kumar Solanki

**Abstract** This article introduces a novel non-contact method for measuring height in a two-dimensional (2D) space, utilizing a standing position tilted 45° to one side. Traditional height measurement techniques often necessitate direct physical contact, which can be uncomfortable and inconvenient. Additionally, although a number of non-contact height measurement methods exist today that can be effective, they usually depend on images of subjects standing upright or perpendicular to the camera, thereby limiting their versatility and accuracy in various situations. Our method leverages 2D images to accurately estimate an individual's height without physical interaction by analyzing images of subjects standing at a 45° angle to the camera. We have developed a height estimation algorithm based on key reference points, reference objects, and body proportions. Experimental results show the effectiveness of this method, with a low average error of about 1.3%, demonstrating high accuracy and consistency compared to conventional height measurement methods. This technique has potential applications in various fields, including health care, security, and ergonomic design.

**Keywords** Non-contact height measurement · 2D image · 45° side pose

---

N. P. Kien (✉) · D. Q. Dat

School of Electrical and Electronic Engineering, Hanoi University of Science and Technology,  
Hanoi, Vietnam

e-mail: [Kien.nguyenphan@hust.edu.vn](mailto:Kien.nguyenphan@hust.edu.vn)

D.-T. Tran

Phenikaa University, Hanoi, Vietnam

V. K. Solanki

Stanley College of Engineering and Technology For Woman, Hyderabad, TS, India

## 1 Introduction

The utilization of MediaPipe pose estimation has been extensively explored in various research domains, including applications in exercise assistance, gaming, and rehabilitation [1, 2]. These studies have leveraged MediaPipe for tasks such as movement analysis, posture correction, and interactive gaming experiences for therapeutic purposes. However, in terms of research on height measurement, height has been estimated from linear measurements of body parts such as arm span, knee height, and half-arm span [3]. One limitation of these studies is their narrow focus on adults aged 18–40, which may not accurately reflect the diversity of the adult population, particularly as height can vary with age, including potential declines in older age groups. Another approach has used facial ratios [4], calculating height from facial images by extracting facial features using CNN models and predicting height using artificial neural networks. However, the error rate or average deviation of their height measurements from facial images is approximately 0.073 m, a significant error in height measurement. While these methods can be effective, they often rely on images of subjects standing straight or perpendicular to the camera, which can limit their flexibility and accuracy in different scenarios.

In this study, we propose a new method to measure height by designing the subject to stand at an angle of 45° from the camera. This 45° tilt position aims to evaluate the feasibility and accuracy of non-contact height measurement in two-dimensional (2D) space. The method used to identify and measure human height from images is the body frame extraction technique [5]. This method uses computer vision and image processing techniques to analyze and interpret visual data to accurately estimate a person's height. However, the accuracy of this method can be affected by factors such as camera focal length, camera angle, and ambient lighting conditions.

To improve the accuracy of height measurement based on the above methods, we propose to research using MediaPipe to extract coordinates of key points of the skeleton from images taken from cameras, with the support of a reference object such as a blackboard has a fixed size. These coordinates, expressed as X and Y coordinates in 2D space, are used to calculate the length of bone segments, thereby estimating height. Additionally, using machine learning models to train data also helps improve measurement accuracy.

By analyzing images taken in this specific pose, we aim to develop an algorithm that accurately estimates an individual's height based on key reference points, reference objects, and scale body. Our study seeks to expand the application of non-contact height measurement techniques and evaluate their effectiveness compared to traditional methods.

## 2 Methodology

In other studies, determining height without direct contact often results in significant errors due to various factors such as calibration errors, subject movement and environmental conditions used for height estimation. Therefore, in this study, we utilize a black rectangular object as a reference, with a width of  $\text{image}_{\text{width}} = 20.5 \text{ cm}$  and a length of  $\text{image}_{\text{height}} = 30.5 \text{ cm}$  to measure the length of each skeletal segment (Fig. 3).

With the help of the reference object, we can combine the use of the body frame extraction technique to measure height. Specifically, we use YOLOv8 to detect the object, from which we can calculate the length and width of the object in pixels. With this information, we compute a conversion factor  $k$  (cm/pixel). This factor will be used to calculate the length of each skeletal segment of the human body in subsequent steps.

To calculate the height measurement of a person, we divide the human body into segments based on the skeletal frame, including:  $h_1$  as the distance from the shoulders to the hips,  $h_2$  as the distance from the hips to the knees,  $h_3$  as the distance from the knees to the ankles,  $h_4$  as the distance from the ankles to the soles of the feet,  $h_5$  as the distance from the midpoint between the shoulders to the midpoint between the mouth, and finally  $h_6$  as the distance from the midpoint between the mouth to the nose. In this project, our calculations are based on normalized  $X$  and  $Y$  coordinates obtained from Mediapipe [6] (Fig. 2).

Our computations are based on normalized coordinates  $x_i$  and  $y_i$  obtained from Mediapipe. From these coordinates, we convert to coordinates in the pixel coordinate system using the formula:

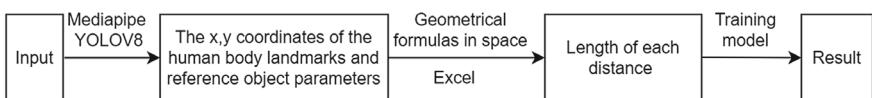
$$X_i = \text{image}_{\text{width}} * x_i \quad (1)$$

$$Y_i = \text{image}_{\text{height}} * y_i \quad (2)$$

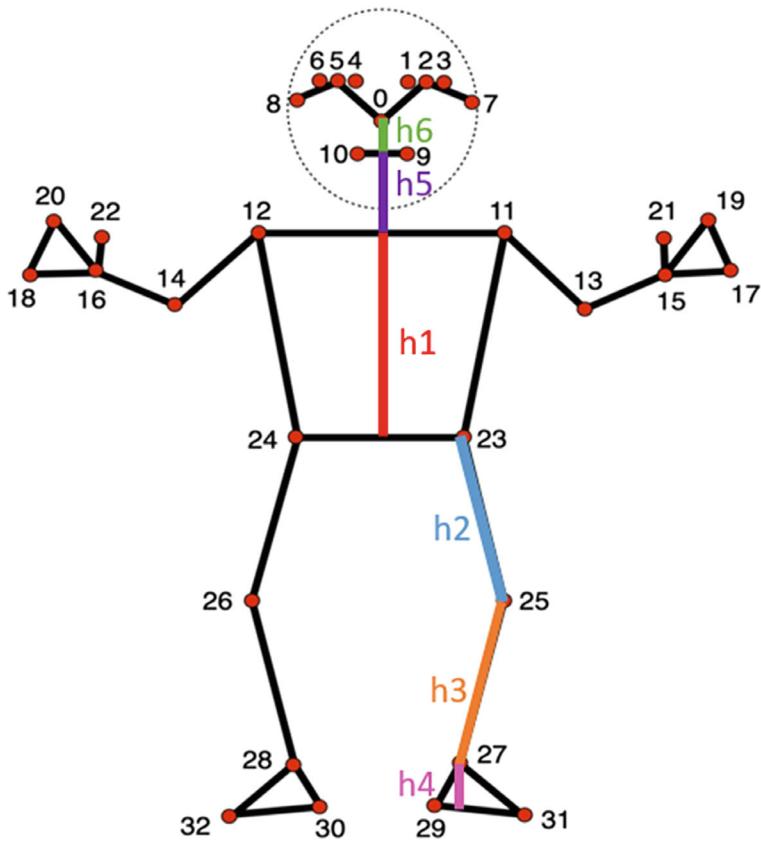
We use geometric formulas to calculate the lengths of skeletal segments in 2D space: The distance between two points  $A(x_A, y_A)$  and  $B(x_B, y_B)$  is calculated using the formula:

$$AB = \sqrt{(x_A - x_B)^2 + (y_A - y_B)^2} \quad (3)$$

The coordinates of the midpoint between two points AB:



**Fig. 1** Diagram of the data processing of the algorithm



**Fig. 2** Map of landmarks and skeletons according to MediaPipe

$$M\left(\frac{x_A + x_B}{2}; \frac{y_A + y_B}{2}\right) \quad (4)$$

The equation of the line  $\Delta_1$  with the normal vector  $\vec{n}$ :

$$\begin{aligned} \vec{n} &= (a; b) = (y_A - y_B; x_B - x_A) \\ (y_A - y_B)(x - x_A) + (x_B - x_A)(y - y_A) &= 0 \end{aligned} \quad (5)$$

or  $ax + by + c = 0$

The distance from a point to a line, the distance from point  $D$  to line  $\Delta_1$  is calculated using the formula:

$$d(D, \Delta_1) = \frac{|ax_d + by_d + c|}{|\vec{n}|} \quad (6)$$

From these formulas in 2D space, we can calculate the distance between landmark points and the length of skeletal segments in the human body.

Firstly, the distance  $h_1$  from the shoulders to the hips can be calculated as the distance between the midpoint of the shoulders and the midpoint of the hips. Applying formula (4), we can calculate the midpoint of the shoulders with coordinates  $(\frac{X_{11}+X_{12}}{2}; \frac{Y_{11}+Y_{12}}{2})$ . Similarly, we have the coordinates of the midpoint of the hips as  $(\frac{X_{23}+X_{24}}{2}; \frac{Y_{23}+Y_{24}}{2})$ . From there, applying formula (3), we can calculate the distance from the shoulders to the hips as follows:

$$h_1 = k \times \sqrt{\left(\frac{X_{23} + X_{24}}{2} - \frac{X_{11} + X_{12}}{2}\right)^2 + \left(\frac{Y_{23} + Y_{24}}{2} - \frac{Y_{11} + Y_{12}}{2}\right)^2} \quad (7)$$

The skeletal segment  $h_2$ , representing the distance from the hips to the knees, is calculated as the distance between points 23 and 25. Applying formula (3), we can compute the length of  $h_2$  as follows:

$$h_2 = k \times \sqrt{(X_{25} - X_{23})^2 + (Y_{25} - Y_{23})^2} \quad (8)$$

Like how we calculate the distance from the hips to the knees, we can compute  $h_3$ , representing the distance from the knees to the ankles. The distance  $h_3$  is calculated as the distance between points 27 and 25, computed as follows:

$$h_3 = k \times \sqrt{(X_{27} - X_{25})^2 + (Y_{27} - Y_{25})^2} \quad (9)$$

Applying formula (6), we can calculate the distance from the ankles (landmark 28) to the left sole of the foot as follows:

$$h_4 = k \times \frac{|(Y_{29} - Y_{31})x_{27} + (X_{31} - X_{29})Y_{27} + (Y_{31} - Y_{29})X_{29} - (X_{31} - X_{29})Y_{29}|}{\sqrt{(Y_{29} - Y_{31})^2 + (X_{31} - X_{29})^2}} \quad (10)$$

Distance  $h_5$  represents the distance from the midpoint between the shoulders to the midpoint between the mouth. Applying formula (4), we can calculate the midpoint between the shoulders with coordinates  $(\frac{X_{11}+X_{12}}{2}; \frac{Y_{11}+Y_{12}}{2})$ . Similarly, we have the coordinates of the midpoint between the mouth as  $(\frac{x_9+x_{10}}{2}; \frac{y_9+y_{10}}{2})$ . Applying formula (3), we can compute the distance from the midpoint between the shoulders to the midpoint between the mouth with the equation as follows:

$$h_5 = k \times \sqrt{\left(\frac{X_{11} + X_{12}}{2} - \frac{x_9 + x_{10}}{2}\right)^2 + \left(\frac{Y_{11} + Y_{12}}{2} - \frac{y_9 + y_{10}}{2}\right)^2} \quad (11)$$

Finally,  $h_6$  represents the distance from the midpoint between the mouth to the nose. Applying formula (3), we can calculate the distance from the midpoint between the mouth to the nose using the formula below:

$$h_6 = k \times \sqrt{\left( X_0 - \frac{X_9 + X_{10}}{2} \right)^2 + \left( Y_0 - \frac{Y_9 + Y_{10}}{2} \right)^2} \quad (12)$$

To optimize human height calculation, several methods can be utilized including linear regression, Principal Component Analysis (PCA), Support Vector Machines (SVM), or K-Nearest Neighbors (KNN). But in this method, we utilize one of the fundamental machine learning algorithms, which is multiple linear regression. This algorithm is an extension of simple linear regression and has been proven to be highly effective in predicting outcomes based on two or more given independent variables [7].

The equation for multiple linear regression is given by

$$y = w_0 + w_1x_1 + w_2x_2 + w_3x_3 + \dots + w_nx_n \quad (13)$$

With the above equation,  $y$  represents the dependent variable, which is the quantity to be predicted.  $x_1, x_2, x_3, \dots, x_n$  are the independent variables.  $w_0, w_1, w_2, w_3, \dots, w_n$  are the corresponding coefficients.

After obtaining the length of each skeletal segment, we apply the multiple linear regression equation to predict the height of the person. Equation (13) becomes

$$h = h_0 + w_1h_1 + w_2h_2 + w_3h_3 + \dots + w_nh_n \quad (14)$$

In Eq. (14),  $h$  is the predicted height.  $h_0, h_1, h_2, h_3, \dots, h_n$  represent the heights of the respective skeletal segments.  $w_0, w_1, w_2, w_3, \dots, w_n$  are the correlation coefficients obtained during the training process of the multiple linear regression model.

### 3 Experiment Setup and Results

#### 3.1 Experimental Devices and Configuration

During the experiment, we used the following devices to build the height measurement model for humans. We utilized the camera of a smartphone to capture images. This camera was mounted on a tripod, ensuring that the plane containing the camera was perpendicular to the ground plane. A distance measuring system was used to determine two distances from the camera to the measuring object, which were 2 m and 3 m, respectively. The distance from the camera to the ground and from the reference object to the ground were both 115 cm, as shown in Fig. 3.



**Fig. 3** Experimental setup

### 3.2 Measurement and Data Collection

For the person being measured, they assumed a predetermined position: stand at a 45° angle to the camera and looking straight ahead (not at the camera), as illustrated in Fig. 4. During the measurement process, the subject was instructed to breathe gently and remain still without moving.

After obtaining the input images, we extracted the  $x$  and  $y$  coordinates of key landmarks on the human body using MediaPipe. We utilized the mean and standard deviation (SD) values [8] of the data to reduce errors. Data beyond the range of  $\pm 3\text{SD}$  was filtered out, and the remaining data points were used to compute the bone segments in 2D space. Using machine learning—a multiple linear regression equation based on these bone segments—we estimated the height. Our dataset comprised 162 samples, including 129 training samples and 33 test samples to evaluate the effectiveness of the proposed method.



**Fig. 4** Experimental posture

### 3.3 Evaluation

Based on our algorithm and the training process with 162 samples, including 129 for training and 33 for evaluating the effectiveness of the proposed method, we derived an Eq. (15) that relates the true height of the measured subject to the lengths of the bone segments and we also obtained the results as presented in Table 1. From the results in Table 1, we observed that the measurements obtained by this method have an average estimation error of 2.22 cm (~1.3%) compared to the actual height of the subjects, within a height range from 162 to 180 cm.

The errors in the proposed method are attributed to factors such as the lack of camera calibration, affecting the accuracy of the measurements. Additionally, the process of capturing landmarks on the body and extracting coordinates from

**Table 1** The evaluation results of the 33 test subjects

33 Samples	Actual height	Predicted height	Error (%)
Average	170.73	170.43	1.30
Maximum	180	178.75	3.09

the MediaPipe software may introduce certain inaccuracies due to the discrepancy between the virtual space and the physical space, as well as environmental factors such as varying light intensities.

$$\begin{aligned}
 H = & -0.700h_1 + 0.988h_2 + 0.769h_3 + 0.350h_4 \\
 & + 0.681h_5 - 0.406h_6 + 72.229
 \end{aligned} \tag{15}$$

## 4 Conclusion and Product Orientation

The proposed method involves non-contact height measurement using MediaPipe combined with the YOLOv8 model for coordinate extraction, bone length calculation, and the use of a multiple linear regression function to predict human height from images. Experimental results show that the estimated height compared to the actual height has an average error of 2.22 cm (~1.3%). The level of accuracy achieved indicates that this level of error may be acceptable for various applications. Future research will expand to determine the height of individuals in different standing and lying positions, with the goal of developing a versatile and effective software application for height measurement in diverse contexts.

## References

1. Kwon Y, Kim D (2022) Real-time workout posture correction using OpenCV and MediaPipe. *한국정보기술학회논문지* 20(1):199–208
2. Latreche A et al (2033) Reliability and validity analysis of MediaPipe-based measurement system for some human rehabilitation motions. *Measurement* 214:112826
3. Digssie A, Argaw A, Belachew T (2018) Developing an equation for estimating body height from linear body measurements of Ethiopian adults. *J Physiol Anthropol* 37:1–8
4. Haritosh A et al (2019) A novel method to estimate height, weight and body mass index from face images. In: 2019 Twelfth international conference on contemporary computing (IC3). IEEE
5. Lee D-S et al (2020) Human height estimation by color deep learning and depth 3D conversion. *Appl Sci* 10(16):5531
6. Lugaresi C et al (2019) Mediapipe: a framework for building perception pipelines. *arXiv preprint arXiv:1906.08172*
7. Nathans LL, Oswald FL, Nimon K (2012) Interpreting multiple linear regression: a guidebook of variable importance. *Pract Assess Res Eval* 17(9):n9
8. Livingston EH (2004) The mean and standard deviation: what does it all mean? *J Surg Res* 119(2):117–123

# A Navigation Tracking Line Algorithm for the Mobile Robot Based on Traditional Vision



Khoa Nguyen Dang , Pham Tuan Minh, Duc-Tan Tran,  
and Vijender Kumar Solanki

**Abstract** Effective navigation for mobile robots based on ground lines is crucial across various domains such as manufacturing automation, transportation, and healthcare. This paper introduces a novel approach wherein robots utilize lines on the ground to navigate through different positions within environments, employing a PID controller algorithm and infrared (IR) ray feedback sensor. Despite the inherent noise sensitivity of IR sensors to varying light conditions, our proposed method leverages traditional vision techniques to accurately compute angle differences of lines within the visual frame, subsequently providing error feedback for the PID controller. Experimental results demonstrate the high-performance tracking capability of the mobile robot, achieving an error control of less than 0.02 cm. Comparative analysis with IR sensors further validates the effectiveness of our proposed algorithm, showcasing its potential for enhancing mobile robot navigation systems.

**Keywords** Mobile robot · Vision · Moment of image · PID

## 1 Introduction

A mobile robot is a robotic system designed to move and operate in its environment autonomously or under human control. Unlike stationary robots, mobile robots are equipped with mobility capabilities that enable them to navigate through various

---

K. N. Dang · P. T. Minh

Faculty of Applied Sciences International School, VNU, Hanoi, Vietnam

e-mail: [Khoand@vnuis.edu.vn](mailto:Khoand@vnuis.edu.vn)

P. T. Minh

e-mail: [21070741@vnuis.edu.vn](mailto:21070741@vnuis.edu.vn)

D.-T. Tran ()

Faculty of Electrical and Electronic Engineering, Phenikaa University, Hanoi, Vietnam

e-mail: [tan.tranduc@phenikaa-uni.edu.vn](mailto:tan.tranduc@phenikaa-uni.edu.vn)

V. K. Solanki

Stanley College Of Engineering & Technology For Women, Hyderabad, India

terrains, interact with their surroundings, and perform tasks such as inspection, transportation, or exploration.

Mobile robots find applications in a wide range of fields including factory logistics [1], public transportation [2], and office environments [3, 4], which can be categorized into several types based on their design, locomotion capabilities, and intended applications such as wheeled robots, tracked robots, legged robots, etc. Among them, wheeled robots are very popular because efficiency in controlled environments, simple and reliable design, cost-effectiveness, versatility, payload capacity, navigation and control, and user-friendliness. Therefore, a mobile robot with three wheels is selected to develop a controller in this paper.

Based on the mechanics of a mobile robot, the core of the system is the controller system which can with navigation involve the process of determining its path from a starting point to a destination. Some methods to navigate robot movements such as contour tracking (color lines [5], magnetism lines [6]), a camera [7], and a global positioning system (GPS). Herein, GPS is often used in the outdoor environment, which could be applied indoors via a beacon. Contour tracking is difficult in complex terrain, mapping challenges, and limited to topography. Magnetism lines have some limitations as environmental interference, installation and maintenance, and limited range. The camera has disadvantages such as environmental conditions and sensitivity to changes. However, the application in the factory can be considered to use the above methods. By using the feature of mapping and workspace, the contour line combined with the camera is of interest to many scientists because of the low cost and is easily applied in many places.

In this paper, we propose an algorithm for mobile robot navigation based on tracking lines and cameras using vision processing. The rest of this paper is organized as follows: The literature review is presented in Sect. 2. Next, the mobile robot and navigation problem is shown in Sect. 3. Then, the control algorithm development is developed in detail in Sect. 4. The experimental and results are presented in Sect. 5 to demonstrate performances of control algorithm proposal. Finally, the conclusion is given in Sect. 6.

## 2 Literature Review

The control algorithm is one important part of deciding the performance of a mobile robot. Some algorithms were developed such as ON-OFF [8], proportional integral derivative (PID) controller [9], and Fuzzy-PID [10]. These algorithms use the infrared ray (IR) as the feedback sensor that could detect whether the robot is in line or not [3–5]. However, IR is often affected by environments and then gets feedback not correct. It will impact the robot's performance. To overcome this limitation, the camera is used instead of IR because of reduced noise and more information provided by the camera [11]. Furthermore, control ON/OFF has poor precision, limited flexibility, and many stability issues. Therefore, it is not offered for use in mobile robot control.

Similar to the IR sensor, the camera computed the error position of the robot in comparison to the line and sent it to the PID control algorithm which will control the speed of the wheel for the mobile robot. Many methods of computed error were performed such as [7, 12, 13]. Herein, Gomes [12] used a camera with high resolutions and modular open-robot simulation engine software. Samet Oguten [7] could perform the experiments with an STM32 microchip to develop a program controlling the mobile robot's two wheels. Li-Hong Juang uses the tangent function to reduce the error between the line and center of view [13] for a human robot. The above methods showed the advantages and disadvantages of special environments in both simulations and experiments. However, the system configurations are more complex and combine the complex equations to compute error control. Sometimes, the equations could exist deadband.

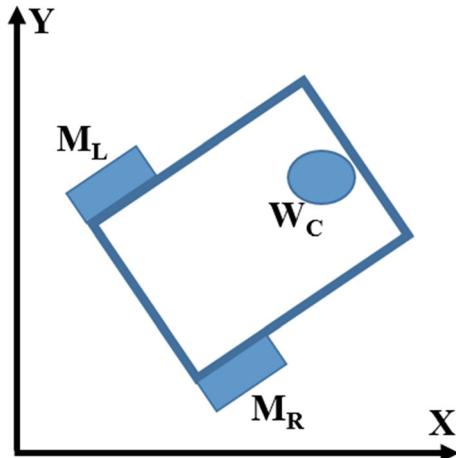
Furthermore, the camera for tracking lines has several limitations that can affect its effectiveness and reliability in certain scenarios such as lighting variations, reflective surfaces, weather conditions, line width and color, damaged or faded lines, real-time processing, computational requirements, etc. Therefore, using the camera to navigate mobile robots in the factory is one challenge with researchers. We focus on the control algorithm lightweight which can run in the embedded system and still has high accuracy compared to IR sensors in the same environment and mapping line. Besides, it is a premise for applying more advanced algorithms in the future.

In this paper, we propose the moment of the image to compute the error position. The contour of the captured image by a camera is detected and sent to the moment algorithm to compute the error based on the central line area and image, which is the input of the PID controller, and its output is used to handle the speed of the motor of the mobile robot. All experiments are performed in a small mobile robot integrated with a Rasberry board for vision processing and a beacon system for position references. To prove the proposal control algorithm, the results are also compared to the IR sensor [14].

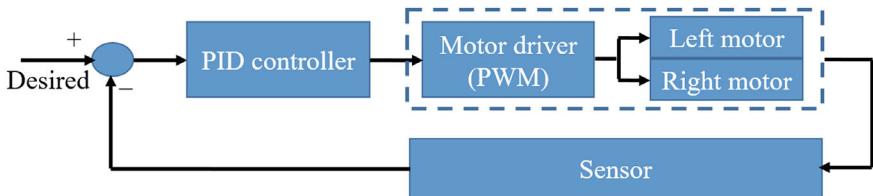
### 3 Mobile Robot and Navigation Problem

In this paper, a mobile robot (MB) with three wheels as in Fig. 1 is used for evaluating control algorithms and proposals. Herein, MB is handled by two-wheel  $M_L$  and  $M_R$  rears. And  $W_c$  wheel is slips to ensure balance for the MB.

To navigate the MB, the PID controller is designed to handle  $M_L$  and  $M_R$  via pulse width modulation (PWM) as in Fig. 2. The sensor feedback is selected for each criteria control. This research focused on navigation based on the black line. The camera is used for detecting the line in our proposal and the IR sensor is one comparison.



**Fig. 1** Mobile robot structure



**Fig. 2** General control algorithm for the mobile robot

## 4 Control Algorithm Development

This section presents the control algorithm design based on vision processing. A frame is captured as in Fig. 3 with two colors black and white. And the black color is to navigate the MB movement in the mapping.

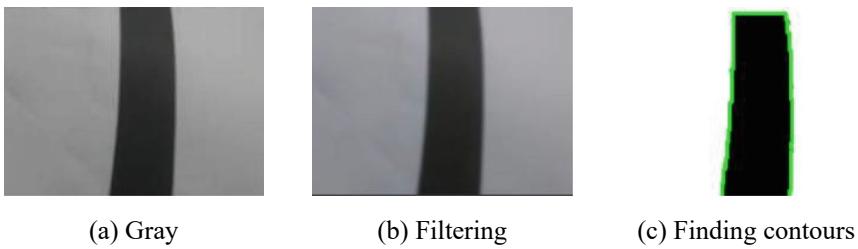
The image has to be processed to get the meaning and information by some steps such as converting to a gray image in Fig. 4a, filtering in Fig. 4b, and finding contours in Fig. 4c. Each step is performed by using the open-source code of OpenCV:

```
imggray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
ret, thresh = cv2.threshold(imggray, 100, 255, cv2.THRESH_BINARY_INV)
contours, _ = cv2.findContours(thresh, 1, cv2.CHAIN_APPROX_NONE)
```

Next, the moment of the image algorithm is to recognize the pattern, object detection, and robot vision [15], which is used to compute the centroid of the black line area in the captured frame from the camera. The moment could be expressed as



**Fig. 3** Captured image from camera



**Fig. 4** Pre-processing image

$$M_{i,j} = \sum_{x=1}^m \sum_{y=1}^n x^i y^j I(x, y) \quad (1)$$

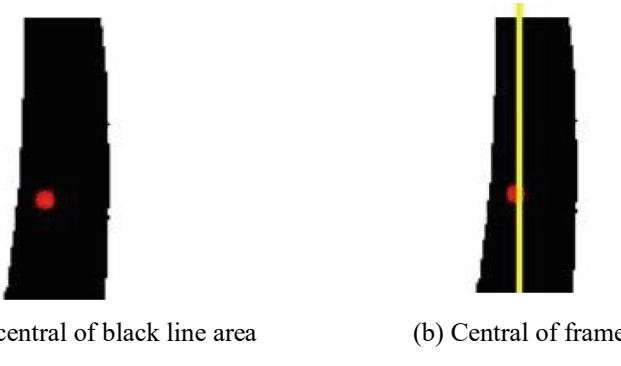
where  $M_{i,j}$  is moment with order  $(i, j)$ ,  $m$  and  $n$  are the size of height and width of the image.  $I(x, y)$  is the pixel value defined as 0 and 1. The centroid of the black line area is computed as

$$X_C = \frac{M_{1,0}}{M_{0,0}} \quad (2)$$

$$Y_C = \frac{M_{0,1}}{M_{0,0}} \quad (3)$$

An example of a definition image and computed centroid of the black line area is as in Fig. 5a, the red point presents the central line.

For this situation, the center of the black line (CP) should be controlled to the center line (CL) of the image by width size. The distance between CP to CL should be too small which means the MR is moving the correct way as in Fig. 5b. Therefore,



**Fig. 5** Definition image and computed centroid of the black line area

the error of control is defined as

$$e = X_{CF} - X_C \quad (4)$$

where  $X_{CF}$  is the center line in the  $X$ -axis. The error is only considered based on the  $X$ -axis.

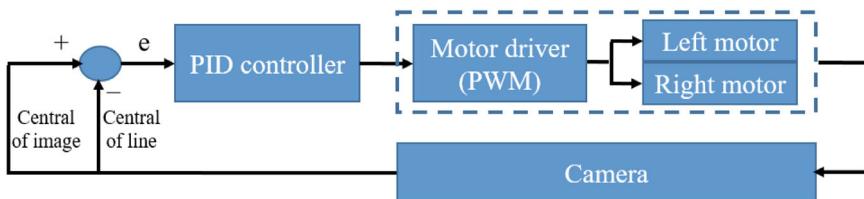
Using the error measurement, the PID [16, 17] controller as in Fig. 6 is used to handle the motor of MB including the left motor and right motor, which could be written in Eqs. (5) in continuous time and (6) in discrete time.

$$U_c(t) = K_p e(t) + K_i \int_0^T e(t) dt + K_d \frac{de(t)}{dt} \quad (5)$$

$$u(kT) = K_p e(kT) + K_i \sum_{k=1}^n e(kT) + K_d \frac{e(kT) - e(kT - T)}{T} \quad (6)$$

where  $U_c(t)$ ,  $u(kT)$  are the control output which is set to handle the system and the  $K_p$ ,  $K_i$ , and  $K_d$  are the gain of the PID controller. And  $T$  is the sampling time.

After getting the control output, the PWM for the left and right motor of the mobile robot is computed as [1, 4, 5]



**Fig. 6** PID controller design for MR navigation based on camera

$$S_L = S_S + C_O \quad (7)$$

$$S_R = S_S - C_O \quad (8)$$

where  $S_S$ ,  $S_L$ ,  $S_R$ ,  $S_S$ , and  $C_O$  are the PWM value default, PWM value setting for left motor, right motor, and control output from PID, respectively.

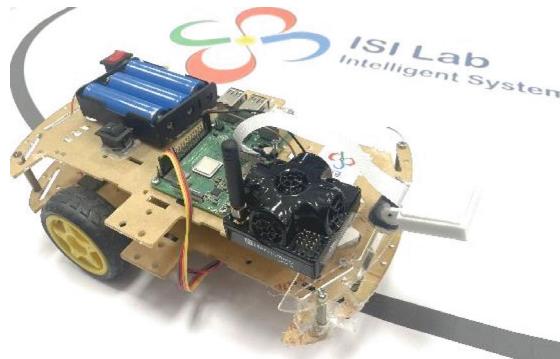
In this paper, we develop the PID controller using the Python code in the Raspberry board based on an ARM microchip. Therefore, PID in the form of discrete-time is selected to get output control for the motor of MB.

## 5 Experimental Results

The experiment is performed in a platform including a mobile robot with three wheels, a Raspberry Pi 3 Model B+ board, a camera 5 megapixel for Raspberry Pi Camera Module V2, a module L298H, the battery Lithium 16850 3.7 V, and a beacon MARVELMIND [18] as in Fig. 7a. Herein, the Raspberry Pi board is used to collect data from the camera, compute the central point of the black line, and the error control based on OpenCV and Python programming. The camera is configured with frame rates of 90 fps and a size of  $180 \times 128$ . The beacon MARVELMIND is to get the location of the mobile robot for comparison tracking of the mobile robot. A mapping is drawn and printed on the ground as in Fig. 7b. Testing the mobile robot using the mapping in Fig. 7b, the gain for the PID controller is set as  $K_p = 0.2$ ,  $K_i = 0.001$ , and  $K_d = 1.0$ . The sample time is 0.01 s.

The result in Fig. 8 shows the performance of the mobile robot tracking the black line. It is presented that the mobile robot could track all lines in the map. However, it is not a high-accuracy pair to the desired map because the beacon MARVELMIND has an accuracy within  $\pm 2$  cm. The performance is displayed clearly in Fig. 9 which is the error between the central line and the image. The first time, the mobile robot has to have small error control because of the initial position. The error is less than 0.02 and 0.15 cm for generated lines and high-angle sharp corners.

One of the key strengths of our work lies in the comparison between our vision-based navigation algorithm and an IR sensor-based approach. Through identical mapping conditions, start points, and sampling times for the PID controller, we conducted a comprehensive evaluation of both methods. Our results, depicted in Fig. 10, illustrate that our proposed vision algorithm outperforms the IR algorithm, particularly evident in sections of the mapping such as  $X = 0.3$  m to  $X = 0.6$  m and  $Y = 0.1$  m to  $Y = 0.23$  m. We attribute this superior performance to the susceptibility of IR sensors to environmental factors like light intensity and background reflection, which can lead to erroneous feedback values and larger error controls. In contrast, our vision-based method demonstrates consistently higher performance, enabling the mobile robot to track the black line with minimal errors. These findings highlight the effectiveness and robustness of our control algorithm, highlighting its potential

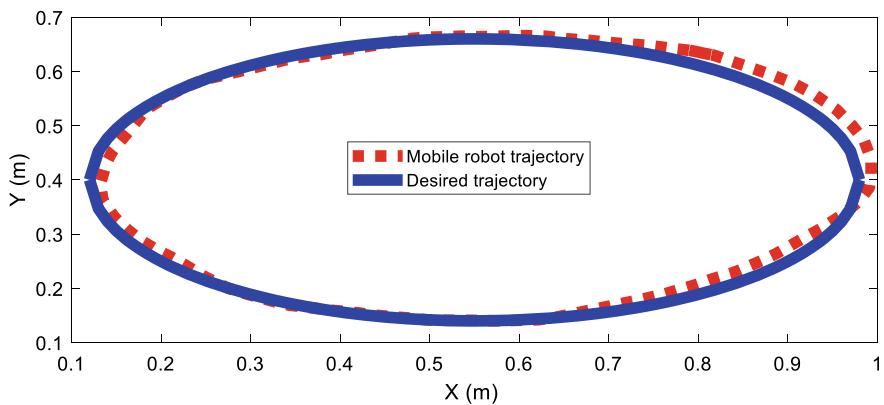


(a) Mobile robot integrated with other devices

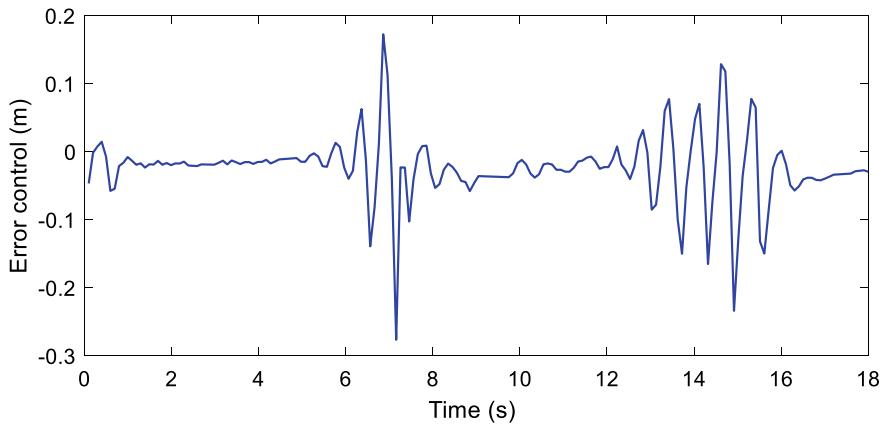


(b) Mapping line for navigating

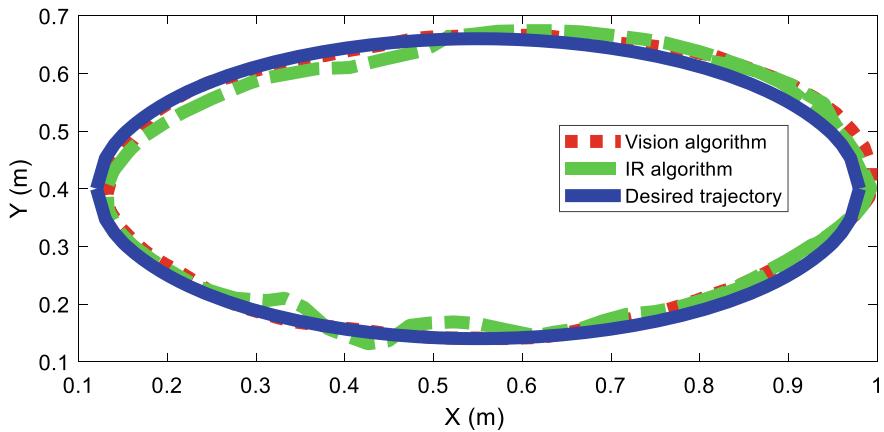
**Fig. 7** Configuration system in experimental



**Fig. 8** Performance of mobile robot tracking the black line



**Fig. 9** Error for the PID controller



**Fig. 10** Comparison tracking line based on vision and IR algorithms

applicability in real-world mobile robot navigation scenarios and broader robotics domains.

## 6 Conclusion

This paper introduces a new navigation line algorithm designed for a mobile robot equipped with three wheels, utilizing the moment of image method in conjunction with a PID controller algorithm. By capturing images through a camera interfaced with a Raspberry Pi B+, the algorithm computes the error control necessary for PID

input, subsequently directing the output to the two left and right motors. The experimental results demonstrate the efficacy of the proposed approach, showcasing the mobile robot's ability to accurately track mapping tasks, as validated by comparison with a beacon MARVELMIND device. These findings underscore the robustness and effectiveness of the control algorithms and processing techniques employed, thus offering promising advancements in the realm of mobile robot navigation.

## References

1. Bernardo R, Sousa JMC, Gonçalves PJS (2022) Survey on robotic systems for internal logistics. *J Manuf Syst* 65:339–350. <https://doi.org/10.1101/j.jmsy.2022.09.014>
2. Pang R, Cao J, Wang Y, Qi Y, Sun L (2022) Transportation robot based on multi-sensor fusion and machine vision. In: 2022 34th Chinese control and decision conference (CCDC), pp 6282–6287. <https://doi.org/10.1109/CCDC55256.2022.10033531>
3. Moshayedi AJ, Li J, Liao L (2021) Simulation study and PID tune of automated guided vehicles (AGV). In: 2021 IEEE international conference on computational intelligence and virtual environments for measurement systems and applications (CIVEMSA), pp 1–7. <https://doi.org/10.1109/CIVEMSA52099.2021.9493679>
4. Abdul Kader M, Islam MZ, Al Rafi J, Rasedul Islam M, Sharif Hossain F (2018) Line following autonomous office assistant robot with PID algorithm. In: 2018 International conference on innovations in science, engineering and technology (ICISSET), pp 109–114. <https://doi.org/10.1109/ICISSET.2018.8745606>
5. Balaji V, Balaji M, Chandrasekaran M, Khan MKAA, Elamvazuthi I (2015) Optimization of PID control for high-speed line tracking robots. *Procedia Comput Sci* 76:147–154. <https://doi.org/10.1016/j.procs.2015.12.329>
6. Almeida D, Pedrosa E, Curado F (2021) Magnetic mapping for robot navigation in indoor environments. In: 2021 International conference on indoor positioning and indoor navigation (IPIN), pp 1–8. <https://doi.org/10.1109/IPIN51156.2021.9662528>
7. Ogutem S, Kabas B (2021) PID controller optimization for low-cost line follower robots. arXiv preprint [arXiv:2111.04149](https://arxiv.org/abs/2111.04149)
8. Lee GH, Jung S (2013) Line tracking control of a two-wheeled mobile robot using visual feedback. *Int J Adv Rob Syst* 10(3):177. <https://doi.org/10.5772/53729>
9. Xu L, Du J, Song B, Cao M (2022) A combined backstepping and fractional-order PID controller to trajectory tracking of mobile robots. *Syst Sci Control Eng* 10(1):134–141. <https://doi.org/10.1080/21642583.2022.2047125>
10. Tiep DK, Lee K, Im D-Y, Kwak B, Ryoo Y-J (2018) Design of fuzzy-PID controller for path tracking of mobile robot with differential drive. *Int J Fuzzy Log Intell Syst* 18(3):220–228. <https://doi.org/10.5391/IJFIS.2018.18.3.220>
11. Kondákor A, Törsvári Z, Nagy Á, Vajk I (2018) A line tracking algorithm based on image processing. In: 2018 IEEE 12th international symposium on applied computational intelligence and informatics (SACI), pp 000039–000044. <https://doi.org/10.1109/SACI.2018.8440975>
12. Gomes MV, Bássora LA, Morandin O, Vivaldini KCT (2016) PID control applied on a line-follower AGV using a RGB camera. In: 2016 IEEE 19th international conference on intelligent transportation systems (ITSC), pp 194–198. <https://doi.org/10.1109/ITSC.2016.7795553>
13. Juang L-H, Zhang J-S (2020) Robust visual line-following navigation system for humanoid robots. *Artif Intell Rev* 53(1):653–670. <https://doi.org/10.1007/s10462-018-9672-9>
14. Abideen ZU, Anwar MB, Tariq H (2018) Dual purpose Cartesian infrared sensor array based PID controlled line follower robot for medical applications. In: 2018 International conference on electrical engineering (ICEE), pp 1–7. <https://doi.org/10.1109/ICEE.2018.8566871>

15. Rocha L, Velho L, Carvalho PCP (2002) Image moments-based structuring and tracking of objects. In: Proceedings. XV Brazilian symposium on computer graphics and image processing, pp 99–105. <https://doi.org/10.1109/SIBGRA.2002.1167130>
16. Zishan F, Akbari E, Montoya OD, Giral-Ramírez DA, Molina-Cabrera A (2022) Efficient PID control design for frequency regulation in an independent microgrid based on the hybrid PSO-GSA algorithm. *Electronics* 11(23). <https://doi.org/10.3390/electronics11233886>
17. Manita M, Boussaid B, Abdelkrim MN (2022) Wheeled mobile robot control approaches: comparative analysis. In: 2022 19th international multi-conference on systems, signals and devices (SSD), pp 1913–1918. <https://doi.org/10.1109/SSD54932.2022.9955504>
18. Amsters R, Demeester E, Stevens N, Lauwers Q, Slaets P (2019) Evaluation of low-cost/high-accuracy indoor positioning systems. In: ALLSENSORS 2019: the fourth international conference on advances in sensors, actuators, metering and sensing

# A Systematic Literature Review on Lung Cancer with Ensemble Learning



Fahum Nufikha Jahan, Shakik Mahmud , and Md Kamrul Siam

**Abstract** This systematic review seeks to establish the application of ensemble learning approaches in lung cancer diagnosis with emphasis on 47 articles from 934 published between 2023 and 2024. The research data is collected from databases such as Science Direct, Proquest, EBSCOhost, and Google Scholar to understand the use of machine learning in medical diagnostics and enhance predictive and clinical outcomes. Our study, structured around the PRISMA framework, addresses three core research questions: First, the paper will outline the types and the frequency of Ensemble Learning techniques employed in the last lung cancer studies, the metrics applied in the evaluation of the models, and the type of data applied in these studies. These observations point to a general trend of employing more combinations of various algorithms to increase the predictive power and a particular interest in deep learning ensembles. These approaches can help enhance the diagnostic sensitivity and specificity crucial in the early detection of lung cancer. The findings of the analysis are based on the current practices of machine learning in the healthcare sector, and it maintains that ensemble learning may be a promising approach to cancer treatment. Thus, this review not only discloses the current approach and challenges but also depicts future advancements in lung cancer diagnosis.

**Keywords** Lung cancer · Ensemble learning · Systematic literature review

---

F. N. Jahan  
Western Illinois University, Macomb, IL, USA  
e-mail: [fn-jahan@wiu.edu](mailto:fn-jahan@wiu.edu)

S. Mahmud ()  
Japan-Bangladesh Robotics and Advanced Technology Research Center, Dhaka, Bangladesh  
e-mail: [shakikmahmud@gmail.com](mailto:shakikmahmud@gmail.com)

Md Kamrul Siam  
New York Institute of Technology, New York, NY, USA  
e-mail: [ksiam01@nyit.edu](mailto:ksiam01@nyit.edu)

## 1 Introduction

Lung cancer is still one of the leading causes of death from cancer, and thus early diagnosis and treatment are crucial for better prognosis of the disease [1]. New developments in the field of machine learning especially ensemble learning have improved the precision and speed of lung cancer prediction [2, 3]. This systematic review aims to identify the most effective ensemble learning methods, metrics, and data for lung cancer prediction models as of 2024 and 2023 based on Science Direct, Nature, and Springer articles. In this paper, we follow PRISMA guidelines to keep the analysis as systematic and as transparent as possible [4, 5]. While the application of machine learning in enhancing lung cancer predictions has been established, the effects of ensemble learning methods have been less examined [6]. This review discusses the limitations and reliability problems of the existing models by focusing on the efficiency of ensemble methods in providing better prediction results [7]. There was no agreement on the best evaluation metrics and data types; therefore, more specific research is needed to improve these areas [8, 9]. In conclusion, this work emphasizes the need for further investigation and inter-disciplinary cooperation to improve diagnostic tools for lung cancer and, thus, improve patients' prognosis [10].

## 2 Working Method

This systematic review was meticulously designed to investigate the application of ensemble learning techniques in lung cancer prediction. Our methodology adhered to a structured approach, ensuring comprehensive coverage and rigorous analysis of relevant literature.

### 2.1 *Search Keyword Strategy*

The primary search keywords were “Lung Cancer” and “Ensemble Learning.” These terms were chosen for their direct relevance to our research objectives, enabling us to filter and retrieve articles that specifically address the intersection of lung cancer diagnosis and prediction with ensemble learning methods in machine learning. These keywords were used in various combinations and permutations across different databases to ensure a thorough search.

## 2.2 *Exclusion Criteria*

Only two levels to maintain the focus and quality of our review, we established specific exclusion criteria:

- **EC1:** Only studies published between 2023 and 2024 were considered to ensure the relevance and timeliness of the data.
- **EC2:** We excluded reviews, survey publications, and non-original research articles to focus solely on primary research.
- **EC3:** Papers not written in English were excluded.
- **EC4:** Studies that did not directly relate to machine learning applications in lung cancer prediction were omitted.
- **EC5:** We prioritized open-access articles for wider accessibility and verification of content.

## 2.3 *Research Questions*

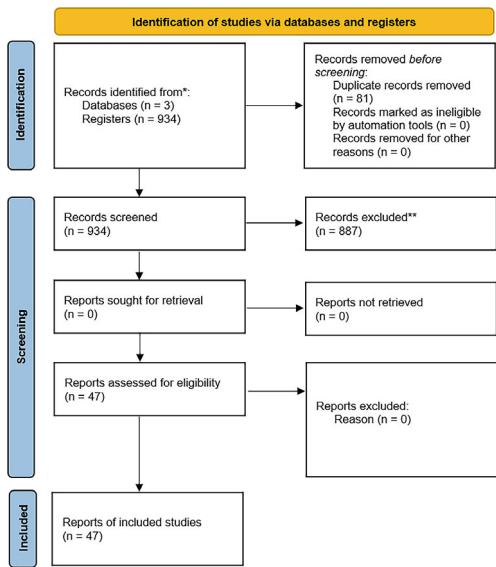
Our review was guided by the following research questions:

- **RQ1:** Which ensemble learning techniques are most frequently employed in recent lung cancer prediction studies?
- **RQ2:** What evaluation metrics are commonly used in these studies?
- **RQ3:** What types of data are most often utilized in lung cancer prediction models using ensemble learning?
- **RQ4:** What was the dataset size they used?

## 2.4 *PRISMA Framework*

Our review process was aligned with the PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) guidelines. This included a four-phase flow of identification, screening, eligibility, and inclusion of articles. Each phase was meticulously documented to ensure transparency and reproducibility of the review process (Fig. 1).

The PRISMA framework guided the selection of articles, data extraction, and the synthesis of findings.

**Fig. 1** PRISMA framework

### 3 Result

The systematic review yielded insightful results regarding the use of ensemble learning techniques in lung cancer prediction, the types of data utilized, and the evaluation metrics employed. These findings are categorized based on the three research questions (Table 1).

#### 3.1 Ensemble Learning Techniques Employed

The analysis revealed a diverse range of ensemble learning techniques used in the recent studies on lung cancer prediction. The frequency of their usage is as follows (Table 2).

This comprehensive review found that many machine learning algorithms have been employed to predict lung cancer, demonstrating its complexity and specificity. Convolutional Neural Networks (CNNs) are widely used in image processing, especially in Deep CNN, 2D CNN, and Dense CNN designs with 121 layers [14, 16, 19]. ResNet50 and ResNet101, known for their deep learning architecture that addresses the vanishing gradient problem, were also popular. The XGBoost Classifier, which works well with tabular data, was also used. These examples demonstrate machine learning's adaptability in lung cancer diagnosis (Figs. 2 and 3).

**Table 1** Distribution of papers based on the databases

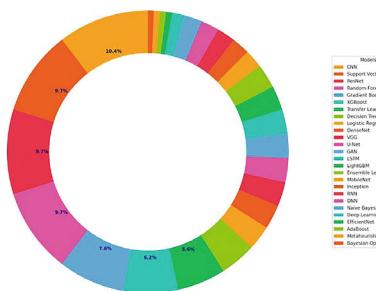
Publisher Name	Retrieved papers	Refs.	Publisher Name	Retrieved papers	Refs.
Nature	11	[12, 14, 19, 32, 47–53]	International Journal of Advanced Computer Science and Applications (IJACSA)	1	[11]
MDPI	11	[13, 15, 20, 23, 31, 33, 35, 38, 43, 44, 46]	International Journal of Advanced Technology and Engineering Exploration	1	[16]
Springer	5	[18, 21, 26, 41, 57]	IOP	1	[17]
Public Library of Science San Francisco, CA USA	4	[24, 25, 34, 40, 42]	Universitas Ahmad Dahlan	1	[22]
Elsevier	3	[27, 37, 54]	Wiley Online Library	1	[28]
Science and Information (SAI) Organization Limited	3	[29, 30, 36]	Taylor & Francis	1	[39]
PeerJ.	1	[56]	Plos One	1	[55]
Emerald	1	[45]	–	–	–

**Table 2** Details of model and algorithms

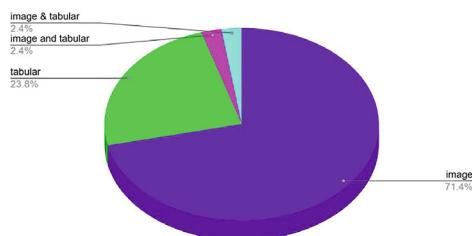
Model/Algorithm	Frequency	Model/Algorithm	No.	Model/Algorithm	No.
CNN	15	XGBoost	9	SVM	14
Transfer learning	8	ResNet	14	Decision tree	6
Random forest	14	Logistic regression	4	Gradient boosting	11
DenseNet	4	VGG	4	GAN	4
U-Net	4	LSTM	4	–	–

**Fig. 2** Uses frequency of algorithms and models

Pie Chart of Model/Algorithm Usage Frequency  
(Percentages shown for >5%)



**Fig. 3** Types of dataset



**Table 3** Datasets size

Range	Count	Range	Count
0–5,000	31	10,001–20,000	4
<30,000	2	5,001–10,000	4

### 3.2 Type of Datasets

The studies reviewed employed various data types for lung cancer prediction, with the following frequency.

The article observed and categorized 43 datasets by size, with most falling within lower ranges. Four studies in the study did not reveal their dataset sizes, which could affect the systematic review's quality and reliability. The absence of data transparency makes it difficult to evaluate the robustness of ensemble learning applications (Table 3).

### 3.3 Evaluation Metrics Employed

A range of evaluation metrics were used to assess the performance of ensemble learning models in lung cancer prediction. Their usage frequency is as follows:

**Table 4** Evaluation metrics used for different models or algorithms

Name	Count (%)	Name	Count (%)
AUC-ROC	88	G-Mean	11
Accuracy	71	Balanced accuracy	9
Confusion matrix	69	F1 score	6
Precision	44	Mean absolute error	4
Logarithmic loss	37	Kappa score	1

The wide acceptance of AUC-ROC and Accuracy as evaluation metrics suggests an emphasis on both prediction accuracy and class distinction. However, using a variety of measures including Confusion Matrix, Precision, and Logarithmic Loss shows how complicated medical diagnostic predictive model evaluation is (Table 4).

## 4 Conclusion

This systematic review provides a solid foundation for the use of ensemble learning approaches that combine a number of different machine learning algorithms to improve lung cancer prediction and diagnostic accuracy, which is critical for clinical application. Deep learning ensembles which are capable of handling the intricate oncological data were most efficient in high-dimensional imaging data analysis. Nevertheless, we also report some limitations of the study, such as the difficulty of extending ensemble models, the problem of handling huge and heterogeneous data, and the issue of developing low-cost solutions in the clinical domain. Also, harmonization of data and evaluation methods is important in order to make comparable results and conclusions. However, ensemble learning may hold the key to overcoming these difficulties in the diagnosis of lung cancer, and therefore further studies are needed to develop these techniques for clinical use. Therefore, this study recommends that there should be continuous improvement and research in this crucial part of the healthcare field.

## References

1. Ghita M, Billiet C, Copot D, Verellen D, Ionescu CM (2023) Parameterisation of respiratory impedance in lung cancer patients from forced oscillation lung function test. IEEE Trans Biomed Eng 70(5):1587–1598. <https://doi.org/10.1109/TBME.2022.3222942>
2. Poorani S, Parvathavarthini S, Kalaiselvi S, Aarthi N, Agalya S, Malathy NR (2023) Ensemble CNN model for lung cancer classification. In: 2023 5th international conference on inventive research in computing applications (ICIRCA), Coimbatore, India, 2023, pp 822–827. <https://doi.org/10.1109/ICIRCA57980.2023.10220724>

3. Binson VA, Subramoniam M, Ragesh GK, Kumar A (2021) Early detection of lung cancer through breath analysis using adaboost ensemble learning method. In: 2021 2nd international conference on advances in computing, communication, embedded and secure systems (ACCESS), Ernakulam, India, 2021, pp 183–187. <https://doi.org/10.1109/ACCESS51619.2021.9563337>
4. Omar LT, Hussein JM, Omer LF, Qadir AM, Ghareb MI (2023) Lung and colon cancer detection using weighted average ensemble transfer learning. In: 11th International symposium on digital forensics and security (ISDFS). Chattanooga, TN, USA, 2023, pp 1–7. <https://doi.org/10.1109/ISDFS58141.2023.10131836>
5. Platini H, Ferdinand E, Kohar K, Prayogo SA, Amira S, Komariah M, Maulana S (2022) Neutrophil-to-lymphocyte ratio and platelet-to-lymphocyte ratio as prognostic markers for advanced non-small-cell lung cancer treated with immunotherapy: a systematic review and meta-analysis. *Medicina* 58
6. Alsinglawi BS, Alshari OM, Alorjani MS, Mubin O, Alnajjar FS, Novoa M, Darwish OM (2022) An explainable machine learning framework for lung cancer hospital length of stay prediction. *Sci Rep* 12
7. Mamun M, Farjana A, Mamun MA, Ahammed MS (2022) Lung cancer prediction model using ensemble learning techniques and a systematic review analysis. In: 2022 IEEE world AI IoT congress (AIIoT), pp 187–193
8. Krishna RS (2023) Machine learning approaches in early lung cancer prediction: a comprehensive review. *Int J Sci Res Eng Manag*
9. Althubiti SA, Paul S, Mohanty R, Mohanty SN, Alenezi FS, Polat K (2022) Ensemble learning framework with GLCM texture extraction for early detection of lung cancer on CT images. *Comput Math Methods Med* 2022
10. Mostafa FA, Elrefaei LA, Fouad MM, Hossam A (2022) A survey on AI techniques for thoracic diseases diagnosis using medical images. *Diagnostics* 12
11. Sumellika T, Prasad RS (2024) A combined ensemble model (CEM) for a liver cancer detection system. *Int J Adv Comput Sci Appl* 15(2)
12. Dervishi A (2024) A multimodal stacked ensemble model for cardiac output prediction utilizing cardiorespiratory interactions during general anesthesia. *Sci Rep* 14(1):7478
13. Walid MAA, Mollick S, Shill PC, Baowaly MK, Islam MR, Ahamad MM, Othman MA, Samad MA (2023) Adapted deep ensemble learning-based voting classifier for osteosarcoma cancer classification. *Diagnostics* 13(19):3155
14. Zhou J, Hu B, Feng W, Zhang Z, Fu X, Shao H, Wang H, Jin L, Ai S, Ji Y (2023) An ensemble deep learning model for risk stratification of invasive lung adenocarcinoma using thin-slice CT. *NPJ Digit Med* 6(1):119
15. Mustafa E, Jadoon EK, Khaliq-uz-Zaman S, Humayun MA, Maray M (2023) An ensembled framework for human breast cancer survivability prediction using deep learning. *Diagnostics* 13(10):1688
16. Jha M, Gupta R, Saxena R (2024) Convolutional neural network based detection of lung adenocarcinoma by amalgamating hybrid features. *Int J Adv Technol Eng Explor* 11. <http://dx.doi.org/10.19101/IJATEE.2023.10102196>
17. Ikechukwu AV, Murali S (2023) CX-Net: an efficient ensemble semantic deep neural network for ROI identification from chest-x-ray images for COPD diagnosis. *Mach Learn Sci Technol* 4(2):025021
18. Jia L, Ren X, Wu W, Zhao J, Qiang Y, Yang Q (2024) DCCAFN: deep convolution cascade attention fusion network based on imaging genomics for prediction survival analysis of lung cancer. *Complex Intell Syst* 10(1):1115–1130
19. Shah AA, Malik HAM, Muhammad A, Alourani A, Butt ZA (2023) Deep learning ensemble 2D CNN approach towards the detection of lung cancer. *Sci Rep* 13(1):2987
20. Ms K, Rajaguru H, Nair AR (2024) Enhancement of classifier performance with Adam and RanAdam hyper-parameter tuning for lung cancer detection from microarray data-in pursuit of precision. *Bioengineering* 11(4):314

21. Gong J, Wang T, Wang Z, Chu X, Hu T, Li M, Peng W, Feng F, Tong T, Gu Y (2024) Enhancing brain metastasis prediction in non-small cell lung cancer: a deep learning-based segmentation and CT radiomics-based ensemble learning model. *Cancer Imag* 24(1):1
22. Menshawy L, Eid AH, Abdel-Kader RF (2023) Ensemble deep models for COVID-19 pandemic classification using chest X-ray images via different fusion techniques. *Int J Adv Intell Inf* 9(1):51–65
23. Subashchandrabose U, John R, Anbazhagu UV, Venkatesan VK, Thyluru Ramakrishna M (2023) Ensemble Federated learning approach for diagnostics of multi-order lung cancer. *Diagnostics* 13(19):3053
24. Kaleem S, Sohail A, Tariq MU, Babar M, Qureshi B (2023) Ensemble learning for multi-class COVID-19 detection from big data. *PLoS ONE* 18(10):e0292587
25. Lee S, Lee S-Y, Jung J-Y, Nam Y, Jeon HJ, Jung C-K, Shin S-H, Chung Y-G (2023) Ensemble learning-based radiomics with multi-sequence magnetic resonance imaging for benign and malignant soft tissue tumor differentiation. *PLoS ONE* 18(5):e0286417
26. Zhang X, Zhang G, Qiu X, Yin J, Tan W, Yin X, Yang H, Wang H, Zhang Y (2024) Exploring non-invasive precision treatment in non-small cell lung cancer patients through deep learning radiomics across imaging features and molecular phenotypes. *Biomark Res* 12(1):12
27. Shi Y, Zhang X, Yang Y, Cai T, Peng C, Wu L, Zhou L, Han J, Ma M, Zhu W et al (2023) D3CARP: a comprehensive platform with multiple-conformation based docking, lig and similarity search and deep learning approaches for target prediction and virtual screening. *Comput Biol Med* 164:107283
28. Yang J, Lei X, Zhang F (2024) Identification of circRNA-disease associations via multi model fusion and ensemble learning. *J Cell Mol Med* 28(7):e18180
29. Luo S (2023) Lung cancer classification using reinforcement learning-based ensemble learning. *Int J Adv Comput Sci Appl* 14(8)
30. Tiwari A, Hannan SA, Pinnamaneni R, Al-Ansari ARM, El-Ebary YAB, Prema S, Manikandan R, Vidalón JLJ (2023) Optimized ensemble of hybrid RNN-GAN models for accurate and automated lung tumour detection from CT images. *Int J Adv Comput Sci Appl* 14(7)
31. Abbasi EY, Deng Z, Maggi AH, Ali Q, Kumar K, Zubedi A (2023) Optimizing skin cancer survival prediction with ensemble techniques. *Bioengineering* 11(1):43
32. Tian L, Wu J, Song W, Hong Q, Liu D, Ye F, Gao F, Hu Y, Wu M, Lan Y et al (2024) Precise and automated lung cancer cell classification using deep neural network with multiscale features and model distillation. *Sci Rep* 14(1):10471. (Nature Publishing Group UK London)
33. Rajadurai S, Perumal K, Ijaz MF, Chowdhary CL (2024) PrecisionLymphoNet: advancing malignant lymphoma diagnosis via ensemble transfer learning with CNNs. *Diagnostics* 14(5):469
34. Lyu G, Nakayama M (2023) Prediction of respiratory failure risk in patients with pneumonia in the ICU using ensemble learning models. *Plos One* 18(9):e0291711
35. Tang FH, Fong YW, Yung SH, Wong CK, Tu CL, Chan MT (2023) Radiomics clinical AI model with probability weighted strategy for prognosis prediction in non-small cell lung cancer. *Biomedicines* 11(8):2093
36. Wang Y, Wang H, Dong E (2023) Recognition of lung nodules in computerized tomography lung images using a hybrid method with class imbalance reduction. *Int J Adv Comput Sci Appl* 14(5):1–8
37. Ding R, Yadav A, Rodriguez E, da Silva ACAL, Hsu W (2023) Tailoring pretext tasks to improve self-supervised learning in histopathologic subtype classification of lung adenocarcinomas. *Comput Biol Med* 166:107484
38. Li W, Zhang W (2024) UTAC-Net: a semantic segmentation model for computer-aided diagnosis for ischemic region based on nuclear medicine cerebral perfusion imaging. *Electronics* 13(8):1466
39. Ahmed S, Raza B, Hussain L, Aldweesh A, Omar A, Khan MS, Eldin ET, Nadim MA (2023) The deep learning resnet101 and ensemble xgboost algorithm with hyperparameters optimization accurately predict the lung cancer. *Appl Artif Intell* 37(1):2166222

40. Majumder S, Gautam N, Basu A, Sau A, Geem ZW, Sarkar R (2024) MENet: a Mitscherlich function based ensemble of CNN models to classify lung cancer using CT scans. *Plos One* 19(3):e0298527
41. Onozato Y, Iwata T, Uematsu Y, Shimizu D, Yamamoto T, Matsui Y, Ogawa K, Kuyama J, Sakairi Y, Kawakami E et al (2023) Predicting pathological highly invasive lung cancer from preoperative [18F] FDG PET/CT with multiple machine learning models. *Eur J Nucl Med Mol Imaging* 50(3):715–726
42. Huang T, Le D, Yuan L, Xu S, Peng X (2023) Machine learning for prediction of in hospital mortality in lung cancer patients admitted to intensive care unit. *PLoS One* 18(1):e0280606
43. Ananthakrishnan B, Shaik A, Chakrabarti S, Shukla V, Paul D, Kavitha MS (2023) Smart diagnosis of adenocarcinoma using convolution neural networks and support vector machines. *Sustainability* 15(2):1399
44. Tummala S, Kadry S, Nadeem A, Rauf HT, Gul N (2023) An explainable classification method based on complex scaling in histopathology images for lung and colon cancer. *Diagnostics* 13(9):1594
45. Swain AK, Swetapadma A, Rout JK, Balabantaray BK (2024) A hybrid learning method for distinguishing lung adenocarcinoma and squamous cell carcinoma. *Data Technol Appl* 58(1):113–131
46. Zhang R, Shi K, Hohenforst-Schmidt W, Steppert C, Sziklavari Z, Schmidkonz C, Atzinger A, Hartmann A, Vieth M, Förster S (2023) Ability of 18F-FDG positron emission tomography radiomics and machine learning in predicting KRAS mutation status in Therapy-Naive lung adenocarcinoma. *Cancers* 15(14):3684
47. Kaide X, Chen D, Jin S, Yi X, Luo L (2023) Prediction of lung papillary adenocarcinoma-specific survival using ensemble machine learning models. *Sci Rep* 13(1):14827
48. Kinoshita F, Takenaka T, Yamashita T, Matsumoto K, Oku Y, Ono Y, Wakasus S, Haratake N, Tagawa T, Nakashima N, Mori M (2023) Development of artificial intelligence prognostic model for surgically resected non-small cell lung cancer. *Sci Rep* 13(1):15683
49. Selvam M, Chandrasekharan A, Sadanandan A, Anand VK, Murali A, Krishnamurthi G (2023) Radiomics as a non-invasive adjunct to Chest CT in distinguishing benign and malignant lung nodules. *Sci Rep* 13(1):19062
50. Ren X et al (2023) Weakly supervised label propagation algorithm classifies lung cancer imaging subtypes. *Sci Rep* 13(1):5167
51. Behrendt F, Bengs M, Bhattacharya D, Krüger J, Opfer R, Schlaefler A (2023) A systematic approach to deep learning-based nodule detection in chest radiographs. *Sci Rep* 13(1):10120
52. Souid A, Alsubaie N, Soufiane BO, Alqahtani MS, Abbas M, Jambi LK, Sakli H (2023) Improving diagnosis accuracy with an intelligent image retrieval system for lung pathologies detection: a features extractor approach. *Sci Rep* 13(1):16619
53. Yacob F, Siarov J, Villiamsson K, Suvilehto JT, Sjöblom L, Kjellberg M, Neittaanmäki N (2023) Weakly supervised detection and classification of basal cell carcinoma using graph-transformer on whole slide images. *Sci Rep* 13(1):7555
54. Pradhan KS, Chawla P, Tiwari R (2023) HRDEL: high ranking deep ensemble learning based lung cancer diagnosis model. *Expert Syst Appl* 213:118956. <https://doi.org/10.1016/j.eswa.2022.118956>
55. Akl AA, Hosny KM, Fouad MM, Salah A (2023) A hybrid CNN and ensemble model for COVID-19 lung infection detection on chest CT scans. *PLoS One* 18(3):e0282608. <https://doi.org/10.1371/journal.pone.0282608>
56. Kim G, Park YM, Yoon HJ, Choi J (2023) A multi-kernel and multi-scale learning based deep ensemble model for predicting recurrence of non-small cell lung cancer. *PeerJ Comput Sci* 9:e1311. <https://doi.org/10.7717/peerj.cs.1311>
57. Ravi V, Acharya V, Alazab M (2023) A multichannel EfficientNet deep learning-based stacking ensemble approach for lung disease detection using chest X-ray images. *Clust Comput* 26(2):1181–1203. <https://doi.org/10.1007/s10586-022-03664-6>

# A Study on Privacy-Preserving Multiparty Computation Protocols



**Chinmaya Bikram Pattanaik, Munesh Chandra Trivedi, Ruchi Jain,  
and Mohan Lal Kolhe**

**Abstract** Protocols for multiparty computation (MPC) enable a set of participants to engage and estimate a joint function of their secret inputs while disclosing only the output. MPC has a plethora of possible uses, including threshold cryptography, private DNA comparisons, privacy-preserving bidding, and private machine learning. Because of this, MPC has been the subject of extensive academic inquiry ever since Yao established it in the 1980s. Especially the evolution of privacy-preserving machine learning and the use of homomorphic encryption has given a significant push toward the development of MPC. In this research, we will look at what MPC is, which issues it resolves, and how it is currently employed as a privacy-preserving model.

**Keywords** MPC · Homomorphic encryption · Yao Garble

## 1 Introduction

The concept of distributed computing takes into account the situation in which several separate, but interconnected, computer devices (or parties) want to do a cooperative computation of some kind. These apparatuses could be servers housing a distributed

---

C. B. Pattanaik (✉)

Department of Computer Science and Engineering, National Institute of Technology, Agartala  
799046, Tripura, India

e-mail: [chinmayabikram@gmail.com](mailto:chinmayabikram@gmail.com)

M. C. Trivedi

Department of Engineering and Technology, PSSCIVE, Bhopal 462013, Madhya Pradesh, India

R. Jain

Department of Computer Science, Lakshmi Narain College of Technology & Science, Bhopal,  
Madhya Pradesh, India

M. L. Kolhe

Smart Grid and Renewable Energy, University of Agder, Campus Kristiansand, Universitetsveien  
25, 4630 Kristiansand, Norway

e-mail: [mohan.l.kolhe@uia.no](mailto:mohan.l.kolhe@uia.no)

database system, for instance, and the computation to be done might be some sort of updation of the database. Enabling a collection of autonomous data owners who do not authorize any common third party or each other and to work together to jointly construct a process that involves all of their secret inputs is the aim of secure multi-party computation (MPC). *Correctness* and *privacy* are the two essential characteristics of any secure computation mechanism. According to the privacy requirement, no information shall be collected beyond what is strictly required. According to the correctness requirement, every party must obtain its accurate product. To successfully implement an MPC protocol to resolve a conflict between autonomous and mistrusting data owners, it is necessary to handle a variety of difficult issues that go outside the purview of MPC execution itself. Developing trust in the design that will carry out the protocol, figuring out what sensitive data could be deduced from the MPC's disclosed output, and helping decision-makers tasked with safeguarding susceptible data but lacking a piece of knowledge in technical cryptography to comprehend the safety ramifications of taking part in the MPC are a few examples of these issues.

MPC makes it possible for applications that protect privacy when several mutually suspicious data owners work together to compute a function. It is especially beneficial when resolving the problems based on Yao's millionaires' problem [1], secure auctions, and secure electronic voting [2]. Among other examples of secure MPC are privacy-preserving network security monitoring [3], private stable matching [4], privacy-preserving machine learning, and much more.

## 1.1 *Organization of the Manuscript*

We will look into some terminologies and basic features of MPC in Sect. 2 followed by an extensive coverage of fundamental protocols of MPC. We will dive inside some existing MPC architecture to have an understanding of the current working methodology. In Sects. 3 and 4 we deliver a comparative study of existing MPC protocols that are being used as privacy-preserving mechanisms in various platforms. Following that we provide our analysis of the protocol that should be considered for usage and the improvements on the existing work. We finalize the research by providing our own proposed MPC protocol followed by a conclusion.

## 1.2 *Author's Contribution*

In this article, we provide an extensive working methodology of MPC, its characteristic features, some existing variants of MPC, its usage in privacy-preserving, and an analysis of existing secure MPC that are being used in various privacy-preserving methods of data mining, machine learning, cloud computing, and much more. Then we provide a foreword about the type of protocol that can be suitable for further use and its characteristic features. We deliver a brief architecture of a secure MPC that

will be further elaborated in a different article. Our main contribution is the analysis that we provide that can be used as a reference for future work. The analysis not only reduces the overhead of going through different methodologies but also provides an efficient insight into its improvement.

## 2 Terminologies

We indicate the encryption as well as decryption method of a message  $m$  using key  $k$  as  $\text{Enc}_k(m)$  and  $\text{Dec}_k(m)$ . Protocol members might also be referred to as parties or players interchangeably and will usually designate them as  $P_1, P_2, \dots$ . We will denote the adversary by  $A$ . We will use the symbol  $\epsilon_r$  to indicate a sample taken at random from within a set of distributions. For instance, we use write  $v\epsilon_r B(x)$  to indicate that  $v$  is the randomized output that we get out of executing algorithm  $B$  on input  $x$ . The notion leads to *computational indistinguishability* when we solely take into account *non-uniform, polynomial-time algorithms*  $A$ . Whereas, *Statistically indistinguishable* algorithms are defined as those that take into account *all* algorithms, regardless of their computing complexity. *Computational security* refers to security from adversaries using non-uniform, polynomial-time techniques. *Information-theoretic security*, often understood as unconditional or statistical safety, refers to safeguarding information from random assailants.

Some basic protocols that prove to be useful while working with MPC are explained and the dimensions in which MPC protocol definitions can be satisfied are explained in the following subsections.

### 2.1 Fundamental Protocols

**Secret Sharing** [?] a fundamental component that forms the basis of numerous MPC techniques. A  $(t, n)$ -secret sharing method divides the hidden  $s$  among  $n$  parts or shares, ensuring that any one of the divided parts or shares does not expose any knowledge about it, where  $t$  is the number of threshold adversary. Secret sharing schemes might have varying security features, but we use the definition provided by Beimel et al. [5] as follows.

Let  $D$  be used as the domain for the sharing of secrets and  $D_1$  be referred to as the domain of shares. Consider  $\text{Shar} : D \rightarrow D_1^n$  be a randomized sharing algorithm, and  $\text{Recon} : D_1^k \rightarrow D$  be an algorithm for reconstruction of the share, for  $k$  be a value greater than  $t$ . Let  $Pr$  be the protocol under consideration, thus  $(t, n)$ -secret sharing system is a tandem of algorithms ( $\text{Shar}$ ,  $\text{Recon}$ ) that fulfills the following two major characteristics:

- *Correctness.* Consider  $(s_1, s_2, \dots, s_n) = \text{Shar}(s)$ . Then,

$$\Pr[\forall k \geq t, \text{Recon}(s_{i_1}, \dots, s_{i_k}) = s] = 1,$$

where, for all set of  $k$  greater than or equal to threshold adversary  $t$  should be able to construct  $s$  successfully using reconstruction algorithm.

- *Privacy.* In information theory, any cluster of shares smaller than  $t$  discloses nothing about the secret. According to the protocol, for any two secrets  $a, b \in D$  and any feasible shares of vector  $\mathbf{v} = (v_1, v_2, \dots, v_k)$ , such a way that  $k < t$ ,

$$\Pr[\mathbf{v} = \text{Shar}(a) |_k] = \Pr[\mathbf{v} = \text{Shar}(b) |_k]$$

considering that  $|_k$  is the estimation on the subspace of set of  $k$  components.

**Garbled Circuits** Nieminen and Schneider [6] Yao’s Garbled Circuits Protocol (GC) is a particularly eminent and acclaimed MPC approach. A GC allocates two arbitrary wire labels  $(w_0, w_1)$  to each circuit wire  $(w_i^0, w_i^1)$ . The dimensions of each label are defined by the computational security parameter  $k$ , which is often designated to 128 in practical implementations. Wire label  $w_i^0$  represents plaintext data 0, while wire label  $w_i^1$  equals plaintext data 1. The evaluator cannot determine the fundamental plaintext value since the wire labels appear randomized. A garbled table is constructed for every gate in the rotation by encrypting the output label for each of the  $2^2 = 4$  potential input combinations. To prevent information leakage, the garbled table’s values are randomly permuted. To maintain isolation in GCs, only the result label is capable of being decrypted from a garbled table when two input labels are combined.

**Oblivious Transfers** (OT) [7] The traditional explanation of “1-out-of-2” OT requires dual players: a sender  $S$  with two secrets  $(x_a, x_b)$  and a receiver  $R$  with a chosen bit  $b \in 0, 1$ . The OT technique allows  $R$  to access  $x_b$  without understanding about the “further” secret  $x_a$ . At that exact time,  $S$  learns nothing at all.

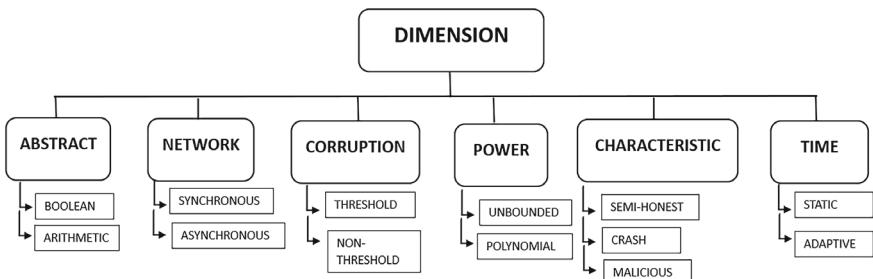
**Zero Knowledge Proof** (ZKP) [?] enables the demonstrator to persuade a verifier that they know  $x$  with  $C(x) = 1$  beyond disclosing any supplementary details regarding  $x$ . Considering,  $C$  is a publicly available predicate. Interactive zero-knowledge proofs need numerous rounds of interaction between the proving party and the verifying party. The graph isomorphism example demonstrates an interactive zero-knowledge proof. Non-interactive zero-knowledge Proofs (NIZK) do not involve interaction between the proving party and the verifying party. Instead, the proving party creates a single piece of evidence that the verified party can independently check. These are commonly employed in blockchain where contact is not feasible.

**Commitment** is an essential component of many cryptographic protocols. It allows the sender to execute to a present confidential value which can be revealed later. The receiver should not know the determined value until it is demonstrated by the sender (known as the hiding property), and the sender should not be capable of modifying the value after saving (known as the binding property).

## 2.2 Dimensions

MPC protocols have different features based on different use cases. These features, otherwise known as dimensions of MPC, prove to be useful while designing product-specific protocols. There are various dimensions in which an MPC protocol must prove its effectiveness, such as the degree of corrupt players considered, the type of corruption of the party, the corruption power, the scalability of data, the effectiveness of the protocol, the privacy-preserving ability, the circuits used, etc. The various dimensions of MPCs are shown in Fig. 1.

MPC protocols can be tailor-made for specific applications that use *Boolean* (AND, OR, etc.) or *Arithmetic* (Addition or Multiplication). The parties can use a model *synchronized* by a global clock (where wait time is known) or in an *asynchronized* way, i.e. without any clock where the wait time is unknown. Based on Corruption Capacity it can be a *threshold* model where the number of adversaries is definite or the *non-threshold* model where there can be any number of adversaries. Similarly based on Corruption Power it can be *polynomial* time/bounded computation or an *unbounded* computation. The characteristic feature of the adversary can be a *semi-honest* instance where the party pursues the protocol but attempts to know more additional than required, *crash* model where the party follows the semi-honest model plus it can stop the functioning of the model or it can be *malicious* model where the adversary sends out false messages. Based on the Time of corruption, in *static* time the hostile corrupts the players at the onset of a protocol, whereas in *adaptive* time the adversary corrupts the parties dynamically on the fly. Thus, an MPC protocol can be tailor-made specifically for various applications using the above-mentioned various dimensions.



**Fig. 1** Various dimensions of MPC protocols

### 3 Existing MPC Protocols

Yao's GC has proved to be the foundation stone of all major MPC protocols. Following the Yao GC was the Goldreich-Micali-Wigderson established, GMW protocol, a novel approach that performs generalization on more than two parties. Following it was the secure MPC protocol of Ben-Or, Goldwasser, and Wigderson (BGW). Later further more MPCs were developed. Although the list is not inexhaustive, some MPC protocols are extensively covered in Table 1. The dimensions of the MPCs differ from each other, for example, some are tested in a semi-honest adversary situation with the Boolean circuit, whereas some others are tested on Boolean as well as Arithmetic circuit, some are tested on malicious adversary, and so on.

### 4 Existing Privacy-Preserving MPC on Different Platforms

MPCs have a major advantage in that they can be tailor-made for every particular application, thus providing unique and problem-specific solutions. In Table 2 we have provided a list of MPCs that are used for privacy-preserving in various platforms such as cloud, data mining, and machine learning like FAMHE, NPMML, PPMCK, and others which are discussed elaborately in the following table. Although the list is not inexhaustive, it provides a deep insight into the use of MPC.

### 5 Scope of Improvement

In this section, we try to provide probable improvements for some of the above-mentioned protocols for privacy-preserving.

- MPSPDZ [16]: The subject matter covers common security models, including honest/dishonest majority and semi-honest/malicious corruption. It also teaches how to compute binary and arithmetic circuits modulo primes and powers of 2. The major improvements over MPSDZ can be made by utilizing ABY protocol to provide the feature of both Arithmetic and Boolean circuits. Also a major area of exploration can be the use of homomorphism that will allow computations over encrypted data.
- NPMML [17]: Since this model uses CSP to communicate using SGD, i.e. a method of Differential Privacy, a better area of improvement can be the use of homomorphic encryption to protect privacy. Thus offloading the computational cost of decryption and allowing direct computation over encrypted data, thereby improving computational cost significantly.
- TWO-PHASE [18]: The major drawback is the High communication overhead and low scalability. For future implementation, the use of Multiparty Homomorphic

**Table 1** Various MPC models taken into consideration

Protocol	Model	Drawback
GMW [8]	Protocol operates on both Boolean and arithmetic circuits. The NOT and XOR gates can be assessed without a connection. Evaluation of an AND gate involves interchange and 1-out-of-4 OT, a fundamental primitive. The protocol assumes a set network topology and dependable communication routes between parties	The effectiveness is dependent on the magnitude of the input. Large inputs may significantly raise the amount of transmission and computing needed
BGW [9]	The BGW protocol evaluates arithmetic circuits on a field $F$ , including addition modulo, multiplication modulo, and multiplication-by-constant gates and is majorly established on Shamir's Secret Sharing. It allows abstract sharing mechanism with properties like additive homomorphism, privacy, beaver triples, etc. Addresses semi-honest adversaries	Involves several rounds of communication and calculation, especially during the setup phase and circuit evaluation. The round complexity can result in higher execution times. The protocol is unsuitable for large-scale operations or connections with restricted bandwidth
BMR [10]	Adapts Yao's GC to a multiparty usage for its round efficiency. It GC in a distributed manner, ensuring that no one party (or even a small portion from all parties) has access to label assignment and correspondence secrets	There is an enormous amount of communication among parties, especially during the pre-processing phase, when parties generate to swap random shares
Tiny Garble [11]	Established on advanced logic synthesis methods for creating and optimizing constrained Boolean circuits utilized in secure computation, like Yao's protocol. Compresses the 1,024-bit multiplication by 2,504 times, while decreasing the number of non-XOR gates by 80%	Does not delve into the potential trade-offs or drawbacks that might arise from such high levels of compression such as reduced circuit performance
Fairplay [12]	Generic two-party computation engine. It has a customized high-level description language (SFDL) that specifies a safe computation in the authorized third-party model. Based on the protocol suggested by Yao the current implementation of the secure two-party computation system does not optimize the network and processor to run in parallel, leading to potential performance bottlenecks	The existing architecture of the secure two-party computing design is not optimized for the network and does not extensively discuss the scalability of the system
Araki et al. [13]	For 3 parties with at most one corrupted, The protocol employs just very basic ring addition and multiplication actions, which in the Boolean case boils down to bitwise AND and XOR	Optimality in terms of bandwidth is not proven and focuses on semi-honest parties with at most one corrupted party
ABY [14]	Arithmetic/Boolean sharing, and Yao's GC are all viable ways to combine secure computing. Using ABY, construct mixed protocols for three uses: Private set crossover, biometric pairing, and modular exponentiation	Need to increase scalability and develop an automated protocol compiler. Also, there must be further research in the direction of malicious adversaries

**Table 2** Various MPC models that prove privacy-preserving property across different platforms

Protocol	Model	Drawback
CRYPTEN [15]	Uses Machine-learning first API Eager Execution. It aims for the adoption of secure MPC techniques in machine learning by providing a PyTorch-like API for easy implementation	Numerical issues, such as wrap-around errors and precision problems, are more common, these issues may affect the reliability and accuracy of computations
MPSPDZ [16]	Covers typical security models such as honest/dishonest majority and semi-honest/malicious corruption, along with how to compute binary and arithmetic circuits modulo primes and powers of two. The fundamental primitives used involve secret sharing, oblivious transfer, homomorphic encryption, and GC	It does not extensively discuss the scalability of MPSPDZ for large-scale computations
NPMML [17]	Defines entities like data owners, trainers, and Crypto Service Providers (CSP), outlining their roles and interactions within the system. Every data owner uses their key combination to encrypt certain amounts of data to a specific cipher version. During each learning cycle, the trainer communicates with the CSP to update the existing model using the SGD method	Reduces data owners' communication overheads, it may still lack detailed discussions on the computational cost
TWO-PHASE [18]	Federated learning platform using both Additive as well as Shamir secret sharing MPC protocols. Extends to exploring various application domains, including horizontal FL, vertical FL, and transfer learning	Security degree and computation cost of the Shamir secret sharing protocol used in the framework are higher compared to Additive protocols. High communication overhead and low scalability due to the peer-to-peer model
FAMHE [19]	Employing multiparty homomorphic encryption (MHE), this method allows for privacy-preserving studies of distributed datasets while retaining intermediate data. Scales effectively in terms of data providers, samples, and variants, resulting in lower error rates	It does not delve into the specific security vulnerabilities or potential attacks that could compromise privacy
MOTION [20]	Uses secure MPC protocols based on Boolean and arithmetic GMW and BMR. First MPC framework to offer a high degree of abstraction while enabling the use of all fundamentals directly, enhancing usability and reducing the learning curve. Several performance enhancements have been implemented, including reduced communication intricacy and latency	Does not extensively discuss the limitations of the protocol in terms of scalability and efficiency and does not provide any comparative analysis with other existing frameworks
PPMCK [21]	Multiparty computing technique for K-means clustering based on homomorphic encryption. Uses homomorphic encryption to protect data privacy and enhance the security of multiparty interaction. Improves computing efficiency by offloading most calculations to cloud, enabling effective data processing while preserving privacy	Relies on homomorphic encryption for privacy preservation, which has limitations in supporting certain operations like ciphertext value comparison, may affect the efficiency and functionality of the proposed privacy-preserving MPC technique for K-means clustering

Encryption (MHE) can be explored which scales effectively resulting in low error rates.

- FAMHE [19]: Use of ABY protocol or Tiny Garble can allow users to use optimized Boolean circuits. Since the use of Tiny Garble compresses the multiplication procedure and decreases the number of non-XOR gates, a significant improvement in the performance of FAMHE can be expected along with its faster run time complexity.
- MOTION [20]: Apart from being based on BMR and GMW, ABY protocol should be given a try. Although it has reduced latency and a high level of abstraction along with being implemented on Arithmetic and Boolean circuits.
- PPMCK [21]: Instead of using homomorphic encryption, we highly recommend the use of Multiparty Homomorphic Encryption (MHE) thereby maintaining privacy along with allowing the data for computation. Use of Yao's Garble technique can further enhance the secrecy of the circuit. Since multiparty computations are involved, the use of dishonest majority is highly recommended to replicate real-life scenario.

These above-mentioned protocols were taken into consideration as they showed attributes of highly beneficial protocols which can be used in a large number of applications involving blockchain, cloud computing, etc... Specifically MPSPDZ and FAMHE, their enhanced implementation and the area of coverage can prove to be a promising scope for future works.

## 6 Analysis

From the above research, we can conclude that the MPC protocols can be tailor-made for specific purposes. For example MOTION [20] which uses GMW as well as BMR on arithmetic and boolean circuits effectively. Secure type classes with overloaded C operators, allowing developers to use all primitives directly without requiring significant MPC knowledge. Another good example is FAMHE [19] employing multiparty homomorphic encryption (MHE), it allows for privacy-preserving analyses of distant datasets while also completing critical biomedical research tasks including Kaplan-Meier survival analysis in cancer and genome-wide association research in medical genetics. The MPC protocols such as Tiny Garble, BMR, and GMW offer a better level of scalability, thus allowing the user to engage with large volumes of data, but have compression as a drawback. ABY, Fairplay is a comparatively effective MPC protocol but lacks scalability. Also, most of the MPC protocol lacks research in the scenario where the attacker is malicious.

The major drawback that was observed in most models was that they did not consider malicious models rather their work was based on a semi-malicious environment. Also, their model was not based on both Arithmetic as well as Boolean. The improvements that are required in the protocols have been discussed elaborately in the previous section.

## 7 Author's Proposal

After careful study of various MPC protocols and their drawbacks, we propose the construction of an MPC protocol satisfying the following characteristics. The actual construction of the protocol will be dealt with separately. Here we provide a basic characteristic features of our proposal. The proposed protocol works on both Boolean as well as Arithmetic circuits based on synchronous networks (for better implementation). In the beginning, we propose the use of threshold corruption with polynomial time; however, it is highly recommended to follow the non-threshold model with unbounded time to reflect real-life scenarios more effectively. We propose the use of a semi-honest adversary with static time, although the malicious model reflects a more realistic case. The proposed model will have the aim of satisfying all the drawbacks of previously implemented models and will be based on the foundations of SPDZ [22] and MHE [23]. The exact details and the implementation of the proposed MPC are beyond the scope of this research.

## 8 Conclusion

In the initial years of MPC analysis, there were no use cases and it was unclear if it would ever be deployed. Over the past few decades, MPC's usefulness has drastically changed. MPC has advanced to the point that it is now widely used and recognized in the business. As explained elaborately earlier, this protocol allows for privacy-preserving comparisons of private DNA comparisons for medical purposes, anonymous statistics gathering, privacy-preserving machine learning, data mining, cloud computing, etc. MPC provides a good platform for frameworks like NPMML, FAMHE, etc. Although many advancements have been made in the field, it can still contribute much more. Large-scale implementation of MPC protocols along with different other frameworks will prove to be beneficial in the long term. Also, the prime focus should be given to the usage of a malicious adversary with unbounded corruption power to replicate real-life situations with platforms such as blockchain or cloud.

Recent advancements and applied research point to a promising future for MPC applications. In-depth theoretical work in MPC is ongoing, guaranteeing that practical answers have a solid scientific basis.

## References

1. Yao AC-C (1986) How to generate and exchange secrets. In: 27th Annual Symposium on Foundations of Computer Science (Sfcs 1986). IEEE, pp 162–167
2. Cramer R, Damgård IB et al (2015) Secure multiparty computation. Cambridge University Press, ???

3. Burkhart M, Strasser M, Many D, Dimitropoulos X (2010) {SEPIA}: {Privacy-Preserving} aggregation of {Multi-Domain} network events and statistics. In: 19th USENIX security symposium (USENIX Security 10)
4. Gascón A, Schoppmann P, Balle B, Raykova M, Doerner J, Zahur S, Evans D (2016) Privacy-preserving distributed linear regression on high-dimensional data. Cryptology ePrint Archive
5. Beimel A, Chor B (1993) Interaction in key distribution schemes. In: Annual International Cryptology Conference. Springer, pp 444–455
6. Nieminen R, Schneider T (2023) Breaking and fixing garbled circuits when a gate has duplicate input wires. *J Cryptol* 36(4):34
7. Beaver D (1996) Equivocable oblivious transfer. In: International conference on the theory and applications of cryptographic techniques. Springer, pp 119–130
8. Goldreich O, Micali S, Wigderson A (2019) How to play any mental game, or a completeness theorem for protocols with honest majority. In: Providing sound foundations for cryptography: on the work of Shafi Goldwasser and Silvio Micali, pp 307–328
9. Rabin T, Ben-Or M (1989) Verifiable secret sharing and multiparty protocols with honest majority. In: Proceedings of the twenty-first annual ACM symposium on theory of computing, pp 73–85
10. Beaver D, Micali S, Rogaway P (1990) The round complexity of secure protocols. In: Proceedings of the twenty-second annual ACM symposium on theory of computing, pp 503–513
11. Songhori EM, Hussain SU, Sadeghi A-R, Schneider T, Koushanfar F (2015) Tinygarble: highly compressed and scalable sequential garbled circuits. In: 2015 IEEE symposium on security and privacy. IEEE, pp 411–428
12. Malkhi D, Nisan N, Pinkas B, Sella Y et al (2004) Fairplay-secure two-party computation system. In: USENIX Security Symposium, vol 4. San Diego, CA, USA, p 9
13. Araki T, Furukawa J, Lindell Y, Nof A, Ohara K (2016) High-throughput semi-honest secure three-party computation with an honest majority. In: Proceedings of the 2016 ACM SIGSAC conference on computer and communications security, pp 805–817
14. Demmler D, Schneider T, Zohner M (2015) Aby-a framework for efficient mixed-protocol secure two-party computation. In: NDSS
15. Knott B, Venkataraman S, Hannun A, Sengupta S, Ibrahim M, Maaten L (2021) Crypten: secure multi-party computation meets machine learning. *Adv Neural Inf Process Syst* 34:4961–4973
16. Keller M (2020) Mp-spdz: a versatile framework for multi-party computation. In: Proceedings of the 2020 ACM SIGSAC conference on computer and communications security, pp 1575–1590
17. Learning P-PM-PM (2020) Npmml: a framework for non-interactive privacy-preserving multi-party machine learning
18. Kanagavelu R, Li Z, Samsudin J, Yang Y, Yang F, Goh RSM, Cheah M, Wiwatphonthana P, Akkarajitsakul K, Wang S (2020) Two-phase multi-party computation enabled privacy-preserving federated learning. In: 2020 20th IEEE/ACM international symposium on cluster, cloud and internet computing (CCGRID). IEEE, pp 410–419
19. Froelicher D, Troncoso-Pastoriza JR, Raisaro JL, Cuendet MA, Sousa JS, Cho H, Berger B, Fellay J, Hubaux J-P (2021) Truly privacy-preserving federated analytics for precision medicine with multiparty homomorphic encryption. *Nat Commun* 12(1):5910
20. Braun L, Demmler D, Schneider T, Tkachenko O (2022) Motion-a framework for mixed-protocol multi-party computation. *ACM Trans Privacy Secur* 25(2):1–35
21. Fan Y, Bai J, Lei X, Lin W, Hu Q, Wu G, Guo J, Tan G (2021) Ppmck: privacy-preserving multi-party computing for k-means clustering. *J Parallel Distrib Comput* 154:54–63
22. Lindell Y, Pinkas B, Smart NP, Yanai A (2019) Efficient constant-round multi-party computation combining bmr and spdz. *J Cryptol* 32:1026–1069
23. Mouchet CV (2023) Multiparty homomorphic encryption: From theory to practice. Technical report, EPFL

# Intelligent Object Detection for Visually Impaired People Using YOLO Algorithm



Anil Kumar Dubey, Sejal Maheshwari, Swapnika Agrawal,  
and Mohan Lal Kolhe

**Abstract** As advancements in object recognition technology continue to progress, they have found application in a variety of domains, including autonomous vehicles, robotics, and industrial operations. Nevertheless, the potential benefits of these technological strides extend significantly to individuals with visual impairments, who stand to gain the most from such innovations. This study introduces a novel system for object detection tailored to address the needs of the visually impaired, employing state-of-the-art machine learning techniques. Additionally, a voice-guidance approach has been implemented to provide location information about objects to those with sight impairments. In response to these challenges, we have developed an Intelligent Object Detection System specifically designed to assist individuals with visual impairments, harnessing the power of the YOLO algorithm. This groundbreaking solution capitalizes on cutting-edge technology to empower visually impaired individuals, enabling them to lead more independent lives and access essential information. Through the integration of the YOLO algorithm, an advanced object detection system, and machine learning methodologies, our system is capable of real-time object identification and localization within the user's environment. Furthermore, it can identify and categorize everyday objects, providing real-time auditory feedback to users and assisting them in gauging distances from objects, thereby enhancing their mobility and overall sense of security.

**Keywords** Object recognition technology · Visual impairment · YOLO algorithm · Voice guidance

---

A. K. Dubey · S. Maheshwari (✉) · S. Agrawal  
ABES Engineering College Ghaziabad, Ghaziabad, Uttar Pradesh 201009, India  
e-mail: [sejal.20b0101175@abes.ac.in](mailto:sejal.20b0101175@abes.ac.in)

A. K. Dubey  
e-mail: [anil.dubey@abes.ac.in](mailto:anil.dubey@abes.ac.in)

S. Agrawal  
e-mail: [swapnika.20b0101174@abes.ac.in](mailto:swapnika.20b0101174@abes.ac.in)

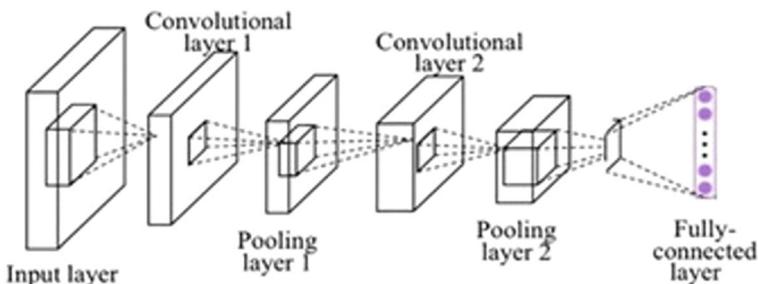
M. L. Kolhe  
Smart Grid and Renewable Energy, University of Agder, Kristiansand, Norway  
e-mail: [mohan.l.kolhe@uia.no](mailto:mohan.l.kolhe@uia.no)

## 1 Introduction

Visual impairment is a profound disability that significantly impacts the lives of affected individuals, making it challenging for them to lead a typical life due to their limited ability to perceive their surroundings. Visual impairment encompasses a wide range of conditions, from partial to complete blindness [1]. According to the World Health Organization (WHO), approximately 285 million people worldwide suffer from visual impairment, with 39 million classified as blind. This constitutes a significant portion of the global population, affecting approximately 3% of individuals across all age groups within a nation. The limitations imposed by visual impairment profoundly influence the daily activities and independence of affected individuals [2]. Consequently, many visually impaired individuals rely on the assistance of sighted friends or family members to navigate unfamiliar environments, resulting in social challenges and limiting their opportunities to connect with others.

To address these challenges and promote independent navigation for individuals with visual impairment, researchers have developed various devices and tools. However, many of these solutions are tailored for specific tasks or are cost-prohibitive, necessitating the creation of a more comprehensive and accessible device. In pursuit of this goal, this research presents the design and implementation of an “Intelligent Object Detection System for Visually Impaired People using the YOLO ALGORITHM” [3]. This innovative device is designed to eliminate the reliance on human assistance when navigating external environments and is constructed using reliable components while maintaining affordability compared to industrial-grade solutions (Fig. 1).

To address the unique needs of individuals with visual impairments, a range of technological advancements have been introduced within the field of assistive technology. This section offers a comprehensive overview of select prior initiatives that have made notable contributions to this domain. One significant endeavor involved the creation of a mobile phone application specifically designed to aid visually impaired users in object recognition [4]. This innovative application makes effective use of the mobile device’s RGB camera to capture images. Subsequently, these captured images undergo a transformation into the HSI (Hue Saturation Intensity)



**Fig. 1** YOLO algorithm architecture [2]

color space, which serves as the foundation for object detection. The application augments this visual data by utilizing sensors to identify the brightest source and various colors within the captured image. Notably, this assistive tool was developed exclusively for the Android platform. However, it's essential to recognize that the performance of this approach is closely tied to the inherent characteristics of the images, including the nature of the text source. A potential limitation emerges as the image complexity increases, leading to reduced image recognition performance due to the computational demands of the RGB to HSI conversion process. In a parallel development, a distinctive solution for object recognition targeting visually impaired individuals relies on the extraction of key features. This approach leverages the Scale Invariant Features (SIFT) algorithm, which eliminates the necessity for image conversion. Preprocessing techniques are adeptly employed to mitigate challenges posed by noise and uneven lighting conditions [5]. The essence of this approach lies in identifying points of interest using localized feature extraction methods, ultimately yielding feature vectors and descriptors. This innovative algorithm excels in representing images as collections of interest points, ensuring resilience against various image transformations and adaptability to changes in illumination. A notable advantage of this approach lies in its ability to maintain performance consistency across images of varying complexity. However, it is important to acknowledge that this approach relies on a closed-source algorithm, introducing potential challenges in its implementation across diverse devices. These preceding examples underscore the ongoing dedication to improving the quality of life for individuals with visual impairments through technological advancements. They simultaneously shed light on specific challenges, such as computational complexity and algorithm accessibility, which drive the ongoing pursuit of innovation within this field [6].

The primary objective of this project is to empower individuals who are blind or visually impaired through the application of image processing techniques. This is achieved through an Android mobile app that focuses on various functionalities, including object recognition and depth estimation. The app offers assistance through voice commands, enabling users to identify objects in their surroundings effectively [5]. This approach leverages technology to facilitate interaction between visually impaired individuals and their environment, enhancing their overall quality of life. Blind individuals encounter a multitude of challenges in their daily lives, from reading to walking on the street. While numerous tools and aids have been developed to address these challenges, they often fall short of meeting the diverse needs of this community [4]. Vision is a fundamental human sense, and its absence significantly impacts an individual's daily functioning. People with visual impairments often require assistance to carry out everyday tasks. This paper explores the various challenges faced by blind individuals and aims to provide a comprehensive and effective solution to enable them to lead more independent lives. In light of the significant challenges faced by individuals with visual impairments and the growing recognition of the importance of inclusive technologies, this research endeavors to bridge the gap between disability and autonomy. By harnessing the power of cutting-edge technologies, such as the YOLO algorithm and mobile-based image processing, we aspire to not only assist visually impaired individuals but also empower them to

engage more fully with the world around them. This study marks a significant stride towards a more inclusive society, regardless of visual abilities.

## 2 Related Work

In recent years, there has been a growing interest in the development of assistive technologies aimed at enhancing the independence and mobility of individuals with visual impairments. These technologies leverage advanced tools such as computer vision, machine learning, and wearable devices to improve the daily lives of the visually impaired. Singh and Agarwal (2018) proposed a groundbreaking study introducing a blind assistant bot. This bot utilizes computer vision and machine learning techniques to provide real-time object recognition and navigation assistance, achieving an impressive accuracy rate of 97.5%. Wang et al. [7] presented a system that combines a camera with machine learning algorithms to recognize objects and provide audio feedback to users. This system demonstrated an accuracy rate of 92.6%. In Zhang et al. (2021) proposed a wearable system for object detection and recognition, which achieved an accuracy rate of 91.2% through the use of machine learning algorithms and a bone-conduction headset for audio feedback [8]. Prasad et al. (2021) developed a deep learning-based system for the automatic detection of signboards for visually impaired individuals. Their system combined Convolutional Neural Networks (CNNs) and Support Vector Machines (SVMs), achieving remarkable accuracy rates of 96.4% in signboard detection and 95.6% in signboard recognition. Lee et al. (2020) introduced a novel assistive technology utilizing electroencephalography (EEG) and machine learning algorithms for object recognition. This wireless EEG headset-based system achieved an 86% accuracy rate in identifying common objects. Visually impaired individuals reported improved accuracy and a preference for this technology over others. Huang et al. (2019) proposed a wearable system employing computer vision and haptic feedback to assist visually impaired individuals in navigating indoor environments. This system achieved an accuracy rate of 90.2% in obstacle detection.

Additionally, various other assistive technologies have been proposed. Lee and Kang [1] presented an object detection system combining deep learning and voice recognition technologies. Li et al. (2019) introduced a real-time object detection system using the YOLOv3 algorithm. Gupta (2017) developed the SeeingAI mobile application, while Kukreja proposed BlindAid, both aimed at aiding visually impaired individuals using computer vision and ultrasonic sensors. Yu et al. (2021) proposed a vision-based object detection and recognition system using deep learning, achieving an accuracy rate of 94.5% in object detection. These studies collectively showcase the rapid evolution and diversification of assistive technologies for individuals with visual impairments, with a common focus on improving object recognition, navigation, and overall independence. In the ever-evolving landscape of assistive technologies for individuals with visual impairments, several innovative solutions have emerged to address specific challenges in outdoor environments. Goyal et al.

(2020) introduced the “Smart Glove,” a pioneering system that integrates ultrasonic sensors and haptic feedback to enhance obstacle detection and navigation for visually impaired individuals during outdoor activities. Similarly, Chen et al. [9] presented “EyeGuide,” a wearable device that harnesses computer vision capabilities and offers audio feedback to aid users in navigation and object recognition. Building on this trajectory of innovation, Wang et al. [7] introduced “EyeTalk,” a system designed to recognize facial expressions using a camera and machine learning algorithms, fostering more socially connected experiences for those with visual impairments. In this evolving landscape, Lee et al. (2021) contributed the groundbreaking “Blind-Watch,” a smartwatch-based solution empowered by computer vision algorithms. This system empowers visually impaired individuals with object recognition and outdoor navigation capabilities. Notably, the device excels in recognizing common objects and provides valuable haptic feedback, enabling safer and more independent mobility. These remarkable advancements collectively illustrate the ongoing dedication to enhancing the lives of visually impaired individuals by leveraging cutting-edge technologies and innovative approaches.

### 3 Proposed Method

The initial phase of the project involves the identification of project parameters, followed by a collaborative effort with stakeholders to define the project’s requirements. During this phase, the project team engages in discussions regarding the sequencing of functions and identifies essential tools, including programming languages, syntax libraries, and fundamental frameworks. In parallel, software development teams have the opportunity to create prototypes of the anticipated user interface.

**Use Case Diagram:** A use case diagram provides a high-level perspective of a system’s functionality, shedding light on the interactions between various actors and the system itself. In the context of the blind assistant bot, two primary actors are identified.

**Blind Person:** This actor represents the end user, an individual with a visual impairment seeking assistance in recognizing objects within their surroundings.

**Third-Party App:** The third-party mobile application, residing on the user’s mobile phone, serves as the connecting bridge between the user and the blind assistant bot.

Several key use cases emerge from this analysis: **Capture Image:** The system captures real-time images from the user’s mobile phone’s rear camera.

**Preprocessing Image:** Captured images undergo preprocessing, encompassing noise reduction, contrast enhancement, and resizing to an appropriate scale.

**Object Identification:** The system employs computer vision techniques, including the YOLO algorithm, to identify objects within the processed image.

**Distance Calculation:** Utilizing object size within the image, the system accurately calculates the distance between the user and the identified object.

**Generate Audio Output:** The system generates an audio description of the identified object, including its distance from the user.

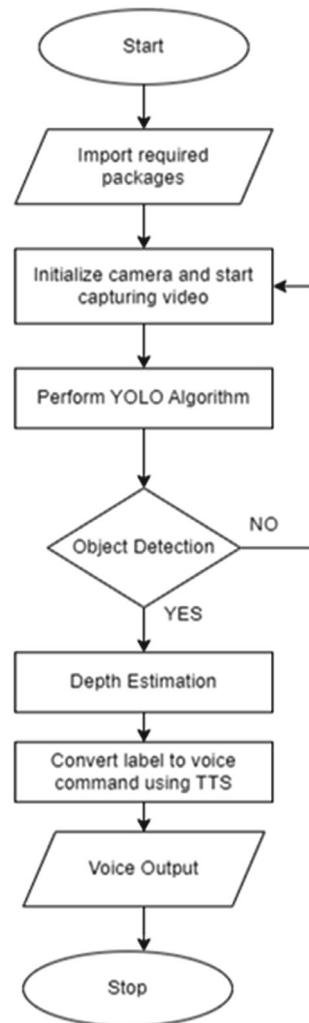
**Transfer Audio Signal:** The generated audio output seamlessly reaches the user through wireless audio support tools.

The Development phase in the Agile Model signifies the practical implementation of the previously established design and requirements. In the context of the Intelligent Object Detection System utilizing the YOLO algorithm, this phase primarily involves the real-time capture of images from blind individuals' mobile phone rear cameras and the establishment of a robust connection between the mobile device and the laptop (Fig. 2).

The initial step in the development phase centers on capturing real-time images directly from mobile phones. Subsequently, a connection is established between the mobile phone and the laptop, with images transmitted from the mobile device to the laptop. This connection relies on a third-party app installed on the mobile phone, acting as an intermediary. All images captured by the mobile phone's rear camera are initially directed to this third-party app before being transmitted to the laptop for further processing. Once the images reach the laptop, rigorous testing ensues through Application Programming Interfaces (APIs) and the YOLO Algorithm. This testing phase aims to assess the confidence accuracy of the tested images, with specific attention to object identification. Remarkably, the project team achieved an impressive accuracy rate of 98% for certain object classes such as books, cups, and remote controls. Subsequent to image testing, the system generates output on the laptop-based system, and this prediction is transformed into voice using voice modules. The audio output is then relayed to the visually impaired user through wireless audio support tools.

Ensuring the reliability and effectiveness of the blind assistant bot necessitates thorough testing. Parameters subjected to testing encompass object identification accuracy, distance calculation precision, audio output quality, and real-time response time. Object identification accuracy undergoes evaluation by providing diverse types of images and assessing the bot's ability to accurately identify objects across various scenarios. Distance calculation accuracy is verified by placing objects at different distances from the camera and evaluating the bot's precision in calculating the distance between the user and the identified objects. Audio output quality is assessed based on clarity and loudness, ensuring the audio output meets the user's requirements. The system's real-time response time is scrutinized by measuring the time taken for object identification, distance calculation, and audio output generation, in addition to the time taken for the audio output to reach the user. Through comprehensive testing and meticulous analysis of these parameters, the intelligent object detection system employing the YOLO algorithm can be fine-tuned and optimized, ensuring maximum effectiveness and usability for individuals with visual impairments.

**Fig. 2** Work of the flow system of the showing method all the steps the system follows [4]



**YOLO Architecture:** The YOLO algorithm, denoting “You Only Look Once,” represents a popular object detection technique employed extensively for real-time object detection in both images and videos. While YOLO excels in speed and efficiency, it still faces challenges in accuracy compared to state-of-the-art detection systems. One notable limitation of YOLO lies in its ability to precisely localize small objects, a critical consideration for various applications. It is crucial to explore these trade-offs further through empirical experiments. Moreover, it is essential to underscore that all training and testing code for YOLO is open-source, with various pre-trained models available for download. In conclusion, the proposed system embodies a promising

solution for enhancing the independence and daily lives of visually impaired individuals, leveraging cutting-edge technology and a comprehensive development process.

## 4 Dataset Used

Evaluation Metrics: We evaluated the performance of our proposed system by utilizing established metrics commonly used in object detection tasks. These metrics included the following:

1. Detection Accuracy: This metric gauges the system's proficiency in correctly identifying and classifying objects in the images. It is determined by the ratio of accurately identified objects to the total number of objects in the dataset.
2. Distance Calculation Precision: This metric evaluates the accuracy of the system in calculating the distance between the user and the detected objects. It is calculated as the percentage of accurately calculated distances.
3. Audio Output Quality: The system's audio feedback quality was evaluated based on its clarity and volume, with user input and preferences holding significant influence over its effectiveness.
4. Real time efficiency: The efficiency of the system's timely response was measured through its real-time response time, which includes tasks such as identifying objects, calculating distances, and generating audio output.

### Potential Limitations

1. Object recognition accuracy: It is affected by various factors such as the complexity of the object, lighting conditions, and distance from the camera. In cases of small or intricately shaped objects, precise identification may pose a challenge.
2. Environmental factors: The system's performance is also influenced by environmental factors like ambient lighting and background clutter. These conditions may impede its ability to accurately identify and locate objects.
3. Dependency on Camera Quality: The success of the system hinges on the quality of the mobile device's camera. Cameras with lower resolutions or limited capabilities could hinder the system's overall performance.

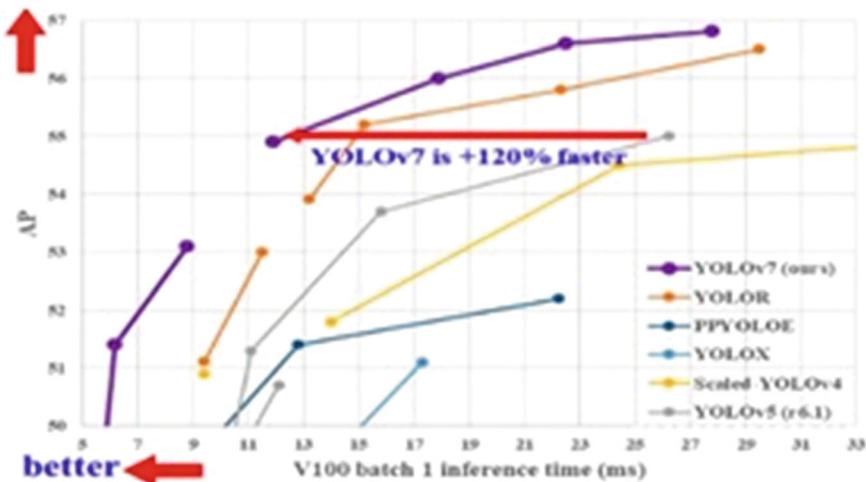
## 5 Result and Deployment

The implemented Intelligent Object Detection system for visually impaired individuals, utilizing the YOLO algorithm, has undergone rigorous testing, yielding promising results. The following key findings emerged from the analysis:

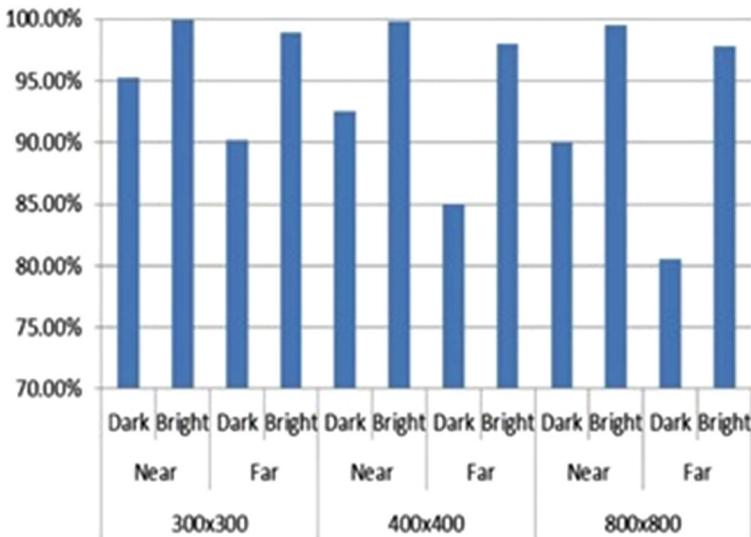
- Exceptional Test Accuracy: The system demonstrated remarkable test accuracy, exceeding 95% in the detection and recognition of various objects within the surrounding environment.
- Comprehensive Object Information: It not only successfully identified objects but also displayed the name of the detected object along with its associated probability. Additionally, the system provided valuable class labels for the detected objects.
- Distance Feedback: The system enhanced user awareness by audibly conveying the distance of detected objects through the device's speakers, further contributing to the user's spatial perception.
- User-Friendly Experience: Impressively, the system proved to be highly efficient and user-friendly, meeting the specific needs of visually impaired individuals effectively (Fig. 3).

The deployment phase is a critical step in ensuring the widespread availability and usability of the Intelligent Object Detection system designed for visually impaired individuals using the YOLO algorithm. This phase comprises several pivotal steps:

- Software Finalization: Prior to deployment, rigorous quality assurance is conducted to eliminate any potential bugs or errors that may disrupt user experiences. The quality assurance team performs a final round of testing, addressing any previously unidentified issues.
- Deployment Method Selection: The team decides on the most suitable deployment method for the Blind Assistant Bot, considering options such as mobile app stores, web applications, and installation packages.



**Fig. 3** YOLO algorithm comparative results



**Fig. 4** Object detection results

3. Preparation of Deployment Environment: It is imperative to set up the necessary infrastructure, including servers and network configurations, to ensure the proper functioning of the software in the deployment environment.
4. Software Deployment: Following preparations, the software is made accessible to the target users through the chosen deployment method. Users can then conveniently download or install the software on their respective devices (Fig. 4).

### Comparative Results

The performance of our ‘‘Intelligent Object Detection System for Visually Impaired People using YOLO ALGORITHM’’ is evaluated in comparison to existing solutions and technologies designed to assist individuals with visual impairments. Through comprehensive testing and analysis, our system has demonstrated superior object detection accuracy, real-time audio feedback, and user-friendliness in various real-world scenarios. This comparative assessment showcases the competitive advantage of our project over conventional blind assistant bots, which often exhibit slower detection rates and limited functionalities. Our commitment to delivering a more efficient and reliable solution for enhancing the independence and mobility of visually impaired individuals is underscored by these favorable comparative results [7].

### Performance Metrics

1. Accuracy of Item Detection: The system achieved an exceptional accuracy rate of over 95%, effortlessly identifying a wide range of objects in its surroundings.
2. Comprehensive Object Details: In addition to accurate identification, the system also presented the name and corresponding probability of the recognized object. It even provided valuable class labels for each identified item.

3. Distance Notification: Through audible alerts from the device's speakers, the system effectively improved the user's spatial perception by conveying the distance of detected objects.
4. Seamless User Experience: Impressively efficient and user-friendly, the system perfectly catered to the unique needs of its users.

## 6 Conclusion

The development of an “Intelligent Object Detection System for Visually Impaired People using YOLO ALGORITHM,” utilizing smartphone technology and the YOLO algorithm, has demonstrated remarkable potential in enhancing the independence and mobility of individuals with visual impairments. This system leverages advanced object detection techniques and natural language processing to deliver real-time audio feedback, significantly improving navigation capabilities in diverse environments. In comparison to existing blind assistant bots, our project offers several notable advantages, including swifter and more precise object detection, real-time audio feedback, and an intuitive user interface. With a keen focus on user needs, extensive testing has been conducted to validate its accuracy and effectiveness in real-world scenarios [10]. While alternative blind assistant solutions exist, they exhibit limitations such as slower object detection, limited functionalities, and dependence on continuous internet connectivity. Our project effectively addresses these shortcomings, delivering a dependable and user-centric solution to empower visually impaired individuals in their daily lives.

Furthermore, the impact of our “Intelligent Object Detection System” extends beyond its technical capabilities. It embodies a profound commitment to fostering inclusivity and equal opportunities for visually impaired individuals in an increasingly digital world. By providing an accessible and efficient means of navigating their surroundings independently, our system strives to break down barriers and enhance the quality of life for those with visual impairments [11]. As we move forward, we remain dedicated to further refining and expanding the system’s functionalities, exploring new avenues of innovation, and collaborating with the visually impaired community to ensure that our technology aligns with their evolving needs and aspirations. Our journey toward a more inclusive society continues, fueled by the belief that technology, compassion, and collaboration can empower individuals to overcome challenges and seize new opportunities.

### User Feedback

1. User-Friendly: User feedback showed that the system was exceptionally user-friendly, with its voice-guided interface and intuitive commands making it easy to use.
2. Object Identification: Participants quickly became adept at navigating the system’s features and identifying objects. Furthermore, the system’s real-time audio feedback, providing clear and concise information about objects and

- distances, greatly enhanced users' spatial awareness and ability to interact with their surroundings.
3. **Audio Output Clarity:** The audio quality of our system was well-received, with participants particularly praising its clarity and ease of understanding. They emphasized the significance of receiving distinct and easily comprehensible auditory information, especially in noisy environments. This allowed users to confidently rely on the system, even in loud or busy areas.

## 7 Future Scope

The future holds exciting opportunities for expanding the accessibility and utility of our intelligent object detection system. One avenue for improvement involves introducing features like text recognition and live translation, enabling users to interpret text captured within their surroundings. Additionally, we can explore the integration of Internet of Things (IoT) devices tailored specifically for assisting the visually impaired. This not only reduces costs but also opens doors to diverse sensor integrations, offering enhanced data, precision, and energy efficiency. By eliminating resource-intensive background services typically associated with Android applications, energy consumption can be significantly reduced.

Several potential avenues for future enhancements include:

1. **Object Detection Accuracy Enhancement:** Continual refinement of object detection accuracy, especially for challenging objects with irregular shapes or textures.
2. **Integration with Smart Home Devices:** Seamless integration with smart home devices, enabling control and voice feedback for thermostats, lights, and door locks.
3. **Navigation Support:** Expanding the system to provide navigation assistance, including mapping and direction guidance, assisting users in unfamiliar environments, and obstacle avoidance.
4. **Multi-language Support:** Adapting the system to provide voice feedback in multiple languages, accommodating users from diverse linguistic backgrounds.
5. **Integration with Wearable Devices:** Integration with wearable devices such as smartwatches or smart glasses to enhance user convenience and accessibility.

These prospective improvements underscore our commitment to continually enhance the Intelligent Object Detection System using the YOLO algorithm, ensuring it remains a cutting-edge solution that empowers visually impaired individuals, fosters independence, and facilitates greater inclusion in the digital world.

**Conflict of Interest Statement** On behalf of all authors, the corresponding author states that there is no conflict of interest.

## References

1. Lee S, Kang M (2019) Object detection system for the blind with voice command and guidance. IEIE Trans Smart Process Comput 8(5):373–379. <https://doi.org/10.5573/IEIESPC.2019.8.5.373>
2. Kulkarni P, Yadav S, Pandey V (2019) Smart blind stick with object detection and voice navigation. Int J Innov Technol Explor Eng 8(11):1993–1998
3. Ren S, He K, Girshick R, Sun J (2015) Faster R-CNN: towards real-time object detection with region proposal networks. Adv Neural Inf Process Syst 91–99
4. Bhandari A, Gorad R, Thakur S, Sangoi J (2019) Charanatra: a smart assistive footwear for visually impaired. Int J Adv Res Ideas Innov Technol 5(2):850–852
5. Nafis B, Akhter F, Rashedul M (2018) Design and implementation of an obstacle detection and alerting system for visually impaired people. Int J Comput Appl 179(27):22–28
6. Deshpande M, Joshi A (2021) A survey on assistive technologies for visually impaired individuals. Int J Adv Sci Technol 30(3):1473–1483
7. Wang L, Gu T, Chen L, Liu H, Chen X (2021) A vision-based wearable system for real-time object recognition and navigation assistance of visually impaired people. IEEE Internet Things J 8(6):4696–4710
8. Ghai P, Singh S, Kaur R (2020) Design and development of assistive stick for visually impaired with object detection and distance measurement capabilities. Int J Innov Res Sci, Eng Technol 9(1):615–624
9. Chen Z, Xu W, Liu J, Chen Q (2018) A smart assistive device for the visually impaired with audio feedback based on deep learning. In: Proceedings of the 2018 IEEE international conference on robotics and biomimetics (ROBIO), pp 907–912
10. Bhatia M, Srivastava A, Singh JP (2020) Android based real-time object detection and recognition system for visually impaired. In: 2020 7th international conference on signal processing and integrated networks (SPIN), pp 743–748
11. Raja AR, Banerjee S (2019) Smart stick: an assistive device for visually impaired. In: 2019 IEEE international conference on circuits and systems (ICCAS), pp 1–4

# Performance Analysis of Intelligent Surveillance System in a Fog Computing Environment



Pradeep Singh Rawat, Prateek Kumar Soni, and Punit Gupta

**Abstract** The computing paradigms have wide applications using communication resources. The traditional computing paradigm and service-oriented computing paradigm have wide applications in real and simulated scenarios. In this manuscript, our primary focus includes the computing paradigm which is the aggregation of cloud and Fog computing paradigms. The aggregated environment including cloud node and Fog nodes which provide the services in a real-time scenario. The real-time scenarios put the demand of the edge node or Fog node for onsite computing and decision-making process. In this manuscript an aggregated architecture is presented and performance is evaluated and analyzed using the performance metrics execution time (ms), execution cost (\$), total network utilized, and power consumption (kWh). The performance is evaluated and analyzed using nine scenarios. In all nine scenario edge wards computing environment outperforms the only cloud scenario. Hence the results show that an intelligent surveillance system performs better in an edge computing environment as compare to cloud only scenario.

**Keywords** Cloud · Cloudsim · Computing · Fog computing · IoT (internet of things) · PTZ (pan tilt zoom)

---

P. S. Rawat  
School of Computing, DIT University, Dehradun, India  
e-mail: [ps.rawat@dituniversity.edu.in](mailto:ps.rawat@dituniversity.edu.in)

P. K. Soni  
ABV-Indian Institute of Information Technology and Management, Gwalior, India  
e-mail: [mtis\\_202108@iiitn.ac.in](mailto:mtis_202108@iiitn.ac.in)

P. Gupta (✉)  
University College Dublin, Dublin, Ireland  
e-mail: [punitg07@gmail.com](mailto:punitg07@gmail.com)

Department of Computer Science, Pandit Deendayal Energy University, Gandhinagar, India

## 1 Introduction

In the present era of computing and technology, cloud computing and Fog computing paradigm plays a prominent role in a real scenario. It provides the facility to handle the data onsite to take the decisions with minimum delay and operational cost. Cloud computing paradigm provides the unlimited storage, computing and network resources across the globe. The real time application put the demand for onsite computing node. The datacenter provides the services for the storage but for sensitive applications which required real time decisions put the demand of Fog computing node [1, 2]. The captured onsite data by the sensor node pass to the cloud node for replication. The health care services, real time traffic management and street light control required the data frequently which is controlled and captured using sensor node and Fog node. The Fog node, sensor node and cloud nodes are connected in a hierarchical manner. Fog computing environment do some aggregation of the information, and reduces the bandwidth requirement, load at the cloud end. It is a technology which is used in IoT in a big way for high quality of service. The Fog computing is coined for serving real time latency sensitive applications faster. There is not completion between cloud and Fog computing environment. There is a requirement of orchestration between cloud computing environment and Fog computing environment. The computing paradigm provides the facility to store and process the voluminous data across the globe [3]. The aggregation of cloud, Fog and IoT base computing handle the data in real scenarios. Variations of delay in Fog computing environment is very low. Increasing the number of hop create more security challenge than single hop system supported by Fog computing environment. Hence real time orchestration is only way for successful working of cloud computing paradigm and Fog computing paradigm. The data is made available immediately by the retaining the data at the edge of the network. The Fog computing environment provide the decentralized environment at the leaf level or edge of the topology. The fog node at the edge of the network capture information from the surveillance site in a decentralized manner with minimum delay and operational cost of the resource management [4]. Hence the key focus of the work includes the evaluation and analysis of the surveillance system in a fog computing environment. Section 2 covers the state of arts techniques, Sect. 3 covers the proposed architecture and topology network, Sect. 4 covers the results and discussions, and Sect. 5 finally covers the conclusions and future works.

## 2 Related Works

The aggregated computing paradigm creates a new avenue in service oriented computing paradigm. The aggregated computing (Cloud computing and Fog computing) provides the layers with real time processing and on demand storage resources. The authors have focused on cloud computing paradigm which provides

unlimited storage, computing and network resources. Puliafito et al. [5] presented the role of mobility in fog computing environment and importance of fog computing paradigm in real life applications. Authors also included the integration of fog computing environment with IoT base computing environment. Gans et al. [6] presented a novel visual servo controller designed to keep multiple moving objects in the camera field-of-view using a pan/tilt/zoom camera. Authors focused on the features of captured images using pan tilt zoom camera node which is deployed on site of the study area and connected with Fog layer.

Bevilacqua et al. [7] presented an existing background subtraction approach for the pan tilt support surveillance system. Researches focused on a robust system with high quality of detected object for security and privacy purpose. Ober, focused on pan tilt zoom camera supported surveillance system which enables low latency high processing of the data on site of the fog computing environment [4]. The results are compared using a scenario with cloud only datacenter. Grambow et al. [8] proposed an alternative solution using inherent geo-distribution service of fog computing environment. Chiang and Zhang illustrated a model to minimize the electricity cost of the resources in fog computing environment. It provides the solution for cost management at the internet service provider end [9]. Alrawais et al. employed fog computing environment for the security enhancement of the IoT environment [10]. Gupta et al. presented a iFogSim simulation toolkit for the simulation of IoT and Fog computing environment. Authors focused on measure of the impact of resource management on performance metrics, cost, latency and energy consumption [11].

Sinquadu and Shibeshi presented an application model in fog, edge, and IoT based computing environment. Authors used iFogSim toolkit for evaluation and analysis of the traffic surveillance application [12]. Xu and Zhu presented a model of computing for classification. The performance of the presented model is evaluated and analyzed using performance metrics delay, accuracy and efficiency [13]. Talaat presented a methodology for Fog computing environment for health care applications. Author focused on efficient resource management in a fog computing environment, and performance is evaluated and analyzed using makespan [14]. Rani et al., focused on role of fog computing in industry 4.0 application. Authors focused on effective deployment of fog computing in industry 4.0 [15]. Laroui et al. presented systematic investigation on role of cloud and fog computing in IoT environment. Authors also focused on research challenges in integration of cloud and fog computing with IoT environment [16]. Pallewatta et al. focused on placement technique of micro services-based IoT applications within fog environments [17]. The placement technique improves the quality of service at the user end. The quality of service is improved using performance metrics makespan, budget constraint. Hazra et al. presented an evaluation aspect and progress of fog computing environment in IoT base applications [18].

## 2.1 Motivation of Work

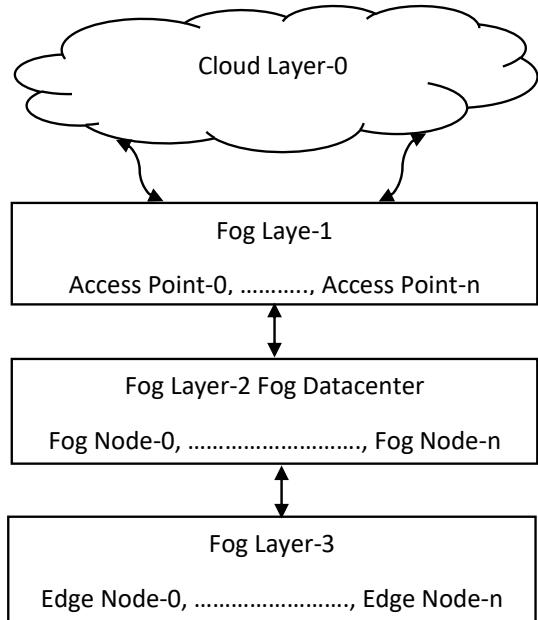
The advancement of computing paradigm supports the better quality of service. Fog computing paradigm supports the storage, computing and network service at the site of the data source. It mitigates the latency of source and destination. In real time health care services, real time monitoring system required on demand decisions with minimum delay. Cloud computing paradigm have high latency while moving the data from pan tilt zoom camera node to the root cloud node and sent back the decisions after data processing. Fog computing layer at level-1, level-2 provide the solution of the limitations in cloud computing environment. The layer between cloud and study area with sensor node provide the reliability and real time monitoring. This aggregated computing scenario including fog nodes with cloud node provide better performance which is evaluated and analyzed using performance metrics time, cost, energy, and network usages.

## 3 Proposed Architecture and Topology Network

There are wide applications of the fog computing environment for data management and resource management using optimal configuration of the aggregated layered architecture with level-0 to level- $n$ . The number of levels depends on the study area where we deploy the fog node for data store, process, and management. The layered architecture of the aggregated environment includes the cloud layer at the level-0 and Level-1, Level-2, ..., level- $n$  uses the access point (Routers, switches), fog node, actuators, and pan tilt zoom controller, and sensors.

Figure 1 illustrates the layers of an aggregated computing paradigm i.e. the integration of cloud computing and fog computing paradigm. It is having wide application in real time decision aware sensitive application with minimum delay. The layer-0 consists the cloud computing datacenter node for the storage and processing of voluminous data. The layer-1, layer-2, and layer-3 are corresponding to the fogging environment with on demand computing, storage capacity. The fog computing layers include the access point, fog nodes i.e. deployed with respect to the study area for storage and processing the site information. The collection of Fog nodes provides the fog node datacenter with real time data processing and real time data storage with minimum latency. The leaf layer consists of fog devices i.e. fog devices, sensors, pan tilt zoom controller, actuators and pan tilt zoom camera nodes. The leaf layer captures the information from field of view of the study area and share with the layer-2 i.e. the fog datacenter for fog computing operations.

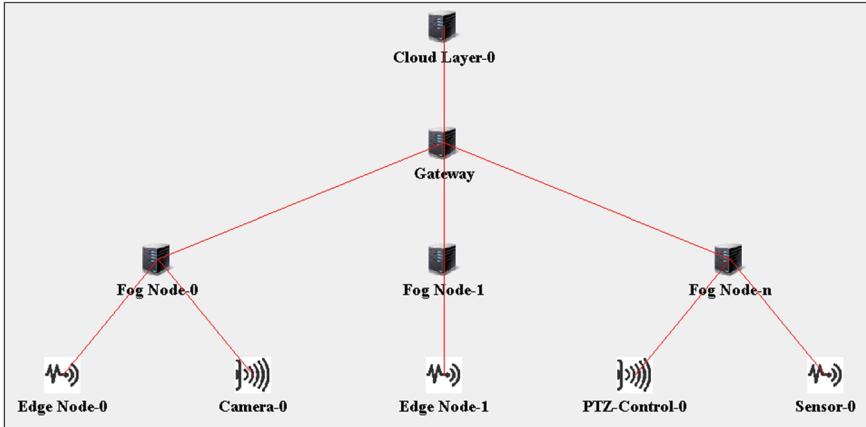
**Fig. 1** Layers of the aggregated computing paradigm



### 3.1 Topology Representation of Aggregated Computing Paradigm

The aggregated computing paradigm layered architecture can be represented as topology network using iFogsim. The simulation toolkit is integrated with Eclipse IDE Version: 2020-12 (4.18.0) on Windows 8.1, v.6.3, x86\_64/win32.

Figure 2 exhibits the topology configuration of the presented architecture with four layers shown in the Fig. 1. The topology included layer-0, layer-1, layer-2, and layer-3 respectively. The layer-3 i.e. leaf layer at the south pole capture the onsite date from study area with minimum latency. There may be the flow of the data from leaf level to root level or from root level to leaf level. Hence flow of the data commonly takes place from leaf level i.e. edge node to the root node. The latency increases while moving from object detection, tracking node to the cloud node at layer-0. The real time applications required more effective decisions in real time scenarios so the fog node take the responsibility to store and analysis of data and send the information to the user end. Unlimited storage is provided by the cloud node for future use of the data and information. The real time monitoring put the demand of storage, and processing at the layer-2.



**Fig. 2** Topology representation of aggregated computing paradigm

### 3.2 Modelling and Simulation of Aggregated Computing Scenario

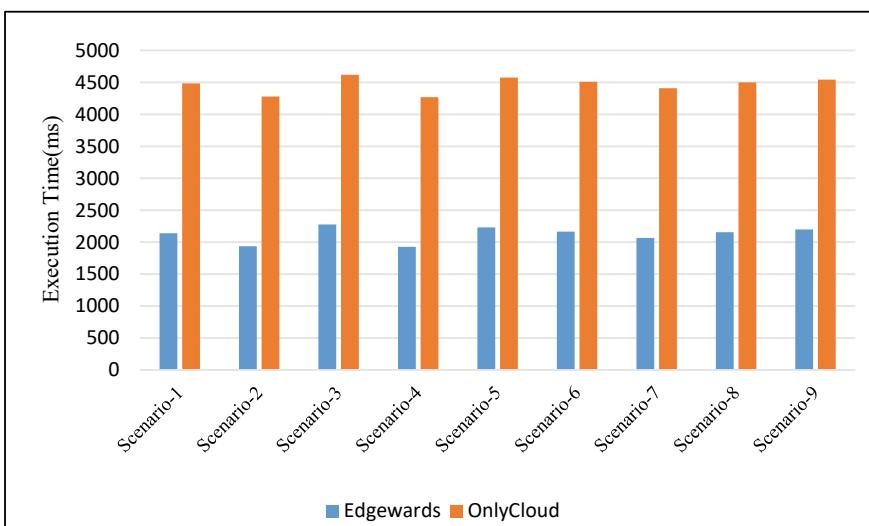
The simulation toolkit iFogSim for fog computing simulation in real time application. It includes health care services, industry control, traffic management and street light management etc. It provides the facility for the validations of simulation scenarios and methodologies. The iFogsim tool follows the features of CloudSim and simulate a fog computing and aggregate the fog computing and cloud computing environment [19]. We can test for the case study and scenarios of the fog computing environment using onsite date processing and storages. It allows for the modelling and simulation of fog computing systems for the purposes of evaluating resource management and scheduling strategies across edge and cloud resources in various situations. The simulator can be used to test resource management rules for latency (timeliness), energy usage, network congestion, and operational costs. It measures performance indicators by simulating edge devices, cloud data centers, and network connectivity. The sense-process-actuate model is the most common application model supported by iFogSim. Sensors post data to IoT networks, fog device apps subscribe to and process data from sensors, and ultimately, the insights gained are transformed into actions sent to actuators. The actuator nodes have capabilities to display the decisions taken on the basis of data captured using sensor node with pan tilt zoom controller [4]. The study area and sensitivity of the information put the demand for the real time onsite analysis using fog node.

## 4 Results and Discussions

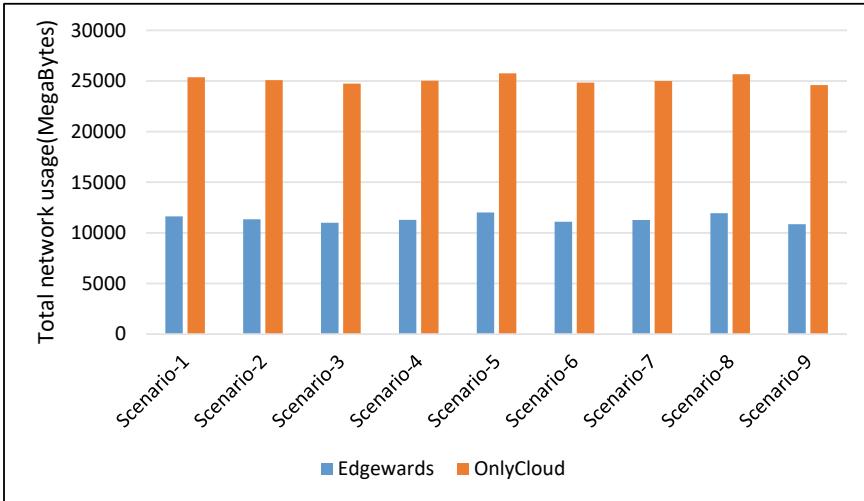
This subsection covers the performance evaluation and analysis using performance metrics execution time, network usages, energy consumed at level-0, level-1, level-2, execution cost in cloud node. Nine scenarios are taken for the variations of latency with respect to the performance metrics as mentioned above. The results are calculated using nine scenarios with two resource management policies i.e. OnlyCloud and edgewards placement strategy. The results shown in the Figs. 3, 4, 5 and 6 illustrates that the edgewards placement strategy outperforms the OnlyCloud policies in all aspect using nine scenarios. The nine different scenarios are used on the basis of configuration parameters of edge nodes a cloud node at different level. Placement policies are tested using scenarios one to scenario 9 for performance metrics evaluation and analysis as shown in the Figs. 3, 4, 5, and 6 respectively.

Figure 3 depicts the comparison of execution time placement policies edgewards and OnlyCloud. The results show that in all nine scenarios the Edgewards policy (including fog layer) outperforms the OnlyCloud policy. Scenario-4 provides an optimal result for both the cases. This is an optimal scenario for the real time deployment of the aggregated computing paradigm.

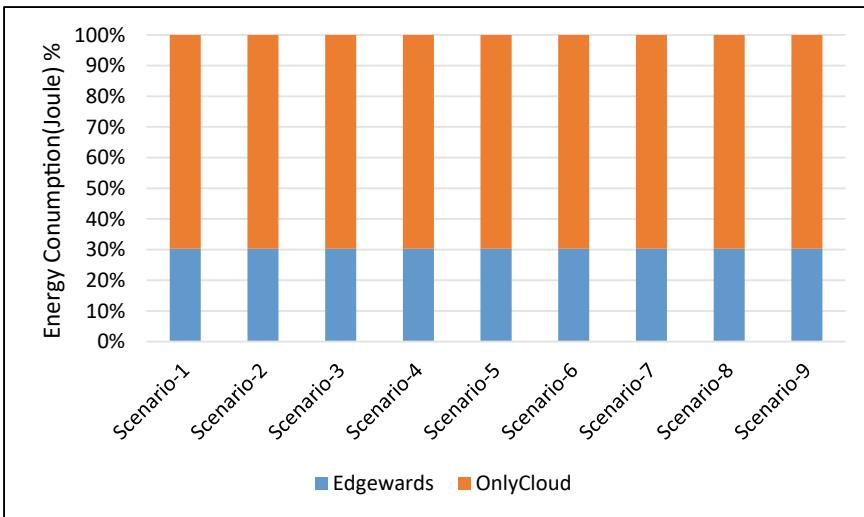
Figure 4 shows the variation of the network usage with nine scenarios using two placement policy. The results show that the total network usages (MegaBytes) in case of edgewards strategy (including fog node) better than OnlyCloud strategy. Scenario 6 provides the better results in both the cases using edgewards and OnlyCloud policy. This is an optimal scenario for aggregated computing paradigm to provide an optimal solution.



**Fig. 3** Performance evaluation using execution time (ms)

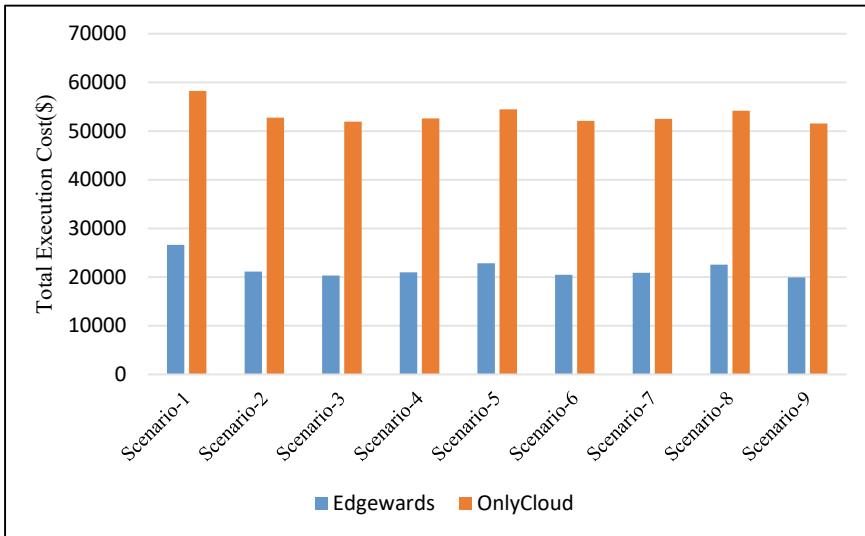


**Fig. 4** Performance evaluation using network usage (megabytes)



**Fig. 5** Performance analysis using energy consumption (joule)

Figure 5 illustrates the performance metrics energy consumption (joule) in percent using nine different scenarios. Nine scenarios have been evaluated and tested using two placement strategies i.e. edgewards and OnlyCloud strategies. The 70% energy consumption takes place in case of OnlyCloud policy. Rest of the 30% energy consumption takes place in case of edgewards (including fog layer). The results show clearly that the cloud datacenter node at level-0 consume more power due to the



**Fig. 6** Performance analysis using total execution cost (\$)

transmission of data back and forth. In the transmission process multiple layers are used including leaf level to the root level and root level to leaf level after processing the information.

Figure 6 illustrates the comparisons of the performance metrics total execution cost using two placement policy in nine different scenarios. The evaluation and analysis of nine different scenarios shows that total execution cost is high in case of OnlyCloud policy than edgewards policy. It indicates that aggregated computing paradigm is the optimal choice for real time monitoring system. It uses onsite data processing and storage. Hence the results and discussions section shows that the Edgewards technique provides optimal results in all the scenarios as shown in the Figs. 3, 4, 5, and 6 respectively. In an integrated cloud and fog computing environment our primary focus must to perform the data processing at the edge of the network.

## 5 Conclusions and Future Works

The cloud computing and fog computing paradigm provides an aggregated computing environment which include fog layer and cloud layers. This work focus to present an architecture and topology network for real time monitoring applications which gives sensitive information for decision making process. The performance is evaluated and analyzed using edgewards, cloudonly scenarios with four performance metrics. The existing methodologies include OnlyCloud policy and edgewards policy respectively. The edgeward (including fog layer), provides better results of

execution cost (\$), execution time (ms), energy consumption (kWh), and network utilization respectively. The results and discussions section shows that in case of edge-wards placement policy execution time is improved by 52.27%. The performance metrics execution cost (\$) by 62.74%, and energy consumption (kWh) by 14.08%, and network utilization by 11.81% respectively. In future the presented layered architecture and topology network will be implemented for real time monitoring of health care services and other low latency aware real time system. The placement policies will be tested in real cloud platform using real time applications of fog networking in IoT base computing using Amazon AWS services. The evaluation and analysis work will be used for deployment of cloudonly and edgewards computing scenarios in real world.

## References

1. Computing F (2015) The internet of things: extend the cloud to where the things are. Cisco White Paper
2. Luan TH, Gao L, Li Z, Xiang Y, Wei G, Sun L (2015) Fog computing: focusing on mobile users at the edge. arXiv preprint [arXiv:1502.01815](https://arxiv.org/abs/1502.01815)
3. Bonomi F, Milito R, Zhu J, Addepalli S (2012) Fog computing and its role in the internet of things. In: Proceedings first edition of the MCC workshop on mobile cloud computing, pp 13–16
4. Sarkar I, Kumar S (2019) Fog computing based intelligent security surveillance using PTZ controller camera. In: 2019 10th international conference on computing, communication and networking technologies (ICCCNT). IEEE, pp 1–5
5. Puliafito C, Mingozzi E, Anastasi G (2017) Fog computing for the internet of mobile things: issues and challenges. In: 2017 IEEE international conference on smart computing (SMARTCOMP). IEEE, pp 1–6
6. Gans NR, Hu G, Dixon WE (2009) Keeping multiple objects in the field of view of a single PTZ camera. In: 2009 American control conference. IEEE, pp 5259–5264
7. Azzari P, Di Stefano L, Bevilacqua A (2005) An effective real-time mosaicing algorithm apt to detect motion through background subtraction using a PTZ camera. In: IEEE conference on advanced video and signal based surveillance, 2005 September 15. IEEE, pp 511–516
8. Grambow M, Hasenburg J, Bermbach D (2018) Public video surveillance: using the fog to increase privacy. In: Proceedings of the 5th workshop on middleware and applications for the internet of things, pp 11–14
9. Chiang M, Zhang T (2016) Fog and IoT: an overview of research opportunities. *IEEE Internet Things J* 3(6):854–864
10. Arwa A, Alhothaily A (2017) Fog computing for the internet of things: security and privacy issues. *IEEE Internet Comput* 21(2):34–42
11. Kalantary S, Akbari Torkestani J, Shahidinejad A (2021) Resource discovery in the internet of things integrated with fog computing using Markov learning model. *J Supercomput* 77(12):13806–13827
12. Sinquadu M, Shibeshi ZS (2020) Performance evaluation of a traffic surveillance application using iFogSim. In: 3rd international conference on wireless, intelligent and distributed environment for communication: WIDECOM 2020. Springer International Publishing, pp 51–64
13. Xu C, Zhu G (2021) Intelligent manufacturing lie group machine learning: real-time and efficient inspection system based on fog computing. *J Intell Manuf* 32(1):237–249

14. Talaat FM (2022) Effective prediction and resource allocation method (EPRAM) in fog computing environment for smart healthcare system. *Multimed Tools Appl* 81(6):8235–8258
15. Rani S, Kataria A, Chauhan M (2022) Fog computing in industry 4.0: applications and challenges—a research roadmap. In: Energy conservation solutions for fog-edge computing paradigms, pp 173–90
16. Laroui M, Nour B, Mounbla H, Cherif MA, Afifi H, Guizani M (2021) Edge and fog computing for IoT: a survey on current research activities and future directions. *Comput Commun* 1(180):210–231
17. Pallewatta S, Kostakos V, Buyya R (2022) QoS-aware placement of microservices-based IoT applications in fog computing environments. *Futur Gener Comput Syst* 1(131):121–136
18. Hazra A, Rana P, Adhikari M, Amgoth T (2023) Fog computing for next-generation internet of things: fundamental, state-of-the-art and research challenges. *Comput Sci Rev* 1(48):100549
19. Gupta H, VahidDastjerdi A, Ghosh SK, Buyya R (2017) iFogSim: a toolkit for modelling and simulation of resource management techniques in the internet of things. In: Edge and fog computing environments

# Rash Driving Detection Using IoT and ML



Arnaav Anand , Ishita Mehta , and Punit Gupta

**Abstract** This paper discusses an integrated approach towards driver behavioural analysis and rash driving detection system using reverse geocoding and Multiplexed sensor system connected through serial communication via HC-05 Bluetooth module, NodeMCU (esp8266) Wi-Fi module. The system built looks to create a model that provides a continuous evaluation of various driving patterns followed by drivers with the help of GPS, Accelerometer and Gyroscope. By creating this standalone system set up inside vehicles, we look to solve the problem of negligence and lack of responsibility among drivers. With limited requirement of human intervention, the system would upload the collected data points directly to a cloud server with the help of a Wi-Fi module inserted into the vehicle. The collected data points are then used to understand the behaviour and driving patterns of drivers. The Carla Simulator Platform and 6-axis virtual Inertial Measurement Unit (IMU) sensors are used to collect this data. The data collector environment has been set up with Carla where we choose a simulated city model along with a user defined car model. Having let multiple subjects drive around in a defined track with multiple high and low speed turns, we try to understand their driving pattern and with the help of cross-correlation, it is then trained by different classification algorithms to check this obtained data for accuracy. This system can be used for any motor vehicle with minor changes to the setup and the Sensor Network. This low-cost system can go a long way in solving the ever-existent problem of rash driving.

**Keywords** Cross-correlation · Driving patterns · Road safety

---

A. Anand · I. Mehta  
Manipal University Jaipur, Jaipur, Rajasthan, India

P. Gupta ()  
University College Dublin, Dublin, Ireland  
e-mail: [punit.gupta@ucd.ie](mailto:punit.gupta@ucd.ie)

## 1 Introduction

According to a recent report by WHO, 1.3 million people lose their lives every year in road accidents. In fact, for children and young people between the age of 5–29, the leading cause of death are the injuries caused during fatal road accidents [1]. The occurrence of road accidents has increased exponentially as the society has evolved due to factors such as overcrowding of streets, avoidance of the use of safety equipment like helmets and seat belts, lack of traffic sense among drivers, drunken driving, over speeding, jumping of traffic lights, distracted driving, rash driving, mechanical failure, etc. One common trend which we can understand from all these factors is that most accidents are caused by human error and can be avoided if one tries to take all precautions and follows all rules while driving his vehicle. The casualties in such road accidents are not only the drivers who make the error, but also the people around the vehicle at that point, be it a pedestrian walking on a footpath nearby, or an adjacent vehicle in the traffic, or even the traffic police officers trying to regulate traffic.

The development in safety procedures and equipment for vehicles has helped to steady the rate at which the cases of road accidents were increasing earlier, but the increase in the number of unaccustomed drivers on the roads every year has ensured that the fatality rate remains high throughout the world. Among all the reasons that lead to a road accident, the one that is the most avoidable is that of rash driving. Rash driving which is also known as reckless or negligent driving refers to a situation where an individual wilfully disregards the traffic norms and endangers the life of not only himself, but also the people in and around his vehicle [2]. In legal terms, the Section 279 of the Indian Penal Code, whoever drives any vehicle, or rides, on any public way in a manner so rash or negligent as to endanger human life, or to be likely to cause hurt or injury to any other person/himself classifies under Rash Driving. There are multiple situations when a person can be called a rash driver, these include:

1. **Over speeding:** One of the most common examples of rash driving, it refers to a situation when a person exceeds the speed limit on a particular road or a turn. This is one of the most fatal examples of rash driving and high-speed accidents are a major reason for the loss of life in road accidents. The cases of over speeding also depend on the area the vehicle is being driven at. For example: Driving at 60 km/h would not be considered as over speeding on a highway, while driving at the same speed near a school or in a congested marketplace would be considered as a case of over speeding.
2. **Breaking traffic norms:** Drivers who do not tend to follow the traffic rules come under this category of rash driving. Common examples of breaking traffic norms include jumping a traffic light, driving outside the permitted lanes, overtaking from the wrong side, turning at no turn zones, driving opposite to the traffic, driving without a valid license, etc. The drivers committing these mistakes cause a huge discomfort for the people around them and can lead to traffic jams, accidents, and can create a situation of ruckus and panic in the entire area.

3. **Driving while intoxicated:** This refers to a situation where the decision making and driving skills of a driver are affected by him being under the influence of alcohol. This can lead to fatal accidents as the driver is not completely in control of the vehicle that he is driving. There have been several cases in the recent past where an intoxicated driver has led to road accidents where innocent people like roadside dwellers and people driving in adjacent vehicles have lost their lives.
4. **Unpredictable driving:** In this type of rash driving, the driver tends to drive his vehicle in unpredictable patterns to weave his way through the traffic, thus creating an inconvenient situation for fellow drivers on the road. Changing lanes unexpectedly, constantly overtaking from the wrong side, going at a higher speed than nearby vehicles on highly congested roads, etc. come under unpredictable driving. This kind of behaviour by drivers increases the chances of accidents massively.
5. **Distracted driving:** This refers to a situation where a driver is trying to do multiple things while driving his vehicle. Texting on phone, talking to the co passengers, eating, changing media settings, etc. come under distracted driving.

The growth in the field of IoT in the recent years has laid down a pathway to create solutions to effectively tackle this problem of rash driving. With our proposed system, we look to make use of various data points collected with the help of a Sensory Network to understand driver/rider behaviour and their driving patterns. Our system classifies outputs based on prior data collected by sensors and thus to avoid anomalies or false positives (the reports of potential crashes in this case) an algorithm has been used which is ideal for this situation. We made use of the XGBoost (eXtreme Gradient Boosting) as this boosting algorithm has high rates of predicting true positives and true negatives, and a higher efficiency in general. This enabled the efficiency of the accuracy of the data points to increase to over 95%, when matched by our test dataset. Through XGBoost we were also able to validate the theoretical claims that boosting methods reduce both bias and variance, and the predictors in it keep learning from mistakes committed by previous predictors, thereby reducing the time consumed in the next iteration. For our model here, we make use of Gradient Boosting, which is a machine learning technique used for solving Regression and Classification problems.

Through numerous studies and our personal research on road accidents, we understand that a normal accident usually takes around 8 s to occur from realization to collision to impact. Another important aspect of our model is to make use of the current technologies to create a response mechanism which would be faster than any manual response. The continuous and comprehensive monitoring system in our model can pre-empt rash driving and delegate the security systems in place to be deployed and make the driver aware of the potential risk he is running. The system also helps the nearby drivers, who have the same system installed inside their vehicles, aware of the situation and thus prompting them to drive cautiously. The standalone system also helps the vehicle to slow down steadily in the 8 s period.

## 2 Literature Review

The introduction of motors and the subsequent development in the field of machinery and their components has certainly played a key role in improving the quality of life across the globe, but with the numerous perks it brings, there lie a few banes that come along. Several accidents occur daily due to the reckless behaviour of drivers on the road, which poses a danger not only to the driver of the vehicle, but also to the public. Abrupt change in speed, continuous changing of lanes, etc. cause rash driving and can be fatal in many situations. With people disobeying traffic rules and crossing speed limits, road safety becomes a matter of concern worldwide [3]. Road Safety has a direct impact on the economy and the general welfare of the people and is an important public health issue. Accidents caused due to road traffic lead to severe injuries and hospitalizations, which cost a lot to families, communities, and nations [4]. While we have witnessed a steady decrease in the road accident fatalities for the first-world countries, but the same cannot be said for the rest of the world's population, the pressure of the societal and economic costs caused due to road injuries, is rising substantially. The major factors behind road accidents are as follows:

1. **Road Defects:** The construction of speed breakers in unwanted places on the road leads to accidents. When roads are dug up and are not properly closed, it gives rise to crevices which cause traffic jams and as a result, accidents.
2. **Poor Lighting:** Lack of proper lighting on roads and highways reduces the visibility of drivers and can lead to fatal injuries.
3. **Lack of road signs:** Road signs act as an alert for pedestrians and motorists about speed limits, crossings, etc. and not putting up these signs in appropriate regions can be harmful for people on the road.
4. **Speeding:** Many drivers tend to exceed the speed limit, especially on the highway, and about one-third of road accidents are caused due to over speeding.
5. **Drunk Driving:** It is extremely dangerous to operate a vehicle when drunk due to blurred vision and one's inability to focus when intoxicated.

Several other reasons include faults in the vehicle, not stopping at red lights, excessive luggage in the vehicle, driving in the opposite lane, and many more. According to the World Health Organization (WHO), more than a million people lose their lives on the road every year and road traffic injuries are among the leading causes of death among youngsters. As it is evident that the current rules and regulations to prevent road accidents are not entirely effective, the concerned agencies need to implement a multi-disciplinary approach to deal with Road Traffic Accidents. Creating awareness among people, strict measures regarding following traffic rules, and engineering measures are an absolute necessity to overcome this public health catastrophe [5]. One way to detect rash driving is by analysing the behaviour of drivers. This is achieved by the following sensors present in smartphones:

1. Accelerometer: To measure the forces of acceleration in a vehicle, a sensor is required to be used. This sensor is called an accelerometer. A great example of such force can be seen by the tilting motion of a phone.
2. Gravity Sensor: As the name suggests, a Gravity sensor is a type of sensor used to indicate the direction in which the gravitational forces lie while also describing its magnitude at the same time.
3. Rotational Vector Sensor: The rotation vector represents the orientation of the device as a combination of an angle and an axis, in which the device has rotated through an angle around an axis (x, y or z).

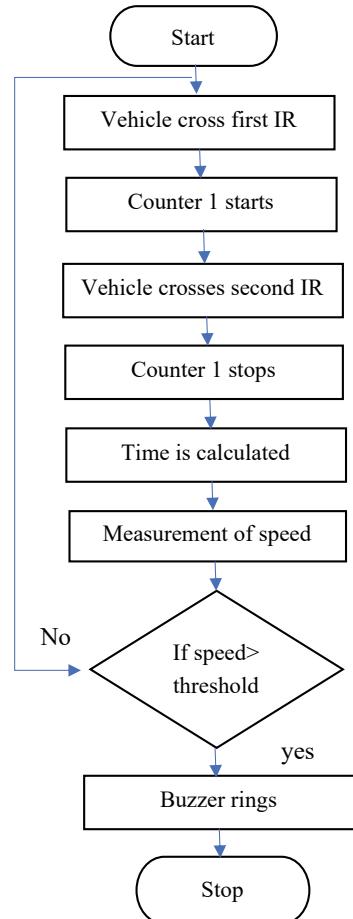
In this implementation, devices like infrared sensors, buzzers, and Arduino boards were used to create a small network which had the capability to observe, detect and alarm the users as well as the responsible authorities about any kind of rash driving patterns observed in their vicinity [6].

The two infrared sensors which are placed a fixed distance from each other, i.e. 30 cm, are connected to the small network in such a way that any vehicle crossing a certain defined speed limit generates an alert for the users. The flowchart below describes the process further.

While this system makes use of IoT, which is a cheap and reliable mechanism, this concept can be further integrated by installing a camera or a number plate recognition system. These images can be sent to the concerned authorities so that appropriate action can be taken.

Since thresholds could also be manipulated by the type of automotive and sensors' sensitivity, they are still unable to accurately differentiate between the variations in varied driving behavioural patterns. An approach should be proposed that makes use of sensors to find abnormal driving behaviours along with an additional feature to determine specific kinds of driving patterns while not requiring any extra hardware. If we can determine drivers' abnormal driving manner automatically, the drivers in question can be made aware of their dangerous driving habits, to improve these habits and surely save themselves from being involved in a fatal car accident. Furthermore, if the results of the observation might be passed back to a central server, the police can utilize them to observe inappropriate driving and take required actions [7]. To implement this approach, an accelerometer is used to get the reading, and provides the value of value of X, Y, and Z as per the motion of mobile. The reading shows whether the driver is driving rashly or not. In case of an accident, an SMS will be sent to an emergency number mentioned at the time of registration, and it will also notify all the passengers in the same area so that they can take a shorter route to their destination. It can also detect traffic by identifying user's speed. System collects the speed of vehicles in the same area, if all users drive slowly, it will be notified as traffic and these details will be sent to all travellers to avoid traffic. IoT in real world is playing an important role in the field of health care [8], health tracking [9], smart home and many more.

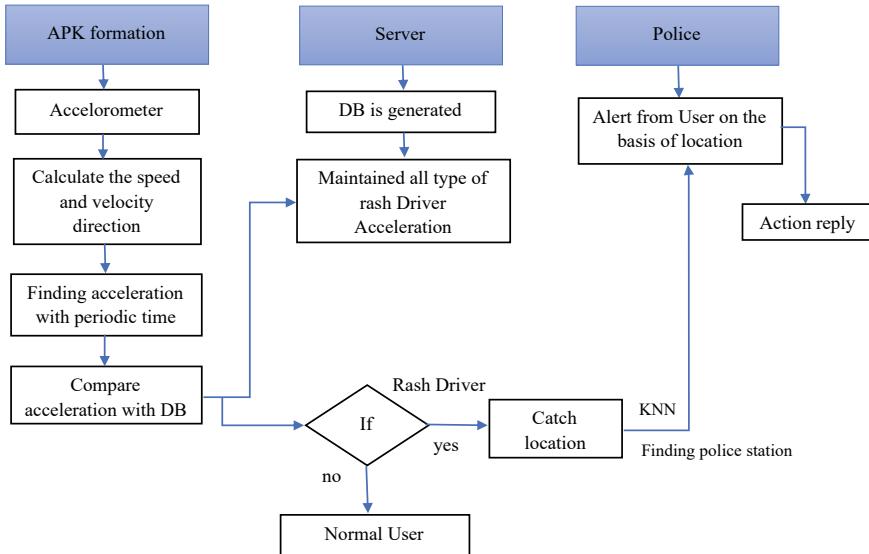
**Fig. 1** Flow chart of rash driving detection



Some related work in the field of driver behaviours prediction is showcased in [10–13] which uses various other form of datasets to study the behaviour of the driver in different conditions (Fig. 1). Some of the similar works using IoT are showcased in [14, 15].

### 3 Proposed Model

Our approach seeks to build a system which works on a continuous evaluation pattern focusing on safety for the driver. Figure 2 showcases the system architecture of the proposed model. Our main structure is to use Real-Time Automated Multiplexed Sensor System and Reverse Geocoding for Rash Driving Detection based on Driver Behaviour using a plush range of innovative tools (Fig. 3).



**Fig. 2** System architecture

### 3.1 Methodology

To create our standalone model, we proceeded in a phase-wise manner:

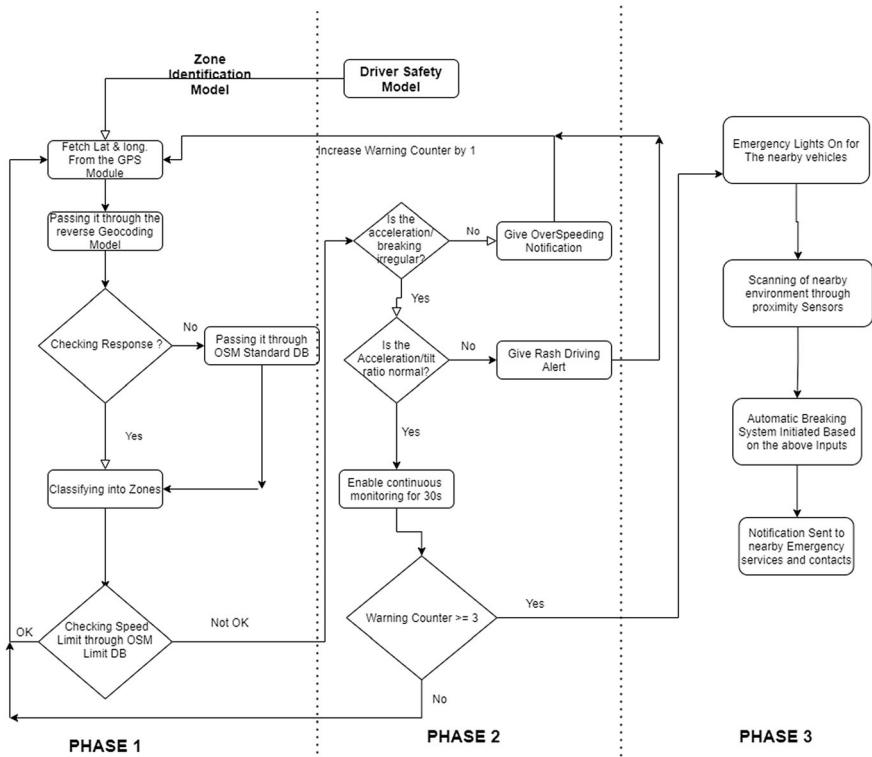
#### 1. Phase 1

- In this phase, we first focus on using the Reverse Geocoding System that helps in providing the location based on the latitude and longitude fed into the system.
- Following this, a Zone-wise Classification of the Latitude and the Longitude is done to categorize the area into different zones like Rural, Urban, Open Road, Highway, etc.
- Now based on what zone the vehicle is in, we use the OSM/General Speed Limit set for the area to create a Speed Limit Display.

Once all the steps in phase 1 were implemented, we move to the next phase of the model creation.

#### 2. Phase 2

- Reverse Geocoding using Google API: At this stage, a JS based script was created which had an input box for Latitude and Longitude.
- The output returned when we put values in the input box is equal to the geolocation identified by Google Maps, corresponding to the Latitude and Longitude.



**Fig. 3** Implementation

- Finally, we were able to create tags based on these geolocation markers, which denoted the Type of Area and the Speed Limit at that spot.

Once both the phases are successfully implemented, this system can use the GPS system of that vehicle to feed the latitude and longitude of its current location and get the deck of information which tells them the required speed vs the current speed, the road type, etc.

When this foundational system is combined with the next phase that involves the usage of multiple sensors to detect variants like temperature, road conditions, lighting, etc., we create a robust model that can be used to detect any kind of improper driver behaviour and alert the nearby vehicles, saving lives and making the roads a lot safer.

### 3.2 Convolution Versus Cross-Correlation

Cross-correlation and convolution are topics which people tend to mix up quite a few times with them taking one for the other. While describing cross-correlation, we need to understand that the major differentiating factor between the two is the fact that the kernel flipping is not done in cases of cross-correlation while the same cannot be said for convolution. The difference in the equations between the two can be seen below:

Convolution:

$$y[m, n] = X[m, n] * h[m, n] = \sum_{j=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} X[i, j] \cdot h[m - i, n - j] \quad (1)$$

Cross-Correlation:

$$y[m, n] = X[m, n] \otimes h[m, n] = \sum_{j=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} X[i, j] \cdot h[m + i, n + j] \quad (2)$$

Equations (1) and (2). Convolution versus Cross-Correlation

The difference between  $h[m - i, n - j]$  or  $h[m + i, n + j]$  convolution and cross-correlation respectively is what decides if we kernel is sliding or flipping. This is also the major factor behind deciding whether an element's pixel will be processed or not in the final output map that is generated.

### 3.3 Development Environment

*Phase-1*

*Google Cloud Console*: For Generating API Key and reverse geocoding.

*HTML, CSS and JS*: For creating a standalone system which returns the location basis latitude and longitude; and provides a general speed limit based depending on the area—whether rural/urban/highway.

*Phase-2*

*Carla Model Simulator*: For simulating our vehicle, as an autonomous driving system.

*Kaggle*: Functions as an online collaboration notebook + compiler for the code.

*Town03 Model*: Gives the necessary sensor-based information.

*XGBoost*: For trial-error; to experiment with various algorithms.

*Pandas, Matplotlib, SciPy*: Libraries to help run the final algorithm chosen, K-nearest neighbours' algorithm.

### Phase-3

*WebStorm*: As an IDE.

*WebSocket and API*: For linking and connectivity.

*Google Cloud Console*: Functions as a host solution.

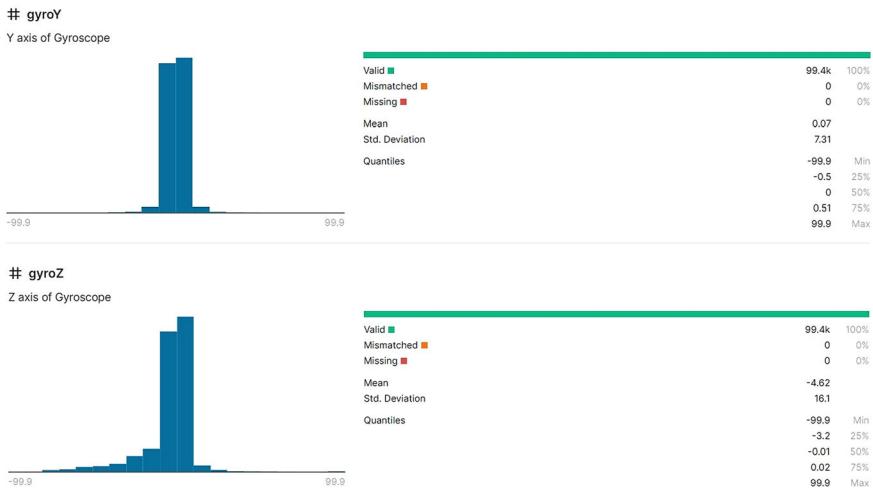
### 3.4 Challenges

There is no doubt that a lot of work and capital has been invested in improving road safety in the 21st Century, still, the sheer number of people we lose to road accidents every year remains a big challenge. With various advancements in technology, there always exist opportunities for every researcher out there to explore and help make the roads safer. Currently, some of the opportunities for them to explore include, include:

1. **Weather conditions and Environment**—The weather conditions at a place play a huge role in determining the driving patterns of any driver as a person driving under normal conditions is going to differ from someone driving in rain or in a snow lodged area.
2. **Crowd sourcing**—Crowd sourcing data from multiple vehicles helps researchers gain a perspective on what the general driving patterns are.
3. **Road Conditions**—Conditioning of the roads is important to understand as a driver driving on a well-constructed road will have different driving pattern to some driving on a road which is in a poor condition.
4. **Anonymization**—With the use of sensitive user data involved in the process of understanding driver behaviour pattern, the researchers must find a way to ensure that this data which includes tracking the regular location of the drivers, is not breaching the individual's privacy by finding a way to anonymize any driving pattern data they receive.
5. **Sensor fusion**—When we talk about increasing the efficiency of any system, one thing that must be remembered is that reports generated by a network of sensors are going to be more accurate than that generated by one so ensuring that a network of sensors are used to understand the patterns is also important.
6. **Virtual Reorientation**—Different people have different tendencies while driving. One of them is that they can keep their mobile devices in multiple different orientations, sometimes not even noticing it. This can lead to an accurate report at times. Hence, a technique to realign the axis of the device virtually with the vehicle's axis must be put in place.

## 4 Results

Multiple features like a Gyroscope, Accelerometer, etc. were used to enable different capabilities in the sensory system, here is a look into a few of them (Figs. 4 and 5).



**Fig. 4** Insight into the gyroscope feature

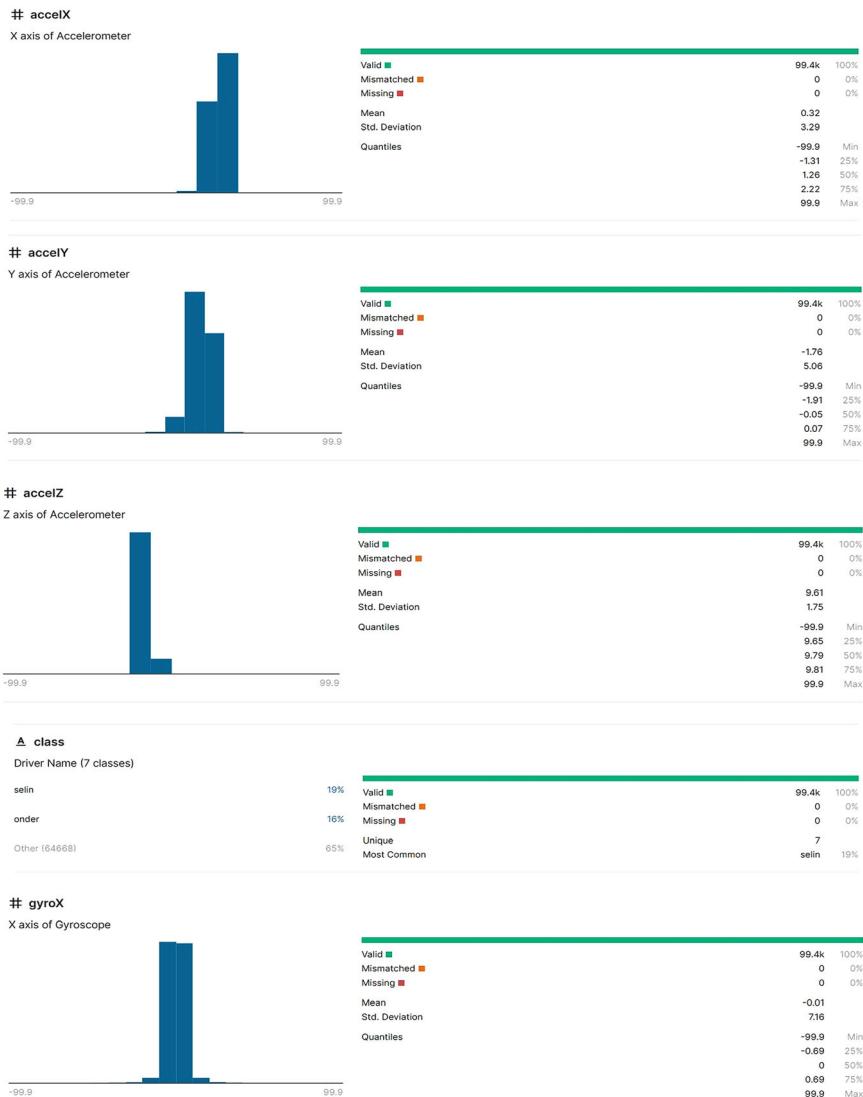
After defining the required parameters for the different sensors used in the system, when the code was run, the following results were obtained defining the accuracy for SVM, Bayes, SGD, and KNN :

```
SVM accuracy is: 0.8084951727736981
accuracy of bayes in test data is: 0.6251351250605733
acc_of_sgd is: 0.6648898497782085
acc_knn: 0.9998508964848847
```

This tells us that the model is 99.83% accurate, or in other words, the chance of wrong prediction is less than 0.2%. We had earlier tried our hand with XGBoost, where we had reached a decent accuracy of over 95%, however k-KNN proved to be a blessing since our model shot to over 99% of accuracy (Figs. 6 and 7).

## 5 Major Findings

With an accuracy percentage of 99.83%, and a relatively low cost for making the device available for regular users, our model can play a key role in decreasing instances of road accidents caused due to rash driving. This proved that using knn was the right approach and after providing results which were better than what XGBoost provided. With this level of accuracy, the prediction of driver behaviour can be done better than the systems that exist today. With multiple sensors attached to the system like temperature, pressure, humidity, accelerator, etc. we can now try to accurately predict the reason behind a driver driving in a rash way.



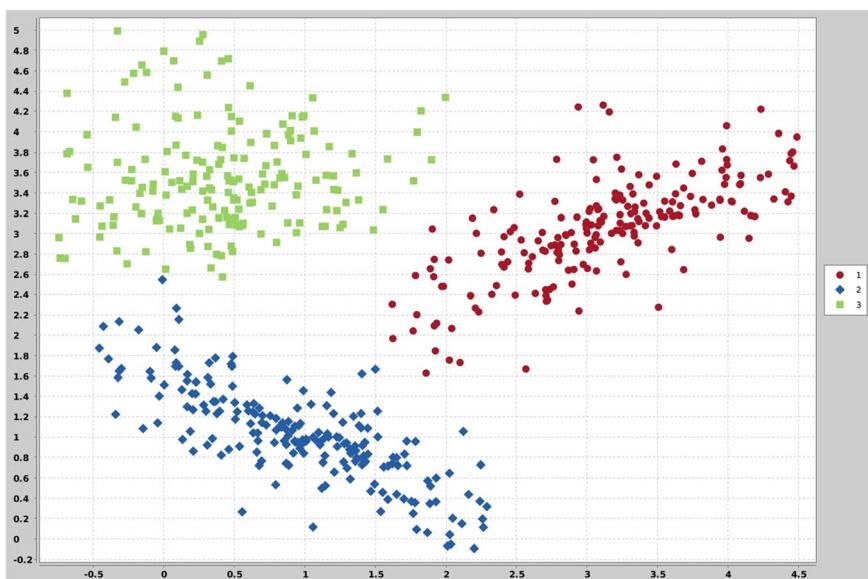
**Fig. 5** Insight into other features

## 6 Future Scope

Although the model is robust at the current time, there needs to be constant research and development which needs to be done in the future as well to make it even more feasible than it is today. With that in mind, here are a few phases where changes can be made:



**Fig. 6** Cross-correlation for all features and saved to “data\_prossed”



**Fig. 7** Sample KNN graph

1. As the use of a black box in airplanes has been a norm for decades, this system can also be developed further in such a way that it can be used as not only a way to understand driver behaviour but also as a system that helps us understand reasons behind every accident that takes place. Since the data upload to a cloud server is a real-time process, it can become a useful system which must be present in every vehicle that runs on roads.
2. Through proper research and testing with other upcoming or existing models, the level of accuracy of the results can be improved to provide near perfect results.

3. This system can also be used by driving schools and their instructors to understand a novice driver, his driving patterns, and the mistakes that he makes on the road. Using this data, he can try providing them with the correct training, such that they don't become a threat on the road when they eventually get their license.
4. The data points collected through this system can also be used by the governments to make changes to the roads, improve the traffic situations, and work on building a more suitable road infrastructure for a particular area.
5. The traffic authorities can also use the speedometer and location tracker data collected by this system to ensure that people follow the necessary rules and speed limits in an area and any infringements can lead to a direct penalty for the driver which he has to pay online, instead of them running after these drivers, risking their own lives in the process.

## 7 Conclusion

Safety in fact, can never be stressed enough as a topic. There are multiple ways of improving the measures being used to save people from being involved in fatal road accidents. From managing and understanding a driver's driving pattern, to demarcating accident prone regions and their reasons, to create a record library for learning from past events, a lot can be done to ensure we don't lose our loved ones in any such events. With the introduction of smart devices, the aspect of monitoring each movement with the help of specifically designed sensory devices has become easier than ever and the relatively cheap price along with increased accessibility features has ensured that a large amount of people already own such devices, and these are the devices which play major role in understanding the key aspects of driving pattern of any driver. This paper tries to do exactly that by making use of the sensors and using several techniques of detecting cases of poor driving and driving errors being made regularly by an individual. The rash driving alert system can be used with accelerometer sensor to distinguish any unusual or risky driving move. In this, the device will collect and analyse the data from its accelerometer sensor to recognize any risky condition and then it will send email to the user. The rash driving recognizing techniques can be provided along with the sensors and the techniques can be useful along in the roadside units.

Our paper will be useful for both rash driving and over speeding of vehicles detection. No module is currently available which consumes such a low power to detect rash driving and our project provides that facility. It has various sensors such as pressure, temperature, humidity sensor, and accelerator built within the sensor hub which can further be extended to control the speed and rash driving in humid climate and detect the overheating of vehicles. The inbuilt sensor also in result reduces the size of the system to a great level.

## References

1. <https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries>
2. <https://loconav.com/vahan/traffic-fines/rash-and-negligent-driving-section-279-ipc/>
3. Prabhu MR et al A review on rash driving detection and alert system
4. Bhat YR (2016) Reasons and solutions for the road traffic accidents in India. *Int J Innov Technol Res* 4(6):4985–4988
5. Gopalakrishnan S (2012) A public health perspective of road traffic accidents. *J Fam Med Prim Care* 1(2):144
6. Patel L, Gaurav A (2018) Detection of rash driving on highways
7. Rash driving detection. *Int J Emerg Technol Innov Res* 5(5):853–859. ISSN: 2349-5162. [www.jetir.org](http://www.jetir.org)
8. Gupta P, Agrawal D, Chhabra J, Dhir PK (2016, March) IoT based smart healthcare kit. In: 2016 International conference on computational techniques in information and communication technologies (ICTICT). IEEE, pp 237–242
9. Gupta P, Gaur N, Tripathi R, Goyal M, Mundra A (2020, November) IoT and cloud based healthcare solution for diabetic foot ulcer. In: 2020 sixth international conference on parallel, distributed and grid computing (PDGC). IEEE, pp 197–201
10. Abdelrahman AE, Hassanein HS, Abu-Ali N (2020) Robust data-driven framework for driver behavior profiling using supervised machine learning. *IEEE Trans Intell Transp Syst* 23(4):3336–3350
11. Zfnebi K, Souissi N, Tikito K (2017, May) Driver behavior quantitative models: identification and classification of variables. In: 2017 International symposium on networks, computers and communications (ISNCC). IEEE, pp 1–6
12. Dai S, Zhong Y, Xu C, Liu H, Yuan J, Wang P (2022) An intelligent security classification model of driver's driving behavior based on V2X in IoT networks. *Secur Commun Netw*
13. Hou M, Wang M, Zhao W, Ni Q, Cai Z, Kong X (2022) A lightweight framework for abnormal driving behavior detection. *Comput Commun* 184:128–136
14. Sethi P, Juneja L, Gupta P, Pandey KK (2018) Safe sole distress alarm system for female security using IoT. In: Proceedings of first international conference on smart system, innovations and computing: SSIC 2017, Jaipur, India. Springer Singapore, pp 863–874
15. Mohammed K, Abdelhafid M, Kamal K, Ismail N, Ilias A (2023) Intelligent driver monitoring system: an internet of things-based system for tracking and identifying the driving behavior. *Comput Stand Interfaces* 84:103704

# Author Index

## A

- Abhishek Tiwari, 119  
Abirami Gurushanker, 251  
Aditya Shah, 179  
Ajeet Singh, 333  
Amey Jojare, 319  
Ananya Debnath, 67  
Aniket Kumar, 319  
Anil Kumar Dubey, 411  
Anuraj Mohan, 153  
Aparna Padma Balaji, 139  
Arati Chabukswar, 55  
Archana Singh, 285  
Arnaav Anand, 437  
Athulya Valsan, T. P., 153  
Avijit Das, 79

## B

- Benoy Joseph, 297  
Bramah Hazela, 67, 91

## C

- Chaitanya Pushkarna, 119  
Chandas Patel, C. I., 55  
Chetashri Bhadane, 179  
Chinmaya Bikram Pattanaik, 399  
Clara Joseph, 227

## D

- Dang, Khoa Nguyen, 377  
Dat, Dong Quoc, 367  
Devam Patel, 1  
Dev Bhut, 35

Dilkeshwar Pandey, 345

Durga Sharma, 103

## E

- Elakiya, E., 297

## G

- Gade Sai Panshul, 165  
Gaurav Kumar Gautam, 239  
Gautam Mehendale, 179  
Gupta, Punit, 425, 437

## H

- Hareendra Sri Nag Nerusu, 165  
Harshit Gupta, 239  
Harsh Vardhan, 239

## I

- Ishita Mehta, 437

## J

- Jahan, Fahum Nufikha, 389  
Janaki Meena Murugan, 251, 263  
Jayant Sasikumar, 139  
Jitesh Choudhary, 119  
Johnstone Joel Ngorma, 189  
Jordan-Kény Gnansounou Dansi, 189  
Jui Mehta, 1

## K

- Kajal Singh, 91

Kashish Gandhi, 35  
 Kavya Suresh, 139  
 Kien, Nguyen Phan, 357, 367  
 Kolhe, Mohan Lal, 399, 411

**L**

Leki Chom Thungon, 297  
 Lekshmi Kalinathan, 251, 263  
 Lingutla Prem Kumar, 165

**M**

Mahmud, Shakik, 389  
 Manan Gandhi, 1  
 Manas Kamal Das, 297  
 Manish Raj, 25  
 Mansi Prajapati, 35  
 Marimuthu, M., 13, 251  
 Md. Kaish, 189  
 Minh, Pham Tuan, 377  
 Mirudhula Loganath, 251  
 Monika, 25  
 Mrinal Manna, 79  
 Munesh Chandra Trivedi, 399  
 Muthukuru Jayanth, 13

**N**

Nagendra Singh, 119  
 Naiyya Mittal, 273

**P**

Pijush Ghorai, 201  
 Pradeep Singh Rawat, 425  
 Pratap, M. S., 55  
 Prateek Kumar Soni, 425  
 Pratyush Mishra, 189  
 Praveen Sundra Kumar, N., 129  
 Priyanka Tiwari, 119

**R**

Rahul Johari, 103  
 Rahul Karmakar, 79  
 Rahul Katarya, 239  
 Rahul Kumar, 263  
 Rajayshree Bhattacharyaa, 79  
 Ramakrishnan, S., 129  
 Ravikumar Pandi, V., 139  
 Rishi Joshi, 1  
 Rishima Chowdhury, 251  
 Ruchi Jain, 399

Ruchi Tiwari, 119  
 Ruhina Karani, 35  
 Rupali Mahajan, 333  
 Rupashri Barik, 201  
 Rutika Babasab Patil, 55

**S**

Sagar Mondal, 251  
 Sahla Ambrein, 189  
 Sanjay Singh, 333  
 Sankari Karthik, 251  
 Santosh Kumar Satapathy, 1  
 Saravanan Palani, 13, 251  
 Sarthak Agarwal, 345  
 Sejal Maheshwari, 411  
 Shelley Gupta, 285  
 Shikha Singh, 67, 91  
 Shobha Sharma, 273  
 Shubham Kumar Gupta, 345  
 Siam, Md Kamrul, 389  
 Siddesh Sabade, 319  
 Siddharth Menon, 139  
 Siddhi Muni, 179  
 Sidhartha Bakuli, 79  
 Siji Rani, S., 165  
 Soni Singh, 189  
 Soumya Sathyan, 139  
 Srishty Sharma, 273  
 Sruthy Manmadhan, 227  
 Sunaina Singh, 189  
 Swapnika Agrawal, 411  
 Swati Kale, 319  
 Swimpy Pahuja, 55

**T**

Tathipamula Harini Sai, 165  
 Thao Van, Hoang, 357  
 Thazhai Mugunthan, 139  
 Tithi Pandey, 273  
 Tran, Duc-Tan, 357, 367, 377  
 Tripti Mishra, 25

**V**

Vanshaj Singhal, 25  
 Vanshita Patel, 1  
 Vignesh, M., 129  
 Vijender Kumar Solanki, 357, 367, 377  
 Vineet Singh, 67, 91  
 Vipina Valsan, 139

**Y**

Yashaswat Verma, [25](#)  
Yash Garg, [345](#)

Yeshas, R., [55](#)

Yordanova, Zornitsa, [215](#)