

EAI/Springer Innovations in Communication and Computing

Anandakumar Haldorai  
Arulmurugan Ramu  
Sudha Mohanram *Editors*

# 5th EAI International Conference on Big Data Innovation for Sustainable Cognitive Computing

BDCC 2022

 **EAI**  
RESEARCH MEETS INNOVATION

 Springer

# **EAI/Springer Innovations in Communication and Computing**

## **Series Editor**

Imrich Chlamtac, European Alliance for Innovation, Ghent, Belgium

The impact of information technologies is creating a new world yet not fully understood. The extent and speed of economic, life style and social changes already perceived in everyday life is hard to estimate without understanding the technological driving forces behind it. This series presents contributed volumes featuring the latest research and development in the various information engineering technologies that play a key role in this process. The range of topics, focusing primarily on communications and computing engineering include, but are not limited to, wireless networks; mobile communication; design and learning; gaming; interaction; e-health and pervasive healthcare; energy management; smart grids; internet of things; cognitive radio networks; computation; cloud computing; ubiquitous connectivity, and in mode general smart living, smart cities, Internet of Things and more. The series publishes a combination of expanded papers selected from hosted and sponsored European Alliance for Innovation (EAI) conferences that present cutting edge, global research as well as provide new perspectives on traditional related engineering fields. This content, complemented with open calls for contribution of book titles and individual chapters, together maintain Springer's and EAI's high standards of academic excellence. The audience for the books consists of researchers, industry professionals, advanced level students as well as practitioners in related fields of activity include information and communication specialists, security experts, economists, urban planners, doctors, and in general representatives in all those walks of life affected ad contributing to the information revolution.

**Indexing:** This series is indexed in Scopus, Ei Compendex, and zbMATH.

**About EAI** - EAI is a grassroots member organization initiated through cooperation between businesses, public, private and government organizations to address the global challenges of Europe's future competitiveness and link the European Research community with its counterparts around the globe. EAI reaches out to hundreds of thousands of individual subscribers on all continents and collaborates with an institutional member base including Fortune 500 companies, government organizations, and educational institutions, provide a free research and innovation platform. Through its open free membership model EAI promotes a new research and innovation culture based on collaboration, connectivity and recognition of excellence by community.

Anandakumar Haldorai • Arulmurugan Ramu •  
Sudha Mohanram  
Editors

# 5th EAI International Conference on Big Data Innovation for Sustainable Cognitive Computing


BDCC 2022

 Springer

 **EAI**  
RESEARCH MEETS INNOVATION



*Editors*

Anandakumar Haldorai   
Department of Computer Science and  
Engineering  
Sri Eshwar College of Engineering  
Coimbatore, Tamil Nadu, India

Arulmurugan Ramu  
Department of Computer Science and  
Engineering  
CMR University  
Bengaluru, Karnataka, India

Sudha Mohanram  
Sri Eshwar College of Engineering  
Coimbatore, Tamil Nadu, India

ISSN 2522-8595

ISSN 2522-8609 (electronic)

EAI/Springer Innovations in Communication and Computing

ISBN 978-3-031-28323-9

ISBN 978-3-031-28324-6 (eBook)

<https://doi.org/10.1007/978-3-031-28324-6>

© European Alliance for Innovation 2023

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

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

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

This Springer imprint is published by the registered company Springer Nature Switzerland AG  
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

# Preface

We are delighted to introduce the proceedings of the fifth edition of European Alliance for Innovation (EAI) International Conference on Big Data Innovation for Sustainable Cognitive Computing (BDCC 2022). This conference has brought researchers, developers and practitioners around the world who are leveraging and developing Big Data technology for a smarter and more resilient data computing.

The technical program of BDCC 2022 consisted of 18 full papers under oral presentation session at the main conference tracks. Aside from the high-quality technical paper presentations, the technical program also featured a keynote designed for professionals working in the early stages of building an advancement program, as well as those with more mature operations. The keynote was Dr. I. Jeena Jacob, Associate Professor, Department of Computer Science and Engineering, GITAM University-Bengaluru Campus.

Coordination with the steering chair Dr. Imrich Chlamtac and Dr. Anandakumar Haldorai was essential for the success of the conference. We sincerely appreciate their constant support and guidance. It was also a great pleasure to work with such an excellent organizing committee for their hard work in organizing and supporting the conference. In particular, the Technical Program Committee, led by our TPC Chair Dr. Arulmurugan Ramu, Publication Chair Prof. M. Suriya, Local Committee Chairs Prof. K. Karthikeyan and Dr. K. Aravindhan, have completed the peer-review process of technical papers and made a high-quality technical program.

We are also grateful to our Conference Manager Mr. Mikita Yelnitski and Publication and Managing Editor Ms. Eliska Vickova for their continuous support and guidance. We thank all the authors who submitted and presented their papers to the BDCC 2022 conference and workshops.

We strongly believe that BDCC 2022 conference has provided a good forum for all researcher, developers and practitioners to discuss all science and technology aspects that are relevant to Big Data technology. We also expect that the future

BDCC 2023 conference will be as successful and stimulating, as indicated by the contributions presented in this series.

Coimbatore, Tamil Nadu, India  
Bengaluru, Karnataka, India  
Coimbatore, Tamil Nadu, India

Anandakumar Haldorai  
Arulmurugan Ramu  
Sudha Mohanram

# Conference Organization

## Steering Committee

|                          |  |
|--------------------------|--|
| Imrich Chlamtac          | Bruno Kessler Professor, University of Trento, Italy             |
| Dr. Sudha Mohanram       | Sri Eshwar College of Engineering, Coimbatore, India             |
| Dr. Anandakumar Haldorai | Sri Eshwar College of Engineering, Coimbatore, Tamil Nadu, India |
| Dr. Arulmurugan Ramu     | CMR University, Bangalore, India                                 |

## Organizing Committee

### *General Chair*

|                          |  |
|--------------------------|--|
| Dr. Anandakumar Haldorai | Sri Eshwar College of Engineering, Coimbatore, India |
|--------------------------|--|

### *TPC Chair*

|                      |  |
|----------------------|--|
| Dr. Arulmurugan Ramu | CMR University, Bangalore, India                     |
| Prof. Suriya Murugan | Sri Eshwar College of Engineering, Coimbatore, India |

### *Workshops Chair*

|                    |   |
|--------------------|---|
| Dr. Aravindhana K. | SNS College of Engineering, Coimbatore, India |
|--------------------|---|

*Publicity & Social Media Chair*

|                  |  |
|------------------|--|
| Dr. Akshaya V.S. | Sri Eshwar College of Engineering, Coimbatore, India |
|------------------|--|

*Publications Chair*

|                      |  |
|----------------------|--|
| Prof. Suriya Murugan | Sri Eshwar College of Engineering, Coimbatore, India |
|----------------------|--|

*Web Chair*

|                      |   |
|----------------------|---|
| Prof. Karthikeyan K. | SNS College of Engineering, Coimbatore, India |
|----------------------|---|

*Local Chair*

|                    |  |
|--------------------|--|
| Prof. Sivakumar K. | National Institute of Technology, Karnataka, India |
|--------------------|--|

*Conference Manager*

|                      |     |
|----------------------|-----|
| Mr. Mikita Yelnitski | EAI |
|----------------------|-----|

**Technical Program Committee**

|                              |  |
|------------------------------|--|
| Dr. Syed S.A. Rehmansarehman | University, Xian, China                            |
| Dr. Debabrata Datta          | GLA University, India                              |
| Prof. Roshini Arumugam       | KLN University, Hyderabad India                    |
| Dr. Lasith Gunawardena       | University of Sri Jayewardenepura, Sri Lanka       |
| Dr. Chow Chee Onn            | University of Malaya, Malaysia                     |
| Dr. Chan-Yun Yang            | National Taipei University, Taiwan                 |
| Dr. Shahram Rahimi           | Southern Illinois University, Illinois, USA        |
| Dr. Hooman Samani            | University of Plymouth, UK                         |
| Dr. Baskaran K.R.            | Kumaraguru College of Technology, Coimbatore       |
| Dr. Umamaheswari K.          | PSG College of Technology, Tamil Nadu, India       |
| Dr. Gokuldev S.              | PSG College of Arts and Science, Coimbatore, India |
| Dr. Thillai Arasu            | REVA University, India                             |

# Contents

|   |           |
|---|-----------|
| <b>Part I Bigdata Services and Analytical Database</b>  |           |
| <b>Enhanced Dense Layers Using a Quadratic Transformation Function ....</b>   | <b>3</b>  |
| Atharva Gundawar and Srishti Lodha  |           |
| <b>Analysis of Metaheuristic Algorithms for Optimized Extreme Learning Machines in Various Sectors .....</b>                | <b>17</b> |
| D. Devikanniga and D. Stalin Alex   |           |
| <b>Metal and Metal Oxide Nanoparticle Image Analysis Using Machine Learning Algorithm.....</b>                              | <b>27</b> |
| Parashuram Bannigidad, Namita Potraj, and Prabhuodeyara Gurubasavaraj   |           |
| <b>Efficient Implementation to Reduce the Data Size in Big Data Using Classification Algorithm of Machine Learning.....</b> | <b>39</b> |
| V. RajKumar and G. Priyadharshini   |           |
| <b>Tracer for Estimation of the Data Changes Delivered and Permalinks ....</b>  | <b>55</b> |
| N. H. Prasad, S. Kavitha, Laxmi Narayana, and G. R. Sanjay  |           |
| <b>Part II Bigdata and Privacy Preserving Services</b>  |           |
| <b>Design and Development of a Smart Home Management System Based on MQTT Incorporated in Mesh Network .....</b>            | <b>65</b> |
| Andrea Antony, Nishanth Benny, Gokul G. Krishnan, Mintu Mary Saju, P. Arun, and Shilpa Lizbeth George                       |           |
| <b>Novel Machine-Learning-Based Decision Support System for Fraud Prevention.....</b>                                       | <b>75</b> |
| Norman Bereczki, Vilmos Simon, and Bernat Wiandt  |           |
| <b>Big Data Challenges in Retail Sector: Perspective from Data Envelopment Analysis .....</b>                               | <b>89</b> |
| Praveen M. Kulkarni, Prayag Gokhale, and Padma S. Dandannavar   |           |

### **Part III Bigdata and Data Management Systems**

|  |            |
|--|------------|
| <b>Restoration of Ancient Kannada Handwritten Palm Leaf Manuscripts Using Image Enhancement Techniques .....</b> | <b>101</b> |
| Parashuram Bannigidad and S. P. Sajjan   |            |

|   |            |
|---|------------|
| <b>Mutli-Label Classification Using Label Tuning Method in Scientific Workflows .....</b> | <b>111</b> |
| P. Shanthi, P. Padmakumari, Naraen Balaji, and A. Jayakumar                               |            |

|  |            |
|--|------------|
| <b>A Comparative Analysis of Assignment Problem .....</b>  | <b>125</b> |
| Shahriar Tanvir Alam, Eshfar Sagor, Tanjeel Ahmed, Tabassum Haque, Md Shoaib Mahmud, Salman Ibrahim, Ononya Shahjahan, and Muhtasim Rubaet |            |

### **Part IV Bigdata in Medical Applications**

|  |            |
|--|------------|
| <b>A Survey on Memory Assistive Technology for Elderly .....</b> | <b>145</b> |
| N. Shikha and Antara Roy Choudhury                               |            |

|   |            |
|---|------------|
| <b>An Experimental Investigation on the Emotion Recognition Using Power Spectrum Density and Machine Learning Algorithms in EEG Signals .....</b> | <b>157</b> |
| Nirmal Varghese Babu and E. Grace Mary Kanaga   |            |

|   |            |
|---|------------|
| <b>Detection and Classification of Pneumonia and COVID-19 from Chest X-Ray Using Convolutional Neural Network .....</b> | <b>173</b> |
| L. Swetha Rani, J. Jenitta, and S. Manasa   |            |

### **Part V Bigdata in Future Advancements**

|   |            |
|---|------------|
| <b>Stopwords Aware Emotion-Based Sentiment Analysis of News Articles...</b> | <b>183</b> |
| Chhaya Yadav and Tirthankar Gayen   |            |

|   |            |
|---|------------|
| <b>An Empirical Study to Assess the Factors Influencing Banking Customers Toward FINTECH Adoption in Tamil Nadu .....</b> | <b>195</b> |
| R. Mary Metilda and S. D. Shamini   |            |

|   |            |
|---|------------|
| <b>Driver's Drowsiness Detection Using SpO2 .....</b>                     | <b>207</b> |
| P. Sugantha Priyadharshini, N. Jayakiruba, A. D. Janani, and A. R. Harini |            |

|   |            |
|---|------------|
| <b>A Blockchain Framework for Investment Authorities to Manage Assets and Funds .....</b> | <b>217</b> |
| P. C. Sherimon, Vinu Sherimon, Jeff Thomas, and Kevin Jaimon                              |            |

|                    |            |
|--------------------|------------|
| <b>Index .....</b> | <b>227</b> |
|--------------------|------------|

**Part I**  
**Bigdata Services and Analytical Database**



# Enhanced Dense Layers Using a Quadratic Transformation Function



Atharva Gundawar  and Srishti Lodha 

## 1 Introduction

Dense layers, consisting of neurons, are the building blocks of any deep learning architecture. Neurons compute output as a sum of the following two features: (1) the summation of the product of the output vectors or logits from the previous layers with the corresponding trainable weight vector and (2) a trainable linear bias (refer Fig. 1).

Here, the function “ $f$ ” refers to the selected activation function, “ $b$ ” refers to the bias, “ $x_i$ ” refers to input to the neuron, and “ $w_i$ ” refers to the weight assigned to that neuron.  $X$  and  $Y$  are the input and output vectors of the neuron, respectively.

The numerous developments in deep neural network architectures, including techniques like dropout [1] and pruning, have helped overcome problems like exploding gradients and biased graphs. Some models involve skip connections (e.g., ResNet [2]), while some contain parallel paths (like InceptionNet [3]). While the difference between these models lies in the arrangement of layers, connections, paths traced by the logits, and so on, the underlying transformation function still remains the same. Hence, by changing this computation, every neuron reflects a minor change. Combined, all the neurons greatly impact the final result of the model.

Numerous mathematical functions can be explored to replace the linear function that calculates the output of a neuron. Within the scope of this paper, we build and test a simple dense neural network with a quadratic transformation function. The output is now a sum of the summation of the products between the corresponding trainable weight vectors firstly with the input and secondly with the input squared, and the trainable linear bias. This equation is presented in Fig. 2 (“ $w_i$ ” and “ $w_j$ ” refer to the weights assigned to that neuron.). We acquired four popular datasets to train

---

A. Gundawar · S. Lodha (✉)  
Vellore Institute of Technology, Vellore, Tamil Nadu, India

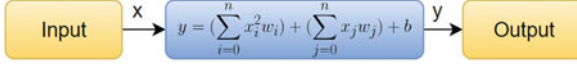


Fig. 1 Linear transformation function in the conventional perceptron

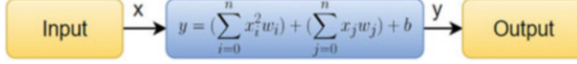


Fig. 2 Quadratic transformation function in the proposed perceptron

and evaluate the model. On comparing the results that conventional neurons [4] and new neurons yielded from a simple architecture with convolutional and dense layers, we observe an improved accuracy on any fixed number of epochs. Analysis of this improvement produces notable results, which are discussed later in this study.

The remaining paper is organized as follows. We conduct a literature review of closely related research and highlight their major contributions and drawbacks (Sect. 2). In Sect. 3, we discuss in depth, the methodology adopted for the implementation of this study. Finally, we move to the results and analysis in Sect. 4, followed by the conclusion (Sect. 5) and the references.

## 2 Literature Review

In this section, several related studies have been reviewed, and their contributions have been highlighted. We also identified certain drawbacks in these papers.

In H. Lin et al. [5], a universal approximation method was implemented by copying the Resnet structure, but with only one hidden neuron in alternating dense layers. This neuron presented a very high-order function, which was a representation of the combination of all neurons from a conventional hidden dense layer. Over-parametrization was successfully reduced and a universal approximation theorem for Resnet was implemented. However, this approach does not perform better than the pre-existing Resnet architecture in terms of accuracy in classification tasks. In F. Fan et al. [6], a successful autoencoder architecture was made using convolutional layers and quadratic dense functions, which replace the traditional single-order dense layers. The paper achieved the best results numerically and clinically in the dataset cited in the paper. As this works very well on the targeted dataset, we have no information if these results are translatable to other datasets and architectures as well.

V. Kůrková et al. [7] used shallow sigmum perceptron to achieve a lower bound on errors in approximation. In this probabilistic approach, the authors have shown that lower bounds on errors can be derived from the total number of neurons on finite domains. Not only is the proposal restrictive to certain domains, but unless a minimum threshold of sigmum perceptron is present in a given layer, the

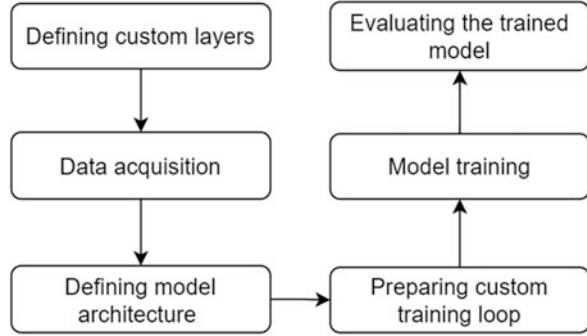
approximation of errors in both binary-valued and real-valued functions fails to give reliable results. In another study, C. L. Giles and Y. Maxwell [8] presented a new system of storing knowledge in higher-order neural networks with the use of priori knowledge. This system resulted in a model which attained a higher accuracy on multiple datasets and handled outliers better than the models used in the comparative study. However, having a priori knowledge system for any dataset in the real world is improbable, and the increase in accuracies does not always make up for the high model size.

S. Du and J. Lee [9] found a strong relationship between the number of hidden nodes activated by a quadratic function and the number of training examples. The theory of Rademacher complex was used to show how a trained model generalizes. Further research in terms of how the local search algorithm using over-parametrization finds very close global minimas can be conducted.

Similar to our study, F. Fan et al. [10] introduced a new type of neuron to replace the original dense layer neuron. Although it also has an order of 2, the function is a sum of two terms entirely different from what is proposed in this study. Instead, it is a summation of: (a) the product of the outputs of the conventional transformation function with a different set of weights for the same input “ $x$ ” and (b) the output of the conventional transformation function but with the input squared. These 2nd order neurons worked well in solving low complexity tasks, like fuzzy logic problems, and representing basic logic gates like “and,” “or,” “nand,” and “nor,” but the research fails to put some light on the working of this principle on multi-layer neural networks. The implementation in this research was confined to testing the working theory on a single perceptron, with the aim of building a perceptron capable of learning a more complex function than the simple linear function. More research and analysis has to be done on multi-layer Neural Network (NN) and deep NN architectures, where the results are compared to the conventional transformation function. F. Fan et al. [11] also introduced a backpropagation algorithm especially to better pass the gradients in the backward pass to update the weights of a second-order neural network. There is a significant change in the accuracy of the models used for comparison in this study; one trained using the traditional backpropagation and the other trained using the new backpropagation algorithm discussed in this paper. However, the paper summarizes results only from benchmarked biomedical and engineering datasets and hence is not enough to prove its working in real-world datasets.

In M. Blondel [12], training algorithms of HOFMs (higher-order factorization machines) [12] and new HOFMs with new formulas that used shared parameters have been presented. The study does a good job in terms of exploration of different functions and augmentations of HOFMs which can be applied to neurons. While the results have proven to be quite significant, the depth of the algorithms was not much. Some training algorithms, if not most, have a lot of scope for fine-tuning.

**Fig. 3** Summarized process pipeline



### 3 Methodology

The objective of this research is to propose an underlying architecture of dense layers that improves the performance of all deep-learning models. This section contains details of the entire research pipeline, starting from the custom layer definition to the evaluation of the trained model. A simple convolutional neural network is used to test the proposed architecture on five different well-known datasets. The process pipeline is summarized in Fig. 3.

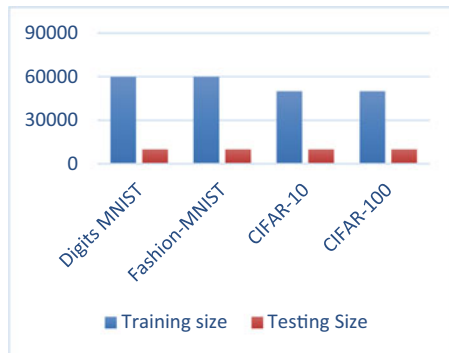
#### 3.1 Defining Custom Layers

To define a new layer, we implement three main steps. These include the layer definition, defining the type of variable and the computational kernel used, and implementing a call function, where we define the forward pass logic of the neuron.

For initialization, we define the units and the activation. We can control the number of units, the type of units, and other input independent initializations required when we build the model. All constant and non-constant variables (which directly or indirectly affect the computation in the forward pass) are defined here and added to the class scope for access by other class functions. The building section of the layer definition consists of defining the scope and the datatypes of the variables defined above, as well as a selection of the kernel. This section of the layer is implemented as a part of the model compilation process. During model compilation, memory is allocated for these variables according to their shape and type. Finally, in the section where we implement the calling functionality, the forward pass is coded, where the mathematical function to define what happens to the input variable is described. The function should contain at least the input to the model and return the augmented logit value.

In our research, we initialized the starting parameters using a randomize function which normalized the values for all the weight parameters between 0 and 1 using a

**Fig. 4** Training and testing sizes of the datasets used



Gaussian distribution. The bias coefficient, on the other hand, was initialized to 0. Both weight and bias coefficients were 32-bit float values.

After these steps have been completed, a simple test to check the randomness of the initialized values of the defined variables and the working of the forward pass confirms the proper working of the new layer.

### 3.2 Data Acquisition

As the datasets used in this research are very well maintained by TensorFlow [13], we use its dataset API to acquire pre-structured and organized data (using the load function). The datasets used for this study include 2 MNIST datasets, namely, Handwritten Digits MNIST [14] (containing handwritten digits in grayscale images of numbers from 0 to 9) and Fashion MNIST [15] (containing 10 different classes of clothing items). Both of these consist of 70,000 images, which have a size of  $28 \times 28$  pixels. The other 2 datasets are the CIFAR-10 [16] and the CIFAR-100 [16] datasets. The CIFAR-10 dataset consists of 60,000  $32 \times 32$  color images in 10 classes, and CIFAR-100 with 100 classes (6000 images per class). All these datasets form a benchmark and are recognized by the community for the task of classification. Their training and testing sizes have been shown in Fig. 4. The datasets used have a huge usability index, which is why the data preparation in this research constituted of only rescaling the data and no more augmentation was required.

### 3.3 Model Architecture and Model Compilation

The model architecture used in this study consists mainly of two types of layers, namely, the convolutional layer and the dense layer. The dense layer can be the conventional or the newly proposed quadratic dense layer. The model consists of

two pairs of convolutional layers followed by a MaxPooling layer and then one single convolutional layer, with the number of filters being 32, 64, 64 from the first layer to the last in that order. The activation for all these layers is ReLU [17], while the input to the first layer is dataset dependent. All the MaxPool layers in these pairs have a kernel size of  $2 \times 2$ . After flattening the output of the last convolutional layer, we have three pairs of the dense layers, followed by a dropout layer, with the number of units for the dense layer being 128, 64, 32 from the first layer to the last in that order. These dense layers are conventional and quadratic respectively in two different research experiments to compare them. The activations for all of these are ReLU and the dropout rate for all the dropout percentages is 20%. Finally, a simple dense layer is added with Softmax activation, and the number of units here equals the number of classes, which is 100 in CIFAR-100 and 10 in the case of the other 3 datasets.

Both the models for all the datasets are compiled with the sparse categorical cross-entropy [18] loss function and the Adam optimizer [19] to handle the gradient. The algorithm for the backpropagation of gradient in the proposed quadratic dense layer can be understood as follows.

Assuming the input variable is  $\in \mathcal{R}^d$ , the intermediate variable can be mathematically represented as:

$$z = W_1^{(1)}x^2 + W_2^{(1)}x \quad (1)$$

where  $W_1^{(1)}, W_2^{(1)} \in \mathcal{R}^{h \times d}$  are the trainable weight parameters. After running the logit or the intermediate variable  $z \in \mathcal{R}^h$  through the activation function  $\phi$ , we get the hidden activation of the intermediate logit:

$$h = \phi(z) \quad (2)$$

Assuming that the parameters of the output layer only possess a weight of  $W^{(2)}1$ ,  $W^{(2)}2 \in \mathcal{R}^q \times h$ , we can obtain an output layer variable with a vector of length  $q$ :

$$o = W_1^{(2)}h^2 + W_2^{(2)}h \quad (3)$$

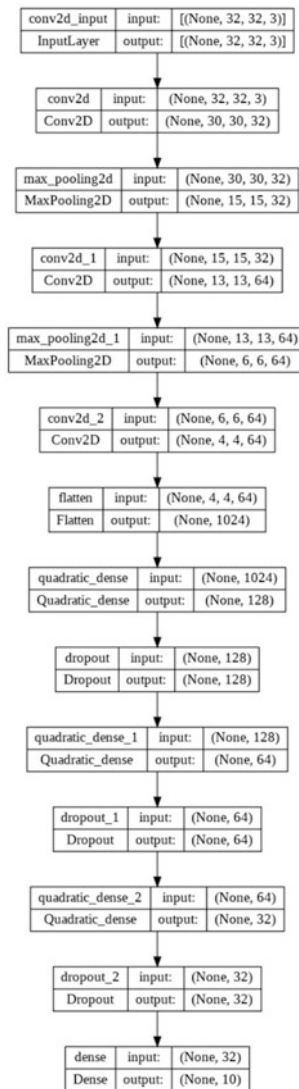
To calculate the loss for a single example, we can denote the loss  $\mathbf{L}$  as the value outputted by the loss function for an output  $h$  and expected real target label  $y$  as:

$$L = l(o, y) \quad (4)$$

If we were to introduce  $\ell_2$  regularization, then considering the hyperparameter term  $\lambda$ , we can calculate the regularization as:

$$s = \frac{\lambda}{2} \left( \left| 2W_1^{(1)} + W_2^{(1)} \right|_F^2 + \left| 2W_1^{(2)} + W_2^{(2)} \right|_F^2 \right) \quad (5)$$

**Fig. 5** BERT Model architecture



Now, a customized sequential dense neural network is built (refer Fig. 5). All the connections between blocks and layers, number of units, activation functions, and other parameters to augment the architecture of the model are defined here. The model is compiled with a sparse categorical cross-entropy loss function and a choice of accuracy metrics.

### 3.4 *Preparing Custom Training Loop*

Custom training loops include augmentations of what happens in a training step, for example, redefining forward and backward passes while training. When epoch-wise logging the accuracy and losses, early stopping, updating the learning rate with momentum, etc. can be done using callback functions. In our implementation, the patience of the early stopping callback was set to 3, and we monitored the loss. The momentum of the optimizer was set to 0.5. In our implementation, we have used callbacks to record certain accuracies and losses, which are presented in the results section of this paper.

### 3.5 *Model Training and Evaluation*

Finally, the model is trained and the losses and accuracies are monitored. TensorBoard [13] has been used in our research to visualize these results.

## 4 **Results and Analysis**

As previously mentioned, the model was evaluated on four different datasets. We recorded the conventional accuracy (neurons with linear transformation function) and the accuracy obtained with the proposed model (neurons with quadratic transformation function) at five different epoch values (refer Tables 1 and 2). The results per dataset have also been visualized (Fig. 6). Several notable inferences can be drawn from the results obtained.

The most important inference is that the quadratic dense layer converges much faster than the traditional layers in terms of both training and validation accuracies. For example, take the validation accuracies yielded on CIFAR-100. While the conventional model reaches an accuracy of approximately 26% at the 20th epoch, the new model reaches the same accuracy at around the 10th epoch itself. From the training accuracies on this dataset, we can observe that the linear model yields an accuracy of around 25% after the 20th epoch, while the proposed model reaches the same accuracy in less than five epochs. This shows a performance that is four times better with our model. Moreover, there is always a significant positive difference in the accuracies at every epoch, suggesting that the quadratic function has higher scope of learning. Although a large number of linear neurons can ultimately form any function, including quadratic, directly using this quadratic function in the neuron itself drastically reduces the number of epochs needed to reach a particular accuracy. This in turn reduces the overall computation time. Hence, if a heavy deep learning architecture with conventional neurons can reach the accuracy of 92% in 50 epochs, the same model will produce a 92% accuracy in half (or even fewer) number



**Table 1** Training accuracy obtained on varying epochs for four datasets

| Dataset                     | Number of epochs | Accuracy obtained with conventional neurons (%) | Accuracy obtained with quadratic neurons (%) |
|-----------------------------|------------------|---|--|
| Handwritten Digits<br>MNIST | 1                | 91.40   | 91.58  |
|                             | 2                | 97.82   | 97.99  |
|                             | 3                | 98.41   | 98.47  |
|                             | 4                | 98.76   | 98.79  |
|                             | 5                | 98.96   | 99   |
| Fashion- MNIST              | 1                | 73.12   | 75.59  |
|                             | 2                | 84.51   | 85.63  |
|                             | 3                | 87.36   | 88.40  |
|                             | 4                | 88.86   | 89.59  |
|                             | 5                | 89.70   | 90.18  |
| CIFAR-10                    | 1                | 32.10   | 94.19  |
|                             | 2                | 45.12   | 94.64  |
|                             | 5                | 54.41   | 94.60  |
|                             | 10               | 73.32   | 95   |
|                             | 20               | 81.69   | 95.70  |
| CIFAR-100                   | 1                | 2.56  | 23.34  |
|                             | 2                | 5.12  | 24   |
|                             | 5                | 12.41   | 25.21  |
|                             | 10               | 18.49   | 27.07  |
|                             | 20               | 24.86   | 30.26  |

of epochs with the new neurons. This hypothesis can be successfully validated by implementing the proposed methodology for ResNet, InceptionNet, VGGNet, etc. However, we can predict that both the types of neurons or dense layers might ultimately reach the same accuracy, just with a drastically varying number of epochs.

It is also interesting to note how the initial accuracy yielded by the new neurons is always much better than the old ones. For instance, training and validation accuracies after the first epoch on CIFAR-10 were 94.19% and 90.30% with the proposed neurons, against just 32.10% and 46.10%, respectively, with the conventional ones. On CIFAR-100, this difference is even higher (23.24% against 2.56% while training, and 24.65% against 4.18% during validation). Importantly, the training time was not affected much by the addition of an extra dimension to the neuron's equation. The execution time for every step in all the datasets remained the same (at 5 ms for both the models). Both the models required an average of 7–8 seconds for every epoch to complete. The model weight was affected; the architecture with the quadratic dense layer resulted in a model with 70% more trainable parameters (at 342,468), compared to the conventional dense layer model, which had a total of 201,156 trainable parameters. Both, the total number of

**Table 2** Validation accuracy obtained on varying epochs for four datasets

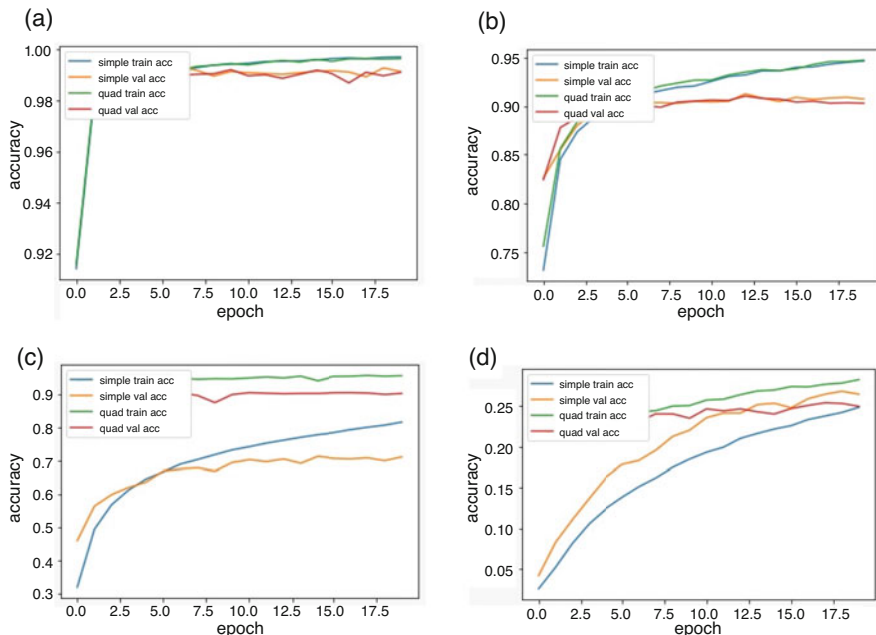
| Dataset                     | Number of epochs | Accuracy obtained with conventional neurons (%) | Accuracy obtained with quadratic neurons (%) |
|-----------------------------|------------------|---|--|
| Handwritten Digits<br>MNIST | 1                | 98.10   | 98.45  |
|                             | 2                | 98.55   | 98.87  |
|                             | 3                | 98.85   | 98.71  |
|                             | 4                | 99.02   | 98.78  |
|                             | 5                | 98.96   | 98.98  |
| Fashion- MNIST              | 1                | 82.58   | 82.40  |
|                             | 2                | 85.51   | 87.79  |
|                             | 3                | 88.01   | 88.83  |
|                             | 4                | 89.29   | 89.34  |
|                             | 5                | 90.12   | 90.11  |
| CIFAR-10                    | 1                | 46.10   | 90.30  |
|                             | 2                | 56.53   | 90.46  |
|                             | 5                | 63.73   | 90.47  |
|                             | 10               | 69.52   | 90   |
|                             | 20               | 71.20   | 90.64  |
| CIFAR-100                   | 1                | 4.18  | 24.65  |
|                             | 2                | 8.28  | 24.89  |
|                             | 5                | 16.24   | 25   |
|                             | 10               | 22.04   | 25.51  |
|                             | 20               | 26.47   | 26.97  |

parameters and the trainable parameters, stay the same as none of the layers or logits were frozen or untrainable in the model architecture.

Finally, as predicted, the improvement in the accuracy for initial epochs is much higher in CIFAR-100 than in the other datasets. This is because of the large learning potential that produces a high scope for improvement in the accuracy obtained from the first epoch on this dataset, high variance in the data, and the high number of output classes. On the contrary, the first epoch itself is yielding a high accuracy on the MNIST datasets due to a good fit [20]. Hence, the model is easily overfitted, and the results are only recorded at a low number of epochs. Here, we can observe that despite the less room for improvement, the quadratic neurons perform better.

## 5 Conclusion

This paper proposed a new methodology for a neuron’s output computation, by replacing the conventional linear transformation function with a quadratic transformation function. When tested on four different popular datasets for a



**Fig. 6** Simple and quadratic (quad) accuracies plotted for the first 20 epochs for all 4 datasets: (a) MNIST, (b) Fashion-MNIST, (c) CIFAR-10, and (d) CIFAR-100

simple dense neural network, an improvement in the accuracy is observed. This improvement is higher when the initial accuracy is low, thus significantly reducing the computation time (and the number of epochs) to arrive at a particular accuracy. Nevertheless, initial convergence to higher accuracies is always much faster in the proposed model. Moreover, the results would become exponentially better with a very large number of neurons in the architecture. The proposed methodology can hence improve the performance of any deep learning architectures containing dense layers.

While the models built using the proposed transformation function do have a higher model weight, they show faster convergence. If we were to increase the number of fully connected layers in the model built using the conventional transformation function with the aim of getting a similar convergence to the quadratic transformation function, we would encounter the vanishing gradient problem. Hence, the proposed methodology also overcomes the vanishing gradient problem in the conventional transformation function when we wish to increase the parameter count, without changing the number of parameters in a given layer.

The real-world applications of these new neurons are numerous. Besides the faster convergence which yields drastically better accuracies when trained for the same duration as conventional neurons, this improved perceptron helps in incredibly reducing the training time of a model in case the same number of parameters are

used in both the models. This in turn means that any research involving DL would be accelerated, and any application utilizing DL models would become more efficient. For instance, in systems like face recognition and real-time threat detection, which use few shot learning techniques, the models will have a lower inference time and yield better accuracies. In the field of medicine, all analyses involving DL would be accelerated, and so on.

There is a vast future scope for this study. While we analyzed the replacement of the linear function of a neuron with a quadratic sum, the same can be replaced with other functions that will potentially yield further improved results [10, 21]. Furthermore, the true power of this methodology can be seen when the results are recorded for larger deep learning architectures, employing a much higher number of parameters. For example, VGG-16 [22], which consists of over 134.7 million parameters, would ideally portray a much better performance on a dataset than ResNet-18, which has 11.4 million parameters.

## References

1. G.E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, R.R. Salakhutdinov, Improving neural networks by preventing co-adaptation of feature detectors. arXiv preprint arXiv:1207.0580 (2012)
2. K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (Las Vegas, 2016), pp. 770–778
3. M. Lin, Q. Chen, S. Yan, Network in network. arXiv preprint arXiv:1312.4400 (2013)
4. F. Rosenblatt, The perceptron: A probabilistic model for information storage and organization in the brain. *Psychol. Rev.* **65**(6), 386 (1958)
5. H. Lin, S. Jegelka, Resnet with one-neuron hidden layers is a universal approximator. *Adv. Neural Inf. Proces. Syst.* **31** (2018)
6. F. Fan, H. Shan, M.K. Kalra, R. Singh, G. Qian, M. Getzin, Y. Teng, J. Hahn, G. Wang, Quadratic autoencoder (Q-AE) for low-dose CT denoising. *IEEE Trans. Med. Imaging* **39**(6), 2035–2050 (2019)
7. V. Kůrková, M. Sanguineti, Probabilistic lower bounds for approximation by shallow perceptron networks. *Neural Netw.* **91**, 34–41 (2017)
8. C.L. Giles, T. Maxwell, Learning, invariance, and generalization in high-order neural networks. *Appl. Opt.* **26**(23), 4972–4978 (1987)
9. S. Du, J. Lee, On the power of over-parametrization in neural networks with quadratic activation, in *International Conference on Machine Learning*, (PMLR, 2018), pp. 1329–1338
10. F. Fan, W. Cong, G. Wang, A new type of neuron for machine learning. *Int. J. Numer. Method Biomed. Eng.* **34**(2), e2920 (2018)
11. F. Fan, W. Cong, G. Wang, Generalized backpropagation algorithm for training second-order neural networks. *Int. J. Numer. Method Biomed. Eng.* **34**(5), e2956 (2018)
12. M. Blondel, A. Fujino, N. Ueda, M. Ishihata, Higher-order factorization machines. *Adv. Neural Inf. Proces. Syst.* **29** (2016)
13. M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al., *TensorFlow: A System for Large-Scale Machine Learning* (OSDI, 2016), pp. 265–283
14. L. Deng, The MNIST database of handwritten digit images for machine learning research. *IEEE Signal Process. Mag.* **29**(6), 141–142 (2012)

15. H. Xiao, K. Rasul, R. Vollgraf, Fashion-mnist: A novel image dataset for benchmarking machine learning algorithms. arXiv preprint arXiv:1708.07747 (2017)
16. A. Krizhevsky, *Learning Multiple Layers of Features from Tiny Images* (University of Toronto, Toronto, 2009)
17. A.F. Agarap, Deep learning using rectified linear units (relu). arXiv preprint arXiv:1803.08375 (2018)
18. J. Bridle, Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters. *Adv. Neural Inf. Proces. Syst.* **2** (1989)
19. D.P. Kingma, J. Ba, Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
20. N. Venkat, *The Curse of Dimensionality: Inside Out* (2018). <https://doi.org/10.13140/RG.2.2.29631.36006>
21. F. Fan, J. Xiong, G. Wang, Universal approximation with quadratic deep networks. *Neural Netw.* **124**, 383–392 (2020)
22. K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)

# Analysis of Metaheuristic Algorithms for Optimized Extreme Learning Machines in Various Sectors



D. Devikanniga and D. Stalin Alex

## 1 Introduction

The extreme learning machine (ELM) [1] is supervised machine learning algorithm, which is used to solve various domain applications in real-time. As a one-time tuning machine, ELM has proved its efficiency. However, it still suffers from overfitting problem due to the enormous hidden neurons found in the hidden layer. The metaheuristic optimization algorithms (MHOAs) play a vital role in this place by producing the optimized weights and biases for ELM. This helps the ELM to give the outstanding performance in many critical applications.

### 1.1 ELM and Its Training

The main characteristic of ELM is its architecture that contains only one hidden layer. It is known for its tuning-free architecture as their synaptic weights get updated in iteration. Figure 1 shows the architecture of ELM architecture, which has one input layer with  $N$  nodes, one hidden layer with  $L$  neurons, and one output layer bearing  $M$  neurons. In the same network, the different types of activation functions, namely multiquadric, sigmoidal, Gaussian, hyperbolic tangent, and others can be used during training and testing. The random weights ( $W$ ) and biases ( $b$ ) in a specific

---

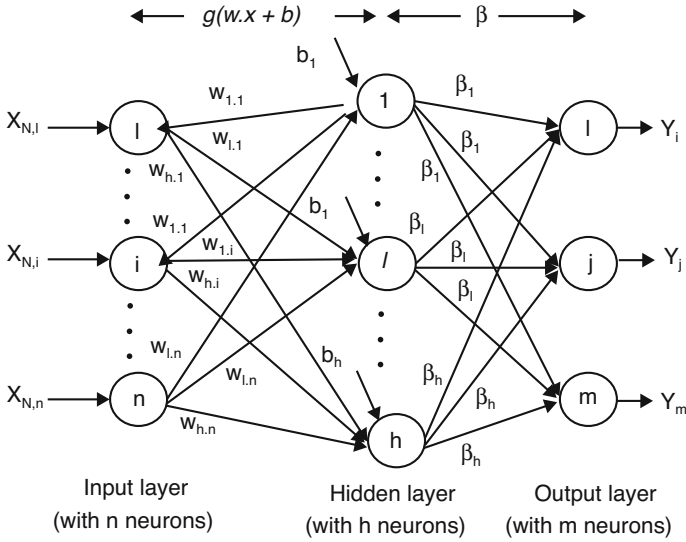
D. Devikanniga (✉)

Department of CSE, GITAM (Deemed-to-be) University, Bangalore, India

D. Stalin Alex

Department of CSE (Data Science), State University of Bangladesh, Dhaka, Bangladesh

e-mail: [drstalinalex.cse@sub.edu.bd](mailto:drstalinalex.cse@sub.edu.bd)



**Fig. 1** Architecture of extreme learning machine

range are used for initializing the network. ELM is trained using two main steps as follows:

- In first step, Random feature mapping is performed where the input ( $X$ ) is mapped to ELM's feature space, and it expressed in non-square matrix ( $H$ ),  $h(X) = g(W, X, b)$ .
- The second step is Linear parameter solving, in this the synaptic weights between the output and hidden layer known as output weight ( $\beta$ ) that is evaluated as  $Q = H^\dagger T$

$H^\dagger$ : Moore – Penrose generalized inverse of matrix  $H$  [ $H^\dagger = (H^T H)^{-1} H^T$ ]

$T$ : target matrix for training.

The minimization of approximation error is taken as the objective function. The working principle of ELM is given in steps as below:

- Step 1: Load the training data.
- Step 2: The training data is grouped into two as input and target.
- Step 3: Initialize the network parameters.
- Step 4: Randomly initialize the input synaptic weights and the biases.
- Step 5: Evaluate the hidden layer  $H$  output matrix.
- Step 6: Calculate output weight  $\beta$ .
- Step 7: Calculate network output  $Y$ .
- Step 8: Load the input test data,  $Z$ .
- Step 9: Calculate  $H_{\text{test}}$  that is output matrix for hidden layer using input test data.
- Step 10: Evaluate test data output,  $y_{\text{test}}$ .

Step 11: Test data is classified as: if  $y_{\text{test}} > 0$ , then the test record is classified as class 1, else as class 2.

## ***1.2 Metaheuristic Optimization***

Optimization is the iterative procedure of maximization or minimization of the fitness or objective function based on one or more constraints to achieve maximum profit or minimum loss, respectively. There are several types of optimization algorithms, such as traditional, heuristic, and metaheuristic. Many efficient nature-inspired metaheuristic algorithms were developed over a past few decades and applied for optimization of many machine learning algorithms to get maximum yield. Normally, they are adopted for feature selection problems, hyperparameter tuning, optimizing weights and biases, and many others [2].

## ***1.3 Training Details of ELM Using Metaheuristic Optimization***

The synaptic weights and biases of the ELM is normally initialized with random values. To enhance the performance of the ELM, the metaheuristic algorithms are used to provide the optimized weights and biases. The training of ELM is discussed in the Sect. 1.1. The generic structure of MHOA-based ELM is given in Fig. 2.

## **2 Applications of MHOA-Based ELM in the Literature**

The Dandelion algorithm (DA) is devised based on sowing strategy of the Dandelion [3]. This algorithm follows with procedures such as the dandelion sowing, self-learning sowing, and selection strategy. The algorithm starts with only one dandelion which is sown in a predefined radius which changes dynamically on finding the optimal solution. The dandelion is assumed to have self-learning ability that guides in optimal search. The best dandelion is taken for the next iteration. The performance is compared with other algorithms, such as EFWA, genetic algorithm (GA), particle swarm optimization (PSO), and Bat algorithm BA. Twelve various test functions are used for the verification of the algorithm. It has given good optimum accuracy and convergence speed. Here the role of DA is for optimizing the synaptic weights and the biases of ELM network. One regression dataset and seven classification dataset is used in the study. The datasets such as Sinc, Diabetes, Landsat satellite image, image segments are taken from the link [www.ntu.edu.sg](http://www.ntu.edu.sg) and the remaining datasets such as Default of credit card clients, Madelon, skin segmentation, statlog (shuttle) are taken from UCI repository. The results of DA-



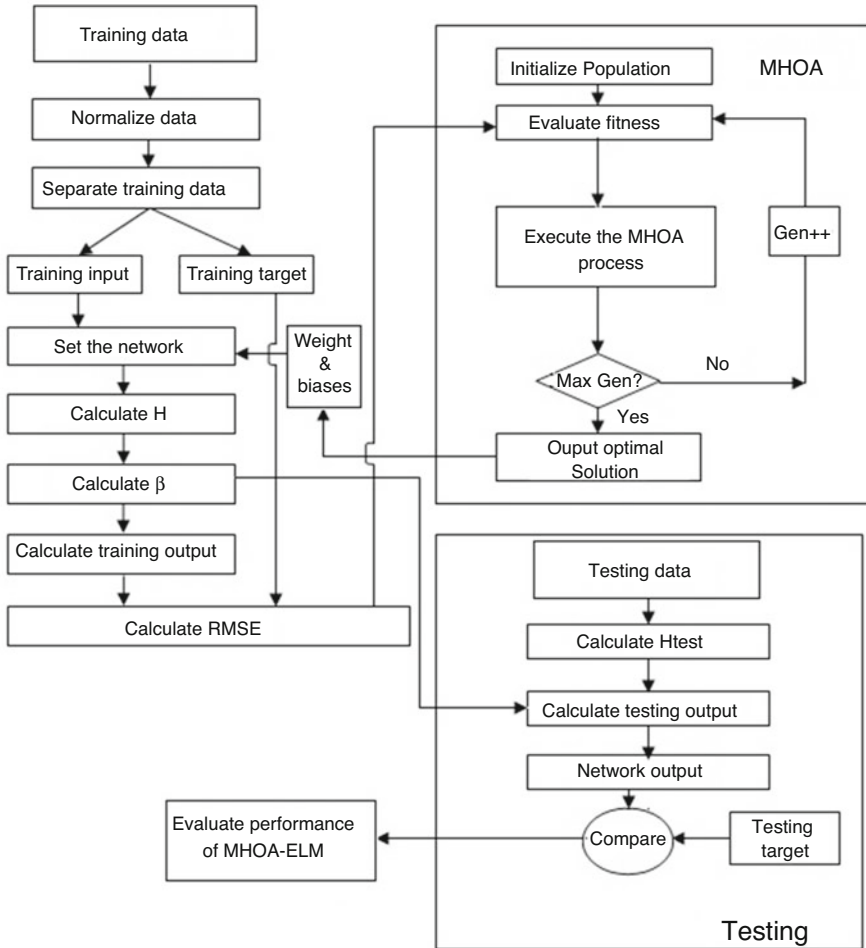


Fig. 2 Generic structure of MHOA-based ELM

ELM are compared with EFWA-based ELM, GA-based ELM, PSO-based ELM, and BA-based ELM. DA-ELM to have better accuracy due to its generalization ability in both classification and regression but has consumed more time than other ELMs.

In [4], the Bat optimization algorithm (BOA) is used as the bio-inspired solver for optimizing ELM. BOA is based on the echolocation behavior of bats. The algorithm starts with bat population, where the position of each bat determines the ELM's synaptic weights and biases. This ELM optimized with bats used for the classification of patients with the Parkinson's disease. The dataset that is retrieved from UCI Machine learning repository is adopted for this work. The performance

of the original ELM is compared with this bats-based ELM and found that the latter has produced minimum loss of 3.27% with maximum accuracy of 96.74%.

An upgraded bats algorithm (UBA) [5] is used to optimize ELM to reduce its learning time. The bats algorithm is upgraded with four modifications, such as modulation to the pulse emission rate vector, frequency, velocity, and a prescribed, predetermined number of allowed trials. The upgraded bats algorithm constitutes the optimal weights and biases for ELM. This algorithm is compared with genetically optimized ELM (GO-ELM), original ELM, improved genetic algorithm for SLFN, self-adaptive evolutionary ELM and the Levenberg Marquardt (LM) SLFN, to prove its efficiency. The seven benchmark datasets namely the Servo, Boston Housing, Automobile MPG, CPU, Ailerons, Cancer, Concrete strength computation were used in this study. The UBA is reported to have less computation time in case of all datasets.

The metaheuristic optimization algorithms such as PSO and the gravitational search algorithm (GSA) are hybridized as HPSOGSA in [6]. This hybrid optimization algorithm is employed for tuning the ELMs weights and biases. It is applied in the forecasting of the wind speed. The wavelet packet decomposition is used with this algorithm for decomposing the wind speed. For this, the two wind speed datasets are collected in a wind farm located in Anhui, China, is used in this work. The binary PSO-GSA is adopted for the feature selection. Hence, this method is alternatively called as WPD-HPSOGSA-ELM. The performance based on the algorithm's forecasting is compared with that of HPSOGSA-ELM with EMD and wavelet transform, WPD-HPSOGSA-ELM without feature selection and original ELM is used. It is concluded in this study [6] that WPDHPSOGSA with FS has yielded minimum RMSE, MAE, MAPE, and normalized mean absolute scaled error than the other models.

PSO is used with ELM in PSO-ELM to replace the evolutionary ELM [7]. The PSO-ELM is compared with PSO-LM and PSO-Backpropagation (BP). The data used is the statistical data about the corn production from 1978 to 2000 from the National Statistics Bureau of China, 2003. The data from 1978 to 1998 is used for training and production is predicted for the years 1999 and 2000. The PSO-ELM is quicker than hybrid PSO and BP. In [8], the ELM's weights and biases are optimized with bacterial foraging algorithm (BFA). The BFA-ELM is used for better performance in the higher dimension problems.

To extend the ability of classification, E-ELM is extended as cross validation E-ELM (EELMCV) and cross validation improved Evolutionary ELM (IEELMCV) [9]. The former is described as E-ELM embedded with cross validation technique in the training stage to solve the over training problem. In latter algorithm, the constant synaptic weights and biases are applied to the hidden nodes instead of deleting it in order to avoid the weakening of generalization ability of the network. The datasets such as GeorgiaTech face database, ORL database, FERET face database, and Combo face dataset, which all contain the face images, are used for accessing its performance. The results of EELMCV and IEELMCV are compared with original ELM, E-ELM, BP Neural network. It is reported that the accuracy of IE\_ELMCV is better than E-ELMCV and others. But both IE-ELM and E-ELMCV required more

training time that ELM and E-ELM but lesser than BP. These two algorithms were reported to be stable in classifying the images.

The PSO is used to obtain the optimized fitness values as the synaptic weights and biases of ELM. This algorithm is also called Improved ELM (IPSO-ELM) [10] is used for the brain tumor tissue characterization. The Surgical planning analyze (SPA) benchmark dataset and real time dataset containing brain tumor images were taken. The features of these images were extracted using Run-length matrix and gray-level co-occurrence matrix (GLCM). The optimal attributes were selected using genetic algorithm. The images were classified with the optimal dataset features using IPSO-ELM. The results were compared with BPN, Support vector machines (SVM), and ELM classifiers. This IPSO-ELM classifier yielded the highest of 98.25% accuracy than others.

The entire hidden layer configuration of ELM is optimized by genetic algorithm and the algorithm is called as genetically optimized ELM (GO-ELM) [11]. The relevant features are selected using auxiliary binary selection. The Tikhonov's-regularization is applied in place of least squares algorithm to get the output weights. The GO-ELM is applied on the benchmark datasets such as Servo, Automobile MPG, Cancer, and Boston housing and Price. The algorithms such as IGA-SHFFN network, SaE-ELM, LM-SLFN, and ELM are used for comparing GO-ELM. It is observed that GO-ELM has resulted in good generalization capacity than others. GA-ELM is also used to check the temperature prevailing in burning zone of a real cement kiln plant, where it has proved its efficiency again.

The artificial algae algorithm that is involved with a multi-light source is combined with ELM [12] and applied for various classification datasets like Statog-heart, Blood, Vertebral column, and Pima-Indian diabetes. It is observed that the outstanding results were produced by this algorithm. The genetic algorithm is combined with upward-based climbing and downward-based climbing ELMs. This algorithm is then applied in power system economic-dispatch problem. This has outperformed the basic GA on accuracy and computational efficiency. The data based on water inflow that is registered from two hydro plants in Hunan province, China, are taken for the study [13]. Quantum Particle swarm optimization, a new algorithm shortly known as QPSO is developed based on PSO and DELTA trap. The QPSO is used to feed the synaptic weights and the biases for ELM. This QPSO-based ELM is applied for solving handwritten numerical recognition problem. The USPS handwriting numerical dataset is used. The results produced by the QPSO-ELM were considered for the comparison with that of BP and ELM. It is found that QPSO-ELM has a faster response to unknown data, has better generalization performance, and avoids overfitting and local optima [14].

The cuckoo search (CS) algorithm that is devised taking on parasitic behavior of cuckoo in laying and hatching eggs. The probability factor and stepsize of this algorithm is modified and improved cuckoo search (ICS) algorithm is developed [15]. This ICS algorithm is used to produce the ELM's synaptic weights and the biases. This algorithm is tested on four benchmark datasets such as Bupa, Hepatitis, Wisconsin breast cancer, and Diabetes. The outcomes are tested based

on performance metrics. The values are compared with ELM and CSELM. It is found that ICS-ELM has performed better than others.

Artificial bee colony (ABC) algorithm is written by inspiring the foraging behavior of honey bees. This original ABC algorithm is improved by modifying the search equation. In this, the best fitness solution that is retrieved from the existing population is considered and the value 0.1 is chosen as inertia weight. The ELM's synaptic weights and the biases are fed by this modified ABC (MABC) algorithm. This MABC-based ELM is applied for forecasting the load within a short-term driven by wavelet transform. For performing this, the two of datasets namely North American electricity-utility data and ISO New England data are used. Six examples were discussed. The performance based on convergence between the MABC-ELM and conventional Neuro-evolution method are discussed based on the examples. It is observed that MABC-ELM had yielded a better output than the other. In second example, forecasting the load for 1 hour and 24-hour are performed using the MABC-ELM and it is found that encouraging results were produced by it for the simulated temperatures. The third example includes the wavelet transform with MABC-ELM (WT-ELM-MABC) on forecasting performance. This is compared with ELM, WT-based ELM- and MABC-based ELM. It is found that WT-ELM-based MABC has outperformed. Other examples prove that the proposed algorithm has yielded best results [16].

To predict the effluent from biological waste water treatment plant, three intelligent algorithms such as self-adaptive differential evolution algorithm, trigonometric mutation operation differential evolution algorithm (TDE), and basic differential evolution algorithm were used to optimize ELM. The original ELM is also used in the experiment. Among all techniques, it was observed that TDE optimized has produced the superior results [17]. The electric energy demand is forecasted accurately by using ELM, optimized by MHOAs such as Jellyfish search, Harris Hawk, Flower pollination optimization algorithms. The electric energy demand datasets that are contributed by Thailand's Provincial Electricity Authority, is used in this work [18]. This dataset is divided into seven subtypes to carry out this research. Overall, the Jellyfish search algorithm combined with ELM is found to be stable and produce less error.

The MHOAs such as PSO, ABC, and Grey Wolf optimization (GWO) algorithms are used with ELM in [19] for optimizing its biases and weights. These algorithms are applied for the task of classification of the datasets such as Australian credit, Heart disease, and Diabetes detection. On experimentation, it is found that GWO optimized ELM has produced better results for Australian credit dataset and PSO optimized ELM has produced better accuracy for the other two datasets.

In agricultural irrigation, the accurate Pan evaporation is predicted by optimizing ELMs with two MHOAs such as whale optimization algorithm and flower pollination algorithm. This case study [20] was conducted for the Poyang Lake Basin, which is located in southern China. The data was collected from four weather stations. From the experiment conducted, it is observed that the ELM hybridized with flower pollination algorithm has produced highest accuracy in prediction, for all the four stations.

The ELM optimized with MHOAs such as Chemical reaction optimization, Teaching and Learning optimization, and Fireworks algorithm are used [21] for forecasting the exchange rate accurately. These algorithms are noted to have produced better results when compared with primitive optimization algorithms. The modified Red Deer optimization ELM Sparse Autoencoder model is used [22] for the classification of sentiments. This model is compared with other algorithms such as gradient boosted SVM, logistic regression, SVM, Random Forest, and Gradient Boosting Tree, for the dataset, which has 64,295 instances with three types of sentiments such as positive, neutral, and negative.

### 3 Results and Discussions

The ELM optimized with several metaheuristic algorithms and its application to different types of problems that are in the literature are discussed. Thus, the extensive use of this single iterated feed forward neural network and performance shows its robust nature. The MHOA-based ELM, thus cater the needs of various sectors such as agriculture, stock market, electricity boards, weather prediction, medicine and many others. It is found that its simpler working rule gives the flexibility to apply it for many decisions making and problem-solving approaches.

### References

1. G.-B. Huang, Q.-Y. Zhu, C.-K. Siew, Extreme learning machine: A new learning scheme of feedforward neural networks, in *Proc. Int. Joint Conf. on Neural Networks*, (Budapest, 2004), pp. 25–29
2. K.L. Du, M.N.S. Swamy, *Search and Optimization by Metaheuristics: Techniques and Algorithms Inspired by Nature* (Springer, New York, 2016), p. 434
3. C. Gong, S. Han, X. Li, L. Zhao, X. Liu, A new dandelion algorithm and optimization for extreme learning machine. *J. Exp. Theor. Artif. Intell.* (2017). <https://doi.org/10.1080/0952813X.2017.1413142>
4. R. Olivares, R. Munoz, R. Soto, B. Crawford, D. Cárdenas, A. Ponce, C. Taramasco, An optimized brain-based algorithm for classifying Parkinson’s disease. *Appl. Sci.* **1**(5), 1827 (2020). <https://doi.org/10.3390/app10051827>
5. A. Alihodzic, E. Tuba, M. Tuba, An upgraded bat algorithm for tuning extreme learning machines for data classification, in *Proceedings of GECCO’17 Companion*, Berlin, Germany, 15–19 July 2017, 2 pages. (2017). <https://doi.org/10.1145/3067695.3076088>
6. S. Sun, F. Jingqi, F. Zhu, D. Dajun, A hybrid structure of an extreme learning machine combined with feature selection, signal decomposition and parameter optimization for short-term wind speed forecasting. *Trans. Inst. Meas. Control.* **42**(1), 3–21 (2020)
7. Y. Xu, Y. Shu, Evolutionary extreme learning machine – Based on particle swarm optimization, in *Advances in Neural Networks – ISNN 2006. ISNN 2006*, Lecture Notes in Computer Science, ed. by J. Wang, Z. Yi, J.M. Zurada, B.L. Lu, H. Yin, vol. 3971, (Springer, Berlin, Heidelberg, 2006). [https://doi.org/10.1007/11759966\\_95](https://doi.org/10.1007/11759966_95)

8. J.-H. Cho, M.-G. Chun, D.-J. Lee, Parameter optimization of extreme learning machine using bacterial foraging algorithm, 2007. *J. Fuzzy Log. Intell. Syst.* **17**(6). <https://doi.org/10.5391/JKHS.2007.17.6.807>
9. N. Liu, H. Wang, Evolutionary extreme learning machine and its application to image analysis. *J. Signal Process. Syst.* **73**, 73–81 (2013)
10. B. Arunadevi, S.N. Deepa, Brain tumor tissue categorization in 3d magnetic resonance images using improved PSO for extreme learning machine. *Prog. Electromagn. Res. B* **49**, 31–54 (2013)
11. T. Matias, R. Araújo, C.H. Antunes, D. Gabriel, Genetically optimized extreme learning machine, in *2013 IEEE 18th Conference on Emerging Technologies & Factory Automation (ETFa)*, (2013), pp. 1–8. <https://doi.org/10.1109/ETFa.2013.6647975>
12. D. Devikanniga, J.S. Raj, Improving classification accuracy using hybrid of extreme learning machine and artificial algae algorithm with multi-light source. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, World Scientific Publishing Company **28**(2), 43–51 (2020)
13. H. Yang, J. Yi, J. Zhao, Z. Dong, Extreme learning machine based genetic algorithm and its application in power system economic dispatch. *Neurocomputing* **102**, 154–162 (2013)
14. X. Sun, L. Qin, An extreme learning machine based on quantum particle swarm optimization and its application in handwritten numeral recognition, in *2014 IEEE 5th International Conference on Software Engineering and Service Science*, (2014), pp. 323–326. <https://doi.org/10.1109/ICSESS.2014.6933573>
15. P. Mohapatra, S. Chakravarty, P.K. Dash, An improved cuckoo search based extreme learning machine for medical data classification. *Swarm Evol. Comput.* **24**, 25–49 (2015)
16. S. Li, P. Wang, L. Goel, Short-term load forecasting by wavelet transform and evolutionary extreme learning machine. *Electr. Power Syst. Res.* **122**, 96–103 (2015)
17. M.-J. Lin, C.-X. Zhang, C.-H. Su, Prediction of effluent from WWTPs using differential evolutionary extreme learning machines, in *2016 35th Chinese Control Conference (CCC)*, (2016), pp. 2034–2038. <https://doi.org/10.1109/ChiCC.2016.7553666>
18. B. Sarunyoo, S. Chitchai, F. Pradit, C. Rongrit, Metaheuristic extreme learning machine for improving performance of electric energy demand forecasting. *Computers* **11**(66), 1–34 (2022)
19. H. Escobar, E. Cuevas, Implementation of metaheuristics with extreme learning machines, in *Metaheuristics in Machine Learning: Theory and Applications. Studies in Computational Intelligence*, ed. by D. Oliva, E.H. Houssein, S. Hinojosa, vol. 967, (Springer, Cham, 2021)
20. W. Lifeng, H. Guomin, F. Junliang, M. Xin, Z. Hanmi, Z. Wenzhi, Hybrid extreme learning machine with meta-heuristic algorithms for monthly pan evaporation prediction. *Comput. Electron. Agric.* **168**, 105115 (2020)
21. K.K. Sahu, S.C. Nayak, H.S. Behera, Extreme learning with metaheuristic optimization for exchange rate forecasting. *Int. J. Swarm Intell. Res. (IJSIR)* **13**(1), 1–25 (2022)
22. R. Bhaskaran, S. Saravanan, M. Kavitha, C. Jeyalakshmi, K. Seifedine, T. Hafiz, R. Alkhamash, Intelligent machine learning with metaheuristics based sentiment analysis and classification. *Comput. Syst. Sci. Eng.* **44**(1), 235–247 (2023)

# Metal and Metal Oxide Nanoparticle Image Analysis Using Machine Learning Algorithm



Parashuram Bannigidad , Namita Potraj ,  
and Prabhuodeyara Gurubasavaraj 

## 1 Introduction

Nanotechnology offers the ability to change the properties of materials by controlling their size and structure. This has facilitated research into a wide range of potential applications for nanomaterials. The synthesis action is an important process that involves the primary process of creating metal or metal oxides. The formation of even-sized nanoparticles is an unpredictable and difficult task. Synthesis processes include many properties such as pH, time, reaction temperature, pressure, and catalyst. The formation of nanoparticles is highly dependent on these factors. Changes in these factors reflect the size and structure of nanomaterials. The characterization techniques help to visualize the nanoparticle FESEM or TEM images [1–3]. This study focuses on various FESEM and TEM nanoparticle images. Hence, these images can be further used in various application-oriented techniques. Therefore, it is very important to know the properties of nanomaterials before utilizing them in applications. Figure 1a shows the sample nanoparticle images of the training dataset, and Fig. 1b shows the sample nanoparticle images of the testing dataset.

The morphology of nanomaterials can be characterized using FESEM and TEM nanoparticle images. Zhijian Sun et al. [4] proposed a system that is composed of three stages: extraction of nanoparticle shape, segmentation of nanoparticles using a powerful lightweight deep learning network (NSNet), and statistical evaluation on SEM and TEM images. Boron TEM nanoparticles have wide applications

---

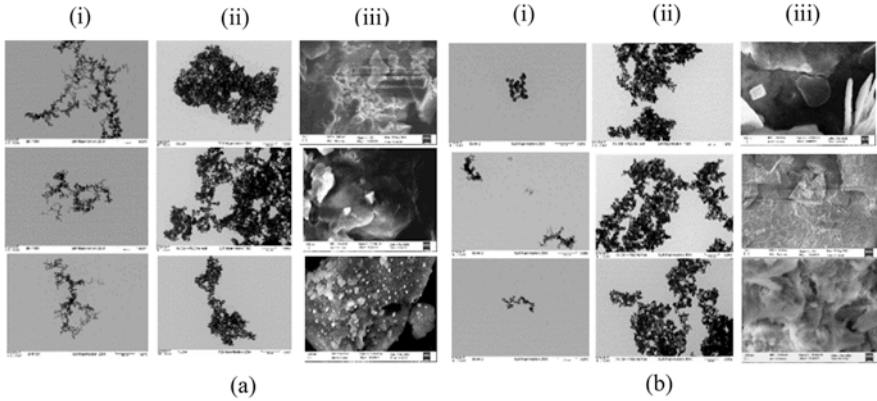
P. Bannigidad · N. Potraj (✉)

Department of Computer Science, Rani Channamma University, Belagavi, Karnataka, India

P. Gurubasavaraj

Department of Chemistry, Rani Channamma University, Belagavi, Karnataka, India





**Fig. 1** Sample FESEM and TEM nanoparticle images – (a) (i) boron, (ii) iron, and (iii) silver nanoparticle images of the training dataset; (b) (i) boron, (ii) iron, and (iii) silver nanoparticle images of testing dataset

from being used as a food preservative to a beneficial nutrient to humans [5] to treatments of cancer cells using neuron capture therapy [6]. Iron oxide nanoparticles (iron oxide NPS) have ruled biomedical packages for protein immobilization, thermal treatment, MRI, and drug conveyance because of their salient features, which include low-slung harmfulness, superparamagnetic capabilities, and facile separation methodology. Segmentation of boron TEM nanoparticle images using digital image processing techniques is vital and helps in analyzing the shapes and structures of boron TEM nanoparticle images. Bannigidad et al. [7] have proposed various clustering techniques such as K-means and Fuzzy C-means to analyze the shapes and structures of boron TEM nanoparticle images. Iron oxide nanoparticles have a variety of uses, including the treatment of cancer, medication transport, antifungal and antibacterial action, imaging, and cellular labeling has been suggested by Nene et al. [8]. Calderon et al. [9] suggested the high impact and applications of silver FESEM nanoparticles in several industries, including agriculture, food, polymers, biomaterials composites, ceramics, and energy. The texture features of any images are significant as they form the basis for the field of utilization. Qian Zaho et al. [10] have presented the role of surface features such as contrast, homogeneity, entropy, energy, and correlation of images in the diagnosis of lung disease. In image processing, the texture may be described as a feature of spatial versions in pixel luminance intensity. Image texture analysis relies heavily on the spatial relationships between pixel gray levels. The statistical texture features such as kurtosis, skewness, and entropy [11] play a vital role in image classification and segmentation. Hung et al. [12] proposed that on the availability of texture capabilities, many classification and segmentation algorithms from conventional sample popularity may be used for labeling textural classes.

The content-based image retrieval [13–14] is beneficial when the features of images such as their color, shape, structure, and texture features are stored in a



database and can be retrieved later for future use. The fundamental tenet of content-based image retrieval before storing or retrieving an image from a database is proposed by Charde et al. [15]. In content-based image retrieval systems, at the beginning features are extracted from the image, which may include color, shape, texture, and other attributes. For later use, the database has these features. Similarly, the extraction of features from a query image is done and compared to feature vectors already present in the database. If the difference between two feature vectors is sufficiently tiny, it is considered as the database's associated image is matching with the query. Users are shown a group of related target images that are grouped according to the similarity index for retrieval results. The classification is the later process that is carried out by keeping these extracted features as a base. Zhang [16] has proposed various traditional texture and contemporary texture features such as tamura, gray-level co-occurrence matrices (GLCM), Markov random field (MRF), fractal dimension (FD), DCT, Gabor, wavelet, and curvelet. Ramola et al. [17] have proposed a gray-level co-occurrence matrix (GLCM), local binary pattern (LBP), autocorrelation function (ACF), and histogram pattern methodologies for texture classification. Przemysław Kupidura [18] compared the efficacy of several texture analysis methods such as GLCM, granulometric analysis, and Laplace filters for enhancing the land or cover classification in satellite imagery. Andrzej Materka [19] defined the capabilities of MaZda; a computer application for quantitative texture evaluation used to research the textures of magnetic resonance imaging. Oliver Meynberg et al. [20] have applied two classification methods: Bag-of-words (BoW) and features based on a Gabor filter bank to automatically detect the crowd in aerial images. Bharati et al. [21] have provided an overview of GLCM, multivariate statistical techniques primarily based totally on PCA and PLS, and wavelet texture analysis methods for image texture analysis. Bannigidad et al. [22] have proposed a technique to extract texture features and used a K-NN classifier to segregate diseased and healthy DIARETDB0 database images. Zhang et al. [23] have proposed a PNN classifier used to classify polarimetric SAR images and produced a novel algorithm. Bannigidad et al. [24] have proposed the extraction of distinct characters from historical handwritten Kannada scripts using the HOG feature extraction method and applied K-NN, SVM, and LDA classifiers for the identification of dynasties. Kumar et al. [25] have proposed a cost-effective technique to rapidly detect microorganisms using image processing parameters and classified them accurately using the PNN classifier. Deepa et al. [26] experimented with contourlet coefficient co-occurrence matrix features for the analysis and classification of mammography images using CCCM and PNN classifiers. R. Lavanyadevi et al. [27] proposed an automatic brain tumor stage classification of MRI images using a probabilistic neural network (PNN). Patel et al. [28] suggested the classification of limestone rock using a PNN classifier by extracting the texture features of skewness and kurtosis. Aliyana et al. [29] applied SVM, K-NN, decision tree, and naive Bayes machine learning models on FESEM images to identify the ammonium contents. Parashuram Bannigidad et al. [30] experimented with recognizing the effect of time on anodized Al<sub>2</sub>O<sub>3</sub> nanopore FESEM image.

**Table 1** The details of chemical compositions for the preparation of boron, iron, and silver nanoparticles

| Synthesis of various nanoparticles |    | Concentration (mM) | Time (min) | Temperature (°C) | pH  |
|------------------------------------|----|--------------------|------------|------------------|-----|
| Boron                              | A1 | 1                  | 240        | 1073             | –   |
|                                    | A2 | 1                  | 240        | 1073             | –   |
|                                    | A3 | 1                  | 240        | 1073             | –   |
|                                    | A4 | 1                  | 240        | 1073             | –   |
|                                    | A5 | 1                  | 240        | 1073             | –   |
| Iron                               | B1 | 1                  | 60         | 300              | –   |
|                                    | B2 | 1                  | 60         | 300              | –   |
|                                    | B3 | 1                  | 60         | 500              | –   |
|                                    | B4 | 1                  | 60         | 500              | –   |
|                                    | B5 | –                  | 60         | 300              | –   |
| Silver                             | C1 | 1                  | 30         | 90               | 3   |
|                                    | C2 | 1                  | 30         | 90               | 5.3 |
|                                    | C3 | 1                  | 30         | 90               | 6   |
|                                    | C4 | 1                  | 30         | 90               | 10  |
|                                    | C5 | 1                  | 30         | 90               | 7   |

## 2 Materials and Methods

The experimentation is done on nanoparticle images of boron, iron, and silver. The FESEM and TEM images of these nanoparticles undergo various image processing techniques, such as preprocessing, segmentation, feature extraction, and classification. For the purpose of experimentation, a total of 330 nanoparticle images of both FESEM and TEM are considered, out of which the training phase consists of a total of 180 images including both FESEM and TEM; 60 boron, 60 iron, and 60 silver. The testing phase includes 150 images of both FESEM and TEM; 50 boron, 50 iron, and 50 silver. In order to prepare the nanoparticle images, the synthesization process is required. Each FESEM and TEM nanoparticle images undergo a unique synthesis process for the determination of nanoparticles. Table 1 shows the chemical composition used for the preparation of boron, iron, and silver nanoparticles.

## 3 Proposed Method

The proposed algorithm is implemented on both FESEM and TEM nanoparticle images of boron, iron, and silver in two phases; the training phase and the testing phase. The training phase includes 180 FESEM and TEM images: 60 boron, 60 iron, and 60 silver. The testing phase includes 150 FESEM and TEM images: 50 boron,

50 iron, and 50 silver, respectively. Feature extraction is carried out on the training dataset and is stored in the database as a knowledge base. Further, the same features are extracted from the testing dataset also, and using the knowledge base from the feature vector, the FESEM and TEM nanoparticle images are classified using PNN and K-NN classifiers. Finally, the image performance analysis is carried out by calculating the performance measuring parameters, such as accuracy, precision, and recall.

### **Algorithm 1: Training Phase**

Step 1: Input FESEM and TEM images of boron, iron, and silver nanoparticle from the training set.

Step 2: Convert to a gray image.

RGB channel values are converted to grayscale values by making a weighted totality of the R, G, and B components:  $\text{gray} = 0.2989 * R + 0.5870 * G + 0.1140 * B$ .

Step 3: Remove the bottom information line.

Step 4: Binaries the image to get the mask.

Gray image converted to a binary image by using thresholding.

Step 5: Remove the non-pore part using a mask.

Step 6: Calculate the mean, kurtosis(k), skewness(s), and entropy(e) which has been described in Sect. 3.1.

Step 7: Store the extracted features in the vector form as a knowledge base.

Step 8: Stop.

### **Algorithm 2: Testing Phase**

Step 1: Input FESEM and TEM images of boron, iron, and silver nanoparticle from the testing set.

Step 2: Convert to a gray image.

RGB channel values are converted to grayscale values by making a weighted totality of the R, G, and B components:  $\text{gray} = 0.2989 * R + 0.5870 * G + 0.1140 * B$ .

Step 3: Remove the bottom information line.

Step 4: Binaries the image to get the mask.

Gray image converted to a binary image by using thresholding.

Step 5: Remove the non-pore part using a mask.

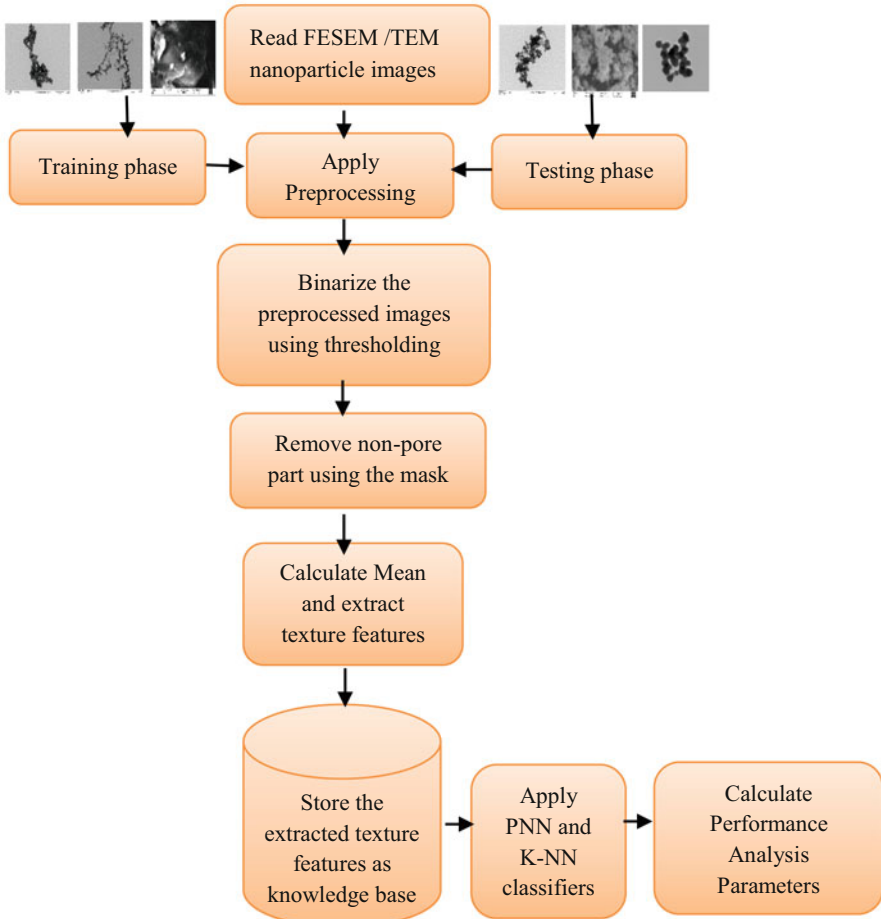
Step 6: Calculate the mean, kurtosis(k), skewness(s), and entropy(e).

Step 7: Performed classification using PNN and K-NN [31] classifiers.

Step 8: Finally, calculate performance analysis parameters for measuring the quality of the nanoparticles.

Step 9: Stop.

The flow diagram of the proposed set of rules is given in Fig. 2.



**Fig. 2** The flow diagram of the proposed study for calculating performance analysis and applying PNN and K-NN classifiers

### 3.1 Feature Extraction and Classification

The suggested work comprises classifying algorithms that investigate textural properties. The suggested approach extracts a variety of texture properties from a grayscale image, including its kurtosis, skewness, and entropy. Kurtosis is a statistical degree that defines how a whole lot of the tails of a distribution range from the one of regular distribution. In other words, it suggests whether or not the tails of a selected distribution include excessive values. Skewness measures the symmetry more specifically the asymmetry of the given data. Another image texture feature is entropy, which measures the content of an image. Entropy is

also a corresponding intensity level to which individual pixels can adjust. In this experiment, various classifiers such as PNN and K-NN have been tested. It is observed that the Probabilistic Neural Network (PNN) produced better results in classification as compared to the K-NN classifier.

The texture features are used in the present algorithm can be weighed using the following Eqs. (1), (2), and (3):

$$\text{Kurtosis } k = \sum \frac{(Ii - \mu)^4}{n\sigma^4} \quad \text{where } \sigma - \text{standard deviation} \quad (1)$$

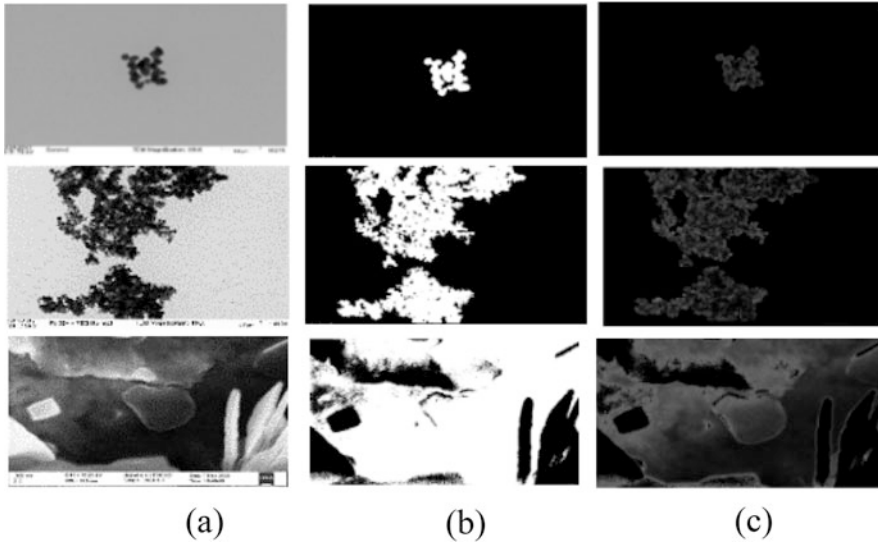
$$\text{Skewness } s = \sum \frac{(Ii - \mu)^3}{n\sigma^3} \quad (2)$$

$$\text{Entropy } e = \text{sum } (p \cdot \log_2(p)) \quad \text{where } p = \text{normalized histogram counts} \quad (3)$$

## 4 Experimental Results and Discussion

The experimentation is carried out on FESEM and TEM nanoparticle images (Fig. 3a). The images of these nanoparticles are obtained using the synthesis and characterization process done in the Department of Chemistry, Rani Channamma University, Belagavi, and images are obtained from IISC, Bangalore. MATLAB R2018a software is used to develop the segmentation of nanoparticles of boron, iron, and silver for extracting the texture features, and classification is completed by using PNN and K-NN classifiers and evaluating the quality of images, and computing performance measuring parameters. In phase I (training), the preprocessing techniques are adopted such as the removal of unwanted background, converting images to gray images (Fig. 3b), binarization of the images (Fig. 3c) by thresholding to get the mask and calculation of mean, and finally extracting the texture feature, namely, kurtosis, skewness, and entropy. Based on these texture features, the images are classified by using PNN and K-NN classifiers. In phase II (testing), same preprocessing techniques are applied to the testing data set as used in phase I, and the texture features are extracted from all the nanoparticle images. Finally, the performance assessment of the nanoparticle images is carried out by calculating the accuracy, precision, and recall.

Table 2 depicts the classification results of boron, iron, and silver nanoparticle images using K-NN and PNN classifiers. To measure the performance of each nanoparticle of both FESEM and TEM by considering the images, and the performance measuring parameters. Table 3 illustrates the comparison results of classification based on performance evaluation measures, i.e., accuracy (AC), precision (Pr), and recall (Re) are given in the Eqs. (4), (5), and (6).



**Fig. 3** Sample images of the suggested algorithm: (a) original sample nanoparticle images of boron, iron, and silver, (b) grayscale nanoparticle images, and (c) binarized nanoparticle image

**Table 2** The classification results of boron, iron, and silver nanoparticle images using K-NN and PNN classifiers

| Classifiers | Nanoparticle images | Classification results |    |    |    |
|-------------|---------------------|------------------------|----|----|----|
|             |                     | TP                     | TN | FP | FN |
| K-NN        | Boron               | 05                     | 03 | 00 | 07 |
|             | Iron                | 03                     | 00 | 02 | 10 |
|             | Silver              | 04                     | 00 | 01 | 10 |
| PNN         | Boron               | 05                     | 02 | 00 | 08 |
|             | Iron                | 04                     | 00 | 01 | 10 |
|             | Silver              | 04                     | 00 | 01 | 10 |

*TP* true positives, *TN* true negatives, *FP* false positives, *FN* false negatives

**Table 3** Comparison results of classification based on performance evaluation measures

| Nanoparticle images | K-NN   |      |      | PNN    |      |      |
|---------------------|--------|------|------|--------|------|------|
|                     | Ac (%) | Pr   | Re   | Ac (%) | Pr   | Re   |
| Boron               | 80.00  | 0.63 | 1.0  | 86.67  | 0.71 | 1.00 |
| Iron                | 86.67  | 1.0  | 0.60 | 93.33  | 1.00 | 0.80 |
| Silver              | 93.33  | 1.0  | 0.80 | 93.33  | 1.00 | 0.80 |

*Ac* accuracy, *Pr* precision, *Re* recall

The working of the proposed methodology is assessed based on different measurement measures such as accuracy (AC), precision (Pr), and recall (Re).

- Accuracy: It refers to the number of properly identified images, whether the images belong to the correct class or not.

$$Ac = \frac{(TP + TN)}{(TP + TN + FP + FN)} * 100 \tag{4}$$

- Precision: Precision is the percentage of applicable images that are classified among the examples that were retrieved.

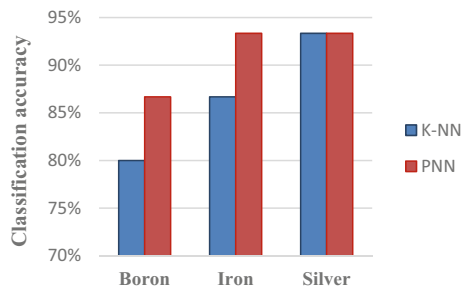
$$Pr = \frac{(TP)}{(TP + FP)} \tag{5}$$

- Recall: It is the capacity of a classifier to properly locate all positive images.

$$Re = \frac{(TP)}{(TP + FN)} \tag{6}$$

It is observed from the experimentation that, the class accuracy of the proposed approach by means of the K-NN classifier for boron images is 80.00%, for iron images is 86.67%, and for silver is 93.33%. Similarly, the classification accuracy using the PNN classifier for boron nanoparticle images is 86.67%, for iron, it is 93.33%, and for silver, it is 93.33%. Figure 4 displays the BAR chart, which compares the performance of K-NN and PNN classifiers based on the texture features. Hence, for the classification ability, as compared to the K-NN classifier, the PNN classifier has yielded good results for boron, iron, and silver nanoparticle images. Tables 4 and 5 show the resulting confusion matrices of K-NN and PNN classifiers, respectively.

**Fig. 4** Comparative analysis of K-NN and PNN classifiers



**Table 4** Confusion matrix for K-NN classifier

| Nanoparticle images | Boron | Iron | Silver | Total |
|---------------------|-------|------|--------|-------|
| Boron               | 50    | 00   | 00     | 50    |
| Iron                | 20    | 30   | 00     | 50    |
| Silver              | 10    | 00   | 40     | 50    |

**Table 5** Confusion matrix for PNN classifier

| Nanoparticle images | Boron | Iron | Silver | Total |
|---------------------|-------|------|--------|-------|
| Boron               | 50    | 00   | 00     | 50    |
| Iron                | 10    | 40   | 00     | 50    |
| Silver              | 10    | 00   | 40     | 50    |

## 5 Conclusion

The purpose of the proposed study is to measure the size and shape of the various metal and metal oxide nanoparticles, namely, boron, iron, and silver. The FESEM and TEM images are used for experimentation of these nanoparticles and are obtained from the Chemistry Department, Rani Channamma University, Belagavi, and IISC Bangalore. The proposed technique uses two phases; training and testing. In the training phase, the preprocessing techniques are applied, the calculation of the mean for nanoparticle images is carried out, and finally extracted the texture features, namely, kurtosis, skewness, and entropy from nanoparticle images. Based on these texture features, the images are classified by using PNN and K-NN classifiers. In the testing phase, the same preprocessing techniques are operated on to the testing data set as used in the training phase, followed by the texture features extraction of the boron, iron, and silver nanoparticle images. Finally, the performance analysis of the nanoparticle images is carried out by calculating the accuracy, precision, and recall. The K-NN classifier has an accuracy of 80.00% for boron, 86.67% for iron, and 93.33% for the silver nanoparticle images, and the PNN classifier has an accuracy of 86.67% for boron, 93.33% for iron, and 93.33% for silver nanoparticle images. Hence, based on the experimentation, the proposed study suggested that the PNN classification with texture features is the best classifier used to classify the boron, iron, and silver nanoparticle images as compared to the K-NN classifier. The same algorithm may be used for other types of metal and metal oxide nanoparticle images, which will be done in our forthcoming work.

**Acknowledgments** The authors are grateful to the “Karnataka Science and Technology Promotion Society (KSTEPS), DST, GOVERNMENT OF KARNATAKA” for the monetary assistance and for approving a Ph.D. fellowship to carry out this research work.

## References

1. A. Alyamani, O. Lemine, FE-SEM characterization of some nanomaterial, in *Scanning Electron Microscopy*, ed. by V. Kazmiruk, (IntechOpen, 2012)
2. M. Havrdova, K. Polakova, J. Skopalik, M. Vujtek, A. Mokdad, M. Homolkova, J. Tucek, J. Nebesarova, R. Zboril, Field emission scanning electron microscopy (FE-SEM) as an approach for nanoparticle detection inside cells. *Micron* **67**, 149–154 (2014)
3. D.J. Smith, Characterization of nanomaterials using transmission electron microscopy, in *Nanocharacterisation* (2015), pp. 1–29 <https://doi.org/10.1039/9781782621867-00001>



4. Z. Sun, J. Shi, J. Wang, M. Jiang, Z. Wang, X. Bai, X. Wang, A deep learning-based framework for automatic analysis of the nanoparticle morphology in SEM/TEM images. *Nanoscale* **14**, 10761–10772 (2022)
5. F.H. Nielsen, The saga of boron in food: From a banished food preservative to a beneficial nutrient for humans. *Curr. Top. Plant Biochem. Physiol.* **10**, 274–286 (1991)
6. A. Wittig, J. Michel, R.L. Moss, F. Stecher-Rasmussen, H.F. Arlinghaus, P. Bendel, P.L. Mauri, S. Altieri, R. Hilger, P.A. Salvadori, L. Menichetti, Boron analysis and boron imaging in biological materials for boron neutron capture therapy (BNCT). *Crit. Rev. Oncol. Hematol.* **68**(1), 66–90 (2008)
7. B. Parashuram, P. Namita, G. Prabhuodeyra, A. Lakkappa, Boron nanoparticle image analysis using machine learning algorithms. *J. Adv. Appl. Sci. Res.* **4**, 28–37 (2022)
8. N. Ajinkya, Y. Xuefeng, P. Kaithal, H. Luo, P. Somani, S. Ramakrishna, Magnetic iron oxide nanoparticle (IONP) synthesis to applications: Present and future. *Materials* **13**, 2–35 (2020)
9. B.J. Calderón-Jiménez, E.M. Monique, A.R. Bustos, E. Murphy Karen, R. Winchester Michael, V. Baudrit, R. Jose, Silver nanoparticles: Technological advances. Societal impacts, and metrological challenges. *Front. Chem.* **5**, 6 (2017)
10. Q. Zhao, C.Z. Shi, L.P. Luo, Role of the texture features of images in the diagnosis of solitary pulmonary nodules in different sizes. *Chin. J. Cancer Res.* **26**(4), 451–458 (2014)
11. L. Armi, S. Fekri-Ershad, Texture image analysis and texture classification methods- A review. *Int. Online J. Image Process. Pattern Recognit.* **2**(1), 1–29 (2019)
12. C.-C. Hung, E. Song, Y. Lan, *Image Texture Analysis (Foundations, Models and Algorithms), Image Texture, Texture Features, and Image Texture Classification and Segmentation* (Springer, Cham, 2019), pp. 3–14
13. F. Long, H. Zhang, D.D. Feng, Fundamentals of content-based image retrieval, in *Multimedia Information Retrieval and Management. Signals and Communication Technology*, (Springer, Berlin, 2003), pp. 1–26
14. W.J. Wang, D. Hoi, S.C. Hong, W. Pengcheng, Z. Jianke, Z. Yongdong, L. Jintao, Deep learning for content-based image retrieval, in *Proceedings of the ACM International Conference on Multimedia*, (ACM, 2014), pp. 157–166
15. P.A. Charde, S.D. Lokhande, Classification using K nearest neighbor for brain image retrieval. *Int. J. Sci. Eng. Res.* **4**(8), 760–765 (2013)
16. Z. Dengsheng, *Texture Feature Extraction, Fundamentals of Image Data Mining: Analysis, Features, Classification and Retrieval* (Springer, Cham, 2019), pp. 81–111
17. A. Ramola, A.K. Shakya, D. Van Pham, Study of statistical methods for texture analysis and their modern evolutions. *Eng. Rep.* **2**(4), 1–24 (2020)
18. P. Kupidura, The comparison of different methods of texture analysis for their efficacy for land use classification in satellite imagery. *Remote Sens.* **11**(10), 1–20 (2019)
19. A. Materka, Texture analysis methodologies for magnetic resonance imaging. *Dialogues Clin. Neurosci.* **6**(2), 243–245 (2004)
20. O. Meynberg, S. Cui, P. Reinartz, Detection of high-density crowds in aerial images using texture classification. *Remote Sens.* **8**(6), 470 (2016)
21. M.H. Bharati, J. Jay Liu, J.F. MacGregor, Image texture analysis: Methods and comparisons. *Chemom. Intell. Lab. Syst.* **72**, 57–71 (2004)
22. P. Bannigidad, A. Deshpande, A multistage approach for exudates detection in fundus images using texture features with K-nn classifier. *Int. J. Adv. Res. Comput. Sci.* **9**(1), 755–759 (2018)
23. Y.-D. Zhang, L. Wu, N. Neggaz, S. Wang, G. Wei, Remote-sensing image classification based on an improved probabilistic neural network. *Sensors* **9**, 7516–7539 (2009)
24. P. Bannigidad, C. Gudada, Historical Kannada handwritten character recognition using machine learning algorithm, in *Proceedings of the 12th International Conference on Soft Computing and Pattern Recognition (SoCPaR 2020)*, (Springer International Publishing, Cham, 2021), pp. 311–319
25. S. Kumar, G.S. Mittal, Rapid detection of microorganisms using image processing parameters and neural network. *Food Bioprocess Technol.* **3**, 741–751 (2010)

26. S. Deepa, V. Subbiah Bharathi, Textural feature extraction and classification of mammogram images using CCCM and PNN. *IOSR J. Comput. Eng. (IOSR-JCE)* **10**(6), 7–13 (2013)
27. R. Lavanyadevi, M. Machakowsalya, J. Nivethitha, A. Niranjil Kumar, Brain tumor classification and segmentation in MRI images using PNN, in *IEEE International Conference on Electrical, Instrumentation, and Communication Engineering (ICEICE)*, (IEEE Press, Karur, 2017), pp. 1–6
28. A.K. Patel, S. Chatterjee, Computer vision-based limestone rock-type classification using probabilistic neural network. *Geosci. Front. Prog. Mach. Learn. Geosci.* **7**(1), 53–60 (2016)
29. A.K. Aliyana, S.K. Naveen Kumar, P. Marimuthu, Machine learning-assisted ammonium detection using zinc oxide/multi-walled carbon nanotube composite based impedance sensors. *Sci. Rep.* **11**, 24321 (2021)
30. P. Bannigidad, C.C. Vidyasagar, Effect of time on anodized Al<sub>2</sub>O<sub>3</sub> nanopore FESEM images using digital image processing techniques: A study on computational chemistry. *Int. J. Emerg. Trends Technol. Comput. Sci. (IJETTCS)* **4**, 15–22 (2015)
31. <https://machinelearningmastery.com/k-nearest-neighbors-for-machine-learning>, last accessed 30 Nov 2022

# Efficient Implementation to Reduce the Data Size in Big Data Using Classification Algorithm of Machine Learning



V. RajKumar and G. Priyadharshini

## 1 Introduction

Big data analytics is the practice of applying sophisticated analytic methods to extremely large and extensive data sets. These data sets may be organized, semi-structured, or unstructured, and they may come from a wide variety of sources and range in size from terabytes to zettabytes. Exactly, what is a considerable amount of data? It will be defined as data sets whose size or type is beyond the ability of existing relative databases to gather, maintain, and process information with minimal latency. Massive data has characteristics such as high volume, high speed, and high selection. Because of computing (AI), mobile devices, social media, and, thus, the Internet of Things, information sources have become far more intricate than those for historical data (IoT). Various data types, for instance, emanate from sensors, devices, video/audio, networks, log files, transactional applications, the Internet, and social media; a substantial amount of it is generated in real time and at a massive scale.

Increased business intelligence, the ability to model and anticipate future outcomes, and the speed and accuracy with which decisions may be made all result from the analysis of large amounts of data. Open source software like Apache Hadoop and Spark, and by extension the entire Hadoop system, are powerful, flexible processing and storage solutions made to handle the massive amounts of data being generated today. Machine learning could be a branch of computing (AI) and applied science that focuses on employing data and algorithms to imitate how humans learn, bit by bit, raising its accuracy. The area of data science, of which machine learning is a subset, is rapidly expanding. Algorithms are trained to make

---

V. RajKumar (✉) · G. Priyadharshini  
Krishnasamy College of Engineering and Technology, Cuddalore, India

classifications or predictions by employing applied mathematics methodologies, which leads to the discovery of crucial insights during data processing. These discoveries then inform choice making across apps and businesses, with the best-case scenario being an effect on key growth KPIs. The need for information scientists to assist in determining the most pressing business questions and, subsequently, the data to address those questions, may grow as the volume of available data continues to explode. Machine learning algorithms typically divide the learning system into three distinct parts:

- **Decision-making:** Algorithms used in machine learning are often units that are tasked with the creation of a prediction or classification. Supported by some computer files, which may be labeled or unlabeled, the algorithmic rule can manufacture an Associate in nursing estimate a couple of patterns within the information.
- **A fault function:** A fault operation serves to gauge the prediction of the model. If their area unit identified examples, a slip operation would create a comparison to assess the model's accuracy.
- **Optimization process model:** If the model is a better fit to the data in the training set, the weights are tweaked to close the gap between the observed value and the predicted value. Once a certain level of accuracy has been reached, the algorithmic rule can perform this evaluation again, optimize the chosen approach, and apply new weights.

### ***1.1 Machine Learning Classifiers Comprise Three Primary Structures of Data Supported***

This article will focus on supervised machine learning, often known as supervised learning, which is defined by the use of labeled data sets to teach computers to correctly categorize data or make predictions. To ensure a perfect fit, the model's weights are dynamically adjusted when the computer file is input. This is done in the cross-validation procedure to ensure that the model does not succumb to overfitting or underfitting. Supervised learning is a powerful tool for helping businesses address a wide variety of real-world problems at scale, such as separating spam messages into a very specific folder in the inbox. Various methods, including as neural networks, naive Bayes, mean regression, supply regression, random forest, support vector machine (SVM), and others, are used in supervised learning.

*Unsupervised machine learning* Unattended learning, also known as unattended machine learning, is the process of doing research and clustering on unlabeled datasets with the aid of machine learning algorithms. Without any help from a human, these algorithms are able to unearth previously unknown relationships between pieces of data. Its ability to find similarities and variations in info create it the perfect answer for explorative information analysis, cross-selling ways,

client segmentation, and image and pattern recognition. Spatiality reduction is also used to reduce the number of variables in a model; principal component analysis (PCA) and singular value decomposition (SVD) are two typical techniques for this. Unattended learning uses a wide variety of techniques, including those based on neural networks, k-means agglomeration, probabilistic agglomeration schemes, and others.

*Semi-supervised learning* Learning that is semi-supervised provides a happy medium between learning that is supervised and learning that is unattended. Throughout the process of coaching, it utilizes a lesser tagged information set as a guide for categorization, and it contains extraction from a significantly more important unlabeled information set. Semi-supervised learning will solve the problem of needing more tagged information (or not having the ability to afford to label enough data) to coach a supervised learning algorithmic rule.

*Reinforcement machine learning* Reinforcement machine learning may be an activity machine learning model that's almost like supervised learning; however, the algorithmic rule is not trained for the mistreatment of sample information. This model is able to learn as a result of going through the process of trial and error. It is planned to improve a certain chain of results in order to come up with the most useful advice or plan of action for a particular shortcoming.

## ***1.2 Benefits of Huge Information Analytics***

*Faster, higher call-making* In order for businesses to discover fresh insights and take action, they will access an excessively huge volume of data and conduct analysis on a wide variety of data sources. Begin on a small scale, and as your needs grow, expand to manage information from historical records and across time.

*Cost reduction and operational efficiency* The availability of flexible processing and storage technologies will make it easier for businesses to reduce the costs associated with storing and analyzing massive amounts of data. Learn the trends and obtain the insights that will assist you in building your business in a timely manner.

*Improved information-driven visit market* The ability to be data-driven can be given to a corporation through the analysis of data obtained through sensors, devices, video, logs, transactional apps, the Internet, and social media. Determine what the customers want, and if there are any potential dangers, then create new products.

In Sect. 2, describe in detail the terminology associated with our proposed system. In Sect. 3, we study different techniques and methods for designing the architecture of our proposed new work BDML system. In Sect. 4, we give a brief summary of the area of research that we are concentrating on, along with an example of the outcomes that are exhibited using a decision tree to categorize credit card

fraud detection. The purpose of Sect. 5 is to exhibit the outcome analysis of the activities that were done in order to reach the goal of the research. In conclusion, our study is summarized in Sect. 6, and the linked explanations of how the research will be improved in the future are included.

## 2 Related Works

Sanjog Kumar Panda, Syed Mohd Ali, and Minerva Panda [7] addressed how data from different real-time integration of sources is possible. The writer also identifies the causes of the rise in medical costs. You have created a helpful mobile/web application physicians and patients may communicate. Wang et al. [17], big data implementation scenarios in healthcare have been examined.

Furthermore, listed five significant data analytics achievements described a few advantages of big data analytics, including operational, managerial, and infrastructural recommendations approaches for implementing big data in health organizational technology for analysis. Kharbouch et al. [16] suggested a system for ongoing monitoring and utilizing big data in conjunction with IoT to process data in real-time technology. The authors' preliminary research and the results of the trials suggest that this platform is usable in an actual situation. Ibrahim et al. [5] researched and had a conversation about the essential requirements for handling big data analytics and allowing methods for Internet of Things systems. The authors have identified the opportunities that are created when big data analytics are combined with the Internet of Things. The writer continues by emphasizing the significance of big data analytics in IoT applications and offering a wide range of susceptibility challenges.

Big data was described by Kumar et al. [15] and [14] as being vast. Data becomes challenging for traditional programs to examine. Using information mining methods, diabetic patients are analyzed in this paper. In order to carry out the study, the data mining techniques of decision tree (DT) and Naive Bayes, in conjunction with Hadoop and MapReduce, are utilized [3]. Namrata Bhattacharya et al. [10] study the healthcare industry, which requires extensive data analysis. Information mining methods were used to weed out irrelevant data and extract crucial instances. ARM is a standard-rooted method that shows how things relate. Here, the AA deteriorates due to the massive data. Here, a Hadoop MapReduce execution of an AA was provided. Younus et al. [9] and [11] suggested ML methods such as DT and RF to create prediction representations based on categorization systems to assess and classify chronic DM illnesses. Moreover, a method based on RF was provided to identify challenging areas with type 2 diabetes. Pani et al. [6] looked into two methods for logistic regression (LR) and SVM classifier training.

Sampath et al. [12] employed the analytical investigation methodology. Diabetes in the Hadoop/MapReduce backdrop problems and classification of treatments. A modest microbial wealth was described by Costantini et al. [2]. And consistency

started in the high-risk individual's pharyngeal microbiome patients. An invasive fungal infection in mice (IFI) provided biological support for the decision that was rejected [13]. The presence of protective anaerobes such as Clostridiales and Bacteroidetes, with tryptophan's apparent low availability, in hematologic patients is directly connected to the IFI risk and defines pathways for metabolic and antibacterial stewardship re-equilibrium in the IFI.

Zainab Alansari [18] provided to use big data in IoT to give the systematical solution. The best solution in an IoT is big data tools. It contains people who can get real-time data and information using the Internet of Things (IoT). Beom-Su Kim [1] provided deeper insight can lead to lacking, so address this gap by introducing a wireless sensor network (WSN) in the extensive data system. WSN can use to gather large-scale data from the different sensor nodes. Mohsen Marjani [8] introduced a new architecture of big data used in the Internet of Things, and it helps to improve the IoT devices to generate extensive data. Numerous notable use cases can be discussed, as well as analytic methods and technologies for extensive data mining. The new technology architecture applied accurate time analysis, offline analytics, BI analytics, and massive analytic. Deepa N [4] is a study to discover using blockchain in effective data methods to improve data integration. Using blockchain in big data includes data acquisition, data storage, data analytics and data privacy conservation in this paper analysis to use of big data in various applications because it gives various secure data nowadays, such as smart grids, innovative healthcare, intelligent transportation, and smart city.

### 3 System Architecture

In the system, the framework described our research based on big data analytics with a machine learning algorithm. Big data can store and process a large number of datasets, and on the other hand, machine learning works to predict the resulting value by using a classification algorithm and encrypting the data to reduce the stored and processed data in machine learning (Fig. 1).

The big data are collected from the sources like social media and live data. Moreover, data is stored and processed in real-time data and stream of data. The stored datasets classify and encode the binary using a machine learning classifier algorithm model. The classified data process to analyze the data storage and reports are produced, further with the help of machine learning to visualize the resulting outcome of trained and tested data.

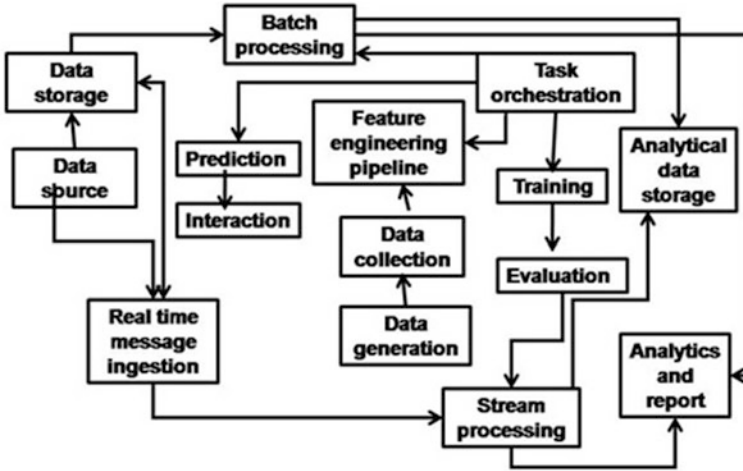


Fig. 1 BDML processing architecture

### 3.1 Big Data Sources

Today’s technology enables us to gather data at a fantastic rate, as well as in terms of quantity and diversity. Data comes from many different sources, and however, in the case of big data, these are the main ones:

*Social networks* The social media networks that have exploded in popularity over the past 5–10 years are perhaps the primary source of big overall data we are aware of today. The vast majority of this data is unstructured and represented by the billions of social media posts and other pieces of information created every second as a result of user interactions throughout the globe. Globally increasing Internet access has contributed to the expansion of information in social networks in a self-fulfilling manner.

*Media* The millions, though perhaps not terabytes, of multimedia and audio uploads that occur daily are primarily due to the development of social networks. Prime examples of material whose density continues to rise unchecked include videos posted on YouTube, music tracks on SoundCloud, and images shared on Instagram.

*Data warehouses* Organizations have long invested in specialized data storage spaces known as data warehouses. A DW is a collection of historical data that businesses want to save and organize for quick retrieval, whether for internal usage or regulatory needs. More and more businesses are migrating data out of their old data warehouses and onto some of the more recent technologies as sectors steadily move toward the practice of keeping data in platforms like Hadoop and NoSQL. Emails from businesses, accounting information, database, and internal documents



are a few examples of DW data currently offloaded onto Hadoop-like systems, which use numerous nodes to produce a highly available and failed platform.

*Sensor* Data collecting through sensor devices is a relatively recent phenomenon in big data. Although sensors have always been around, and sectors like the oil and gas industry have used exploration sensors for measurement methods at oil rigs for many years, the introduction of wearable technology, also known as the IoT, and products like the Fitbit and Apple Watch meant that now each person could stream data within the same rate as a few oil rigs did just 10 years ago.

At any given time, wearable technology may take hundreds of measures from a person. While not currently a big data issue per se, sensor-related data is expected to resemble the type of haphazard data created on the Internet during social network activities as the business continues to develop.

### ***3.2 Data Storage***

You may utilize big data storage, a compute-and-storage architecture, to gather and handle massive datasets and conduct real-time data analysis. The intelligence derived from the metadata may then be produced using these analyses. Due to the cheaper cost of the medium, hard disk drives are typically used for massive data storage. However, flash storage is becoming more and more common due to falling prices. When employed, systems can be constructed entirely on flash media or can be created as a hybridization of flashes and disk storage. Extensive data databases include unstructured data. Big data retention is typically constructed using file- and object-based storage to account for this. These storage formats can expand up to quadrillion or petabyte levels and are not limited to any particular capabilities.

### ***3.3 Stream Processing***

A software paradigm known as “stream processing,” commonly referred to as “data streaming,” ingests, maintains, and processes continuous streams of data while they are still in motion. The capacity to empower data as it is created has become essential to the success of the modern environment since data is seldom static. Current data processing has advanced from live data stream processing to historical batch data processing. Similar to how they used to wait for the complete movie or music to download, customers nowadays monitor and record things like movies on Netflix or songs on Spotify. In the era of big data, the capacity to handle data streams in real time is crucial. Continue reading to discover more about the benefits of stream processing for real-time analysis and data intake.

How should data streaming operating? Because there were fewer sources of data generation in the legacy infrastructure and the system could be designed to identify

and combine the information and structures, the infrastructure was considerably more organized.

It is nearly impossible to manage or enforce the data model or manage the volume and probability of the data generated in the modern world because it is produced by an infinite number of sources, including specialized hardware, servers, portable devices, software, web browsers, internal systems, and external systems. Applications that examine and handle data streams must process data packets one at a time sequentially. Each data packet created will have the source and date for applications to use data streams. Applications that operate with streaming data will always need two significant components: processing and storage. Collection must be capable of sequentially and consistently recording significant streams of data. Processing has to be able to communicate with storage, ingest data, analyze it, and do calculations on it. When dealing with data streams, this also raises new issues and concerns. Currently, several platforms and technologies are available to assist businesses in creating streaming data applications.

### ***3.4 Machine Learning Models***

Linear regression is one technique in supervised machine learning which can give predictive output in a continuous constant slope. Linear regression contains two types of regression such as simple regression and multi variable regression. Logistic regression is one of the machine learning algorithms. A logistic algorithm can solve more complex cost functions defined as “sigmoid functions.” It contains a limited value between 0 and 1. Logistic regression can return the value in the probabilistic concept using a predictive analysis algorithm.

Decision trees can explain the input we should provide and the conterminous output we get from a set of training data. It should split data continuously according to two nodes: a decision node (root node) and a leaves node (child node). For example, color is the decision node containing two leaves nodes; one is red, and the other is blue. Further, the node blue split two nodes (dark blue and sky blue). Support vector machine is one of the best segregation in two data using hyperplane line. This algorithm gives an accurate classification or regression technique in machine learning. We plot the data in  $n$ -dimensional spaces in the SVM algorithm to get crucial output in machine learning.

Naïve Bayes can use Bayes’ theorem to solve classifiers problems in machine learning. Naïve Bayes is the most effective classification algorithm that helps construct a fast machine-learning model to make a quick prediction. Naïve Bayes can predict an object’s probability based on a probabilistic classifier.

The KNN algorithm takes more time to perform actions, also called the lazy learning algorithm. KNN can be a straightforward way to implement when new data can occur, and it can check the already existing data to which one is nearest or similar to new data to allocate the value. The K-means clustering algorithm is an

unsupervised algorithm that separates the groups of unlabeled data that can have a different cluster of data.

A random forest algorithm is one of the algorithms in the supervised algorithm. It can use classification and regression techniques, but most of its use is the classification algorithm. A forest containing more similar trees can take place random forest algorithm. Many decision trees combine to select which one is giving the best solution to choose based on voting. It is better than using a single decision tree because it averages the result.

## 4 Methodology

The Apache Spark group created PySpark, a Python API for Spark, to enable Python with Spark. One may integrate and work with RDDs in the Python programming language by using PySpark. When it comes to handling enormous datasets, PySpark is an outstanding framework thanks to a variety of capabilities. Data engineers use this technology, whether it is to evaluate enormous datasets or to conduct calculations on them.

PySpark Real-Time Computations' Key Features Low latency is demonstrated by the PySpark framework since it processes data in memory.

*Polyglot* The PySpark architecture can analyze enormous datasets since it is interoperable with several other languages, including Scala, Java, Python, and R.

This framework provides powerful caching and outstanding disk durability.

*Processing data quickly* The PySpark framework for processing big data is much quicker than conventional frameworks.

Python is a dynamically typed programming language helpful in working with RDDs. A function that translates input to output is learned through supervised learning using sample input-output pairs. It uses labeled training data, a collection of training samples, to infer a function. Each example in supervised learning consists of a pair, one of which is an input object (usually a vector), and the other is the intended output value (also called the supervisory signal). An inferred function is created by a supervised learning algorithm analyzing the training data and may be applied to mapping new cases. The algorithm will be able to accurately determine the class labels for instances that still need to be visible in an ideal environment.

The following are some of the primary contexts in which categorization cases are used:

- To determine if an email is spam or not.
- To determine consumer segmentation.
- To determine if a loan from the bank is approved.
- To predict if a child will be successful or unsuccessful in a test.

We illustrate how the workings are done in the [kaggle.com](https://www.kaggle.com) online tools to classify the huge dataset in credit card fraud detection using a decision tree classification algorithm in machine learning. The data source is collected in the online credit card fraud detection process. This process is called data preprocessing.

Step 1: Import library function relevant application based and common library functions in machine learning are numpy, panda and matplotlib.

Step 2: Importing the CSV file.

Step 3: Read and store content of a CSV file.

Step 4: Observe the data table // df.head(6).

```
df.head(6)
```

|   | Time | V1        | V2        | V3       | V4        | V5        | V6        | V7        | V8        |
|---|------|-----------|-----------|----------|-----------|-----------|-----------|-----------|-----------|
| 0 | 0.0  | -1.359807 | -0.072781 | 2.536347 | 1.378155  | -0.338321 | 0.462388  | 0.239599  | 0.098698  |
| 1 | 0.0  | 1.191857  | 0.266151  | 0.166480 | 0.448154  | 0.060018  | -0.082361 | -0.078803 | 0.085102  |
| 2 | 1.0  | -1.358354 | -1.340163 | 1.773209 | 0.379780  | -0.503198 | 1.800499  | 0.791461  | 0.247676  |
| 3 | 1.0  | -0.966272 | -0.185226 | 1.792993 | -0.863291 | -0.010309 | 1.247203  | 0.237609  | 0.377436  |
| 4 | 2.0  | -1.158233 | 0.877737  | 1.548718 | 0.403034  | -0.407193 | 0.095921  | 0.592941  | -0.270533 |
| 5 | 2.0  | -0.425966 | 0.960523  | 1.141109 | -0.168252 | 0.420987  | -0.029728 | 0.476201  | 0.260314  |

6 rows × 31 columns

```
df.describe()
```

|       | Time          | V1            | V2            | V3            | V4            | V5            |
|-------|---------------|---------------|---------------|---------------|---------------|---------------|
| count | 283726.000000 | 283726.000000 | 283726.000000 | 283726.000000 | 283726.000000 | 283726.000000 |
| mean  | 94811.077600  | 0.005917      | -0.004135     | 0.001613      | -0.002966     | 0.001828      |
| std   | 47481.047891  | 1.948026      | 1.646703      | 1.508682      | 1.414184      | 1.377008      |
| min   | 0.000000      | -56.407510    | -72.715728    | -48.325589    | -5.683171     | -113.743307   |
| 25%   | 54204.750000  | -0.915951     | -0.600321     | -0.889682     | -0.850134     | -0.689830     |
| 50%   | 84692.500000  | 0.020384      | 0.063949      | 0.179963      | -0.022248     | -0.053468     |
| 75%   | 139298.000000 | 1.316068      | 0.800283      | 1.026960      | 0.739647      | 0.612218      |
| max   | 172792.000000 | 2.454930      | 22.057729     | 9.382558      | 16.875344     | 34.801666     |

8 rows × 30 columns

```
df.corr()
```

|      | Time      | V1        | V2        | V3        | V4        | V5        |
|------|-----------|-----------|-----------|-----------|-----------|-----------|
| Time | 1.000000  | 0.117927  | -0.010556 | -0.422054 | -0.105845 | 0.173223  |
| V1   | 0.117927  | 1.000000  | 0.006875  | -0.008112 | 0.002257  | -0.007036 |
| V2   | -0.010556 | 0.006875  | 1.000000  | 0.005278  | -0.001495 | 0.005210  |
| V3   | -0.422054 | -0.008112 | 0.005278  | 1.000000  | 0.002829  | -0.006879 |
| V4   | -0.105845 | 0.002257  | -0.001495 | 0.002829  | 1.000000  | 0.001744  |
| V5   | 0.173223  | -0.007036 | 0.005210  | -0.006879 | 0.001744  | 1.000000  |

Step 5: Identify the spread of the data set. In our example, we can show below  
The size of data set: 284,807 entries, Number of features: 31.

Step 6: Identify number of rows with null values under each column.

Step 7: Check for data duplications.

```
Duplicate Rows :
```

|        | Time     | V1        | V2        | V3        | V4        | V5        | V6        | V7        |
|--------|----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| 32     | 26.0     | -0.529912 | 0.873892  | 1.347247  | 0.145457  | 0.414209  | 0.100223  | 0.711206  |
| 34     | 26.0     | -0.535388 | 0.865268  | 1.351076  | 0.147575  | 0.433680  | 0.086983  | 0.693039  |
| 112    | 74.0     | 1.038370  | 0.127486  | 0.184456  | 1.109950  | 0.441699  | 0.945283  | -0.036715 |
| 113    | 74.0     | 1.038370  | 0.127486  | 0.184456  | 1.109950  | 0.441699  | 0.945283  | -0.036715 |
| 114    | 74.0     | 1.038370  | 0.127486  | 0.184456  | 1.109950  | 0.441699  | 0.945283  | -0.036715 |
| ...    | ...      | ...       | ...       | ...       | ...       | ...       | ...       | ...       |
| 282986 | 171288.0 | 1.912550  | -0.455240 | -1.750654 | 0.454324  | 2.089130  | 4.160019  | -0.881302 |
| 283482 | 171627.0 | -1.464380 | 1.368119  | 0.815992  | -0.601282 | -0.689115 | -0.487154 | -0.303778 |

Step 8: Split the data to create test and training set.

```
Shape of training set: (377670, 29)
Shape of testing set: (188836, 29)
```

Step 9: Decision Tree Classifier.

Step 10: Print scores of the classifiers.

```
Decision Tree Score: 99.78288038297782
```

Step 11: Correlation matrix construct to our dataset.

Step 12: Finally, we evaluate the model to get the accuracy.

```
# Evaluation of the model

print("Accuracy: {:.5f}".format(accuracy_score(Y_test, predictions)))
print("Precision: {:.5f}".format(precision_score(Y_test, predictions)))
print("Recall: {:.5f}".format(recall_score(Y_test, predictions)))
print("F1-score: {:.5f}".format(f1_score(Y_test, predictions)))

Accuracy: 0.99783
Precision: 0.99675
Recall: 0.99892
F1-score: 0.99783
```

## 5 Result Analysis of Our Proposed Work

Figure 2 on data storage and processing graph shows the storage of enormous amounts of data and processing data Analytical and statistical value. Our proposed works' analysis of variance technique have high storage and processing.

Figure 2 represents our research on resolving the analysis of the challenges in the literature review high efficiency with the help of our research on the Big Data with Machine Learning (BDML) technique.

Data standardization is one of the challenges analyzed by the literature review and should be resolved in Big data with machine learning techniques (Fig. 3).

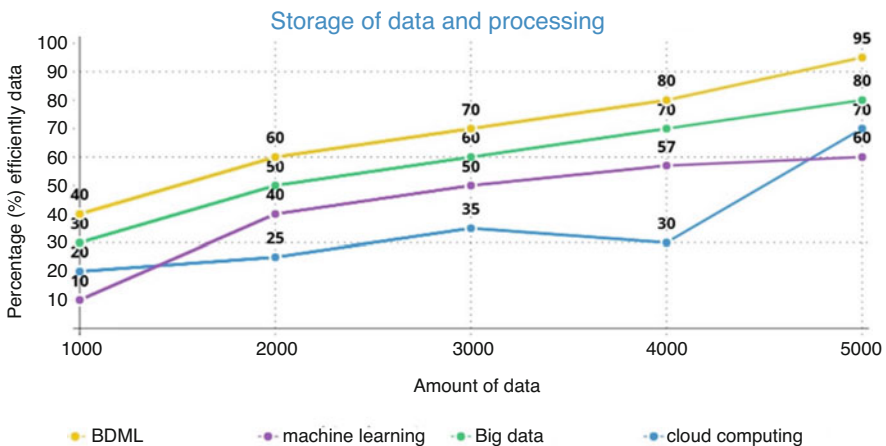


Fig. 2 Data storage and processing

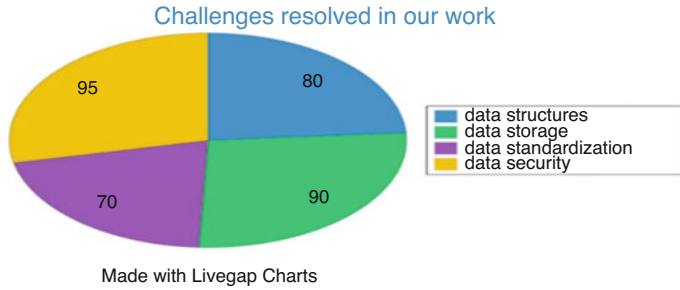


Fig. 3 Challenges resolved in our research

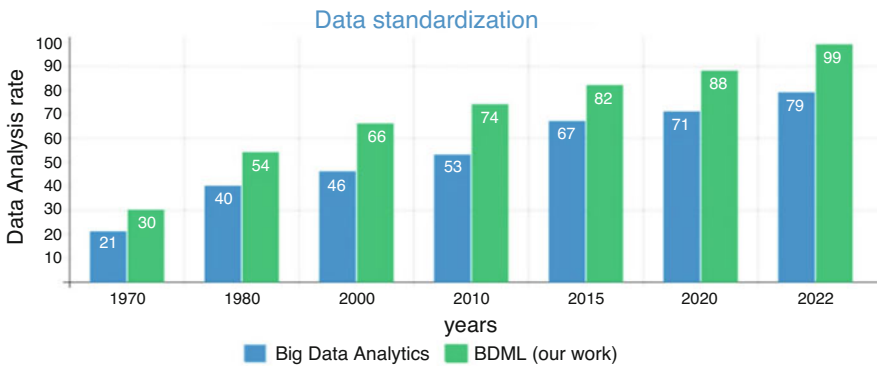


Fig. 4 Data standardization in BDML

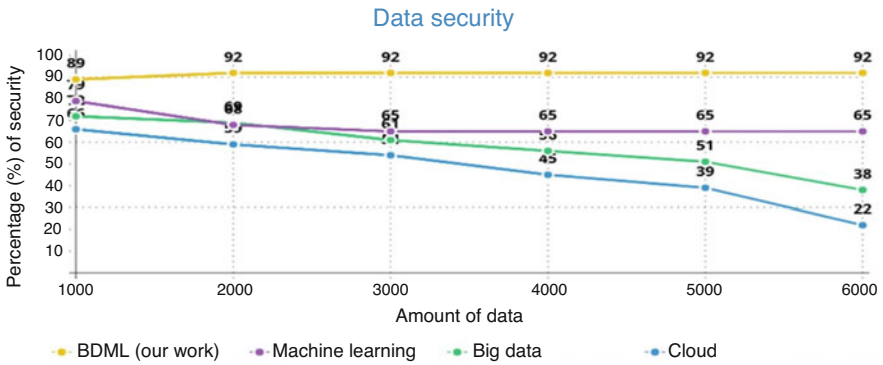


Fig. 5 Data security in BDML techniques

In Fig. 4, we represent the security of the data in our proposed work. The comparison of four different techniques helps us attain the high data security (Fig. 5).

## 6 Conclusion

This technique, big data with machine learning (BDML), is used to provide good efficient outcomes and attain our research goal to resolve the analysis of challenges in the literature review. The challenges in the analysis are data structure, data standardization, data security, and data storage and processing are recovered which are proved in data visualization techniques shown in our research's result analysis. At the same time, another aim of our process is attained, such that the big data increases the size of data storages, and processing exponentially increases in terabytes to zettabytes. That reduced the performance efficiency of big data analytics, involving the machine learning classification algorithm to group the collected related items for their analysis and predicted the outcomes of the trained and tested data set. The result analysis shows that the main goals are attained efficiently with 95% accuracy and predicted data in our research paper. In future many real time application are developed in this proposed work. The big data analytics can process huge volume of data to combine this methodology in machine learning handle to reduce the data storage. They can be applied to construct several real-time application such as detection of spam mail and determine if a loan from the bank is approved.

## References





1. B.S. Kim, M. Aldwairi, K.I. Kim, An efficient real-time data dissemination multicast protocol for big data in wireless sensor networks. *J. Grid Comput.* **17**, 341–355 (2019)
2. C. Costantini, E. Nunzi, A. Spolzino, M. Palmieri, G. Renga, T. Zelante, L. Englmaier, et al., Pharyngeal microbial signatures are predictive of the risk of fungal pneumonia in hematologic patients. *ASM J. Infect. Immun.* **89**(8), e0010521 (2021)
3. D. Ndirangu, W. Mwangi, L. Nderu, An ensemble model for multiclass classification and outlier detection method in data mining. *J. Inf. Eng. Appl.* **9**(2), 38–42 (2019)
4. N. Deepa, Q.-V. Pham, D.C. Nguyen, B. Sweta Bhattacharya, T.R. Prabadevi, P.K. Gadekallu, R. Maddikunta, F. Fang, P.N. Pathirana, A survey on blockchain for big data: Approaches, opportunities, and future directions. *Futur. Gener. Comput. Syst.* **131**, 209–226 (2022)
5. E. Ahmed, I. Yaqoob, I.A.T. Hashem, I. Khan, A.I.A. Ahmed, M. Imran, A.V. Vasilakos, The role of big data analytics in Internet of Things. *Comput. Netw.* **129**(2), 459–471 (2017)
6. L. Pani, S. Karmakar, C. Misra, S.R. Dash, Multilevel classification framework of fMRI data: A big data approach, in *Big Data Analytics for Intelligent Healthcare Management*, (Academic Press, London, 2019), pp. 151–174
7. M. Panda, S.M. Ali, S.K. Panda, Big data in health care: A mobile based solution, in *International Conference on Big Data Analytics and Computational Intelligence (ICBDAC)*, (IEEE, 2017), pp. 149–152
8. M. Marjani et al., Big IoT data analytics: Architecture, opportunities, and open research challenges. *IEEE Access* **5**, 5247–5261 (2017)
9. M. Younus, M.T.A. Munna, M.M. Alam, S.M. Allayear, S.J.F. Ara, Prediction model for prevalence of type-2 diabetes mellitus complications using machine learning approach, in *Data Management and Analysis Studies in Big Data*, vol. 65, (Springer International Publishing, Cham, 2020)



10. N. Bhattacharya, S. Mondal, S. Khatua, A mapreduce based association rule mining using Hadoop cluster—An application of disease analysis, in *Innovations in Computer Science and Engineering*, (Springer, Singapore, 2021), pp. 533–541
11. N. Sohail, M. Ren Jiadong, M.I. Uba, A. Khan, Classification and cost benefit analysis of diabetes mellitus dominance. *Int. J. Comput. Sci. Netw. Secur.* **18**, 29–35 (2018)
12. P. Sampath, S. Tamilselvi, N.M. Saravana Kumar, S. Lavanya, T. Eswari, Diabetic data analysis in healthcare using hadoop architecture over big data. *Int. J. Biomed. Eng. Technol.* **23**, 137–147 (2017)
13. P. Suresh Kumar, S. Pranavi, Performance analysis of machine learning algorithms on diabetes dataset using big data analytics, in *International Conference on Infocom Technologies and Unmanned Systems*, (IEEE, 2017), pp. 508–513
14. P. Chen, C. Pan, Diabetes classification model based on boosting algorithms. *BMC Bioinform.* **19**, 109 (2018)
15. S. Kumar, M. Singh, Diabetes data analysis using mapreduce with Hadoop, in *Engineering Vibration, Communication and Information Processing*, (Springer, Singapore, 2019), pp. 161–176
16. Y. Nait Maleka, A. Kharbouch, H. El Khoukhi, M. Bakhouya, V. De Floriod, D. El Ouadghiri, S. Latre, C. Blondia, On the use of IoT and big data technologies for real-time monitoring and data processing. The 7th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare (ICTH 2017). *Procedia Comput. Sci.* **113**, 422 (2017)
17. Y. Wang, L.A. Kung, T.A. Byrd, Big data analytics: Understanding its capabilities and potential benefits for healthcare organizations. *Technol. Forecast. Soc. Change* **126**, 3–13 (2018)
18. Z. Alansari, N.B. Anuar, A. Kamsin, S. Soomro, M.R. Belgaum, M.H. Miraz, J. Alshaer, *Challenges of Internet of Things and Big Data Integration*, vol 200 (Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, 2018), pp. 47–55

# Tracer for Estimation of the Data Changes Delivered and Permalinks



N. H. Prasad , S. Kavitha , Laxmi Narayana , and G. R. Sanjay 

## 1 Introduction

Source Code Management (SCM) plays an important role in organizing, managing, and controlling the changes delivered by the developers to the source code, documents, and other entities. In LGSI, there is no system that can track the changes delivered by the various stakeholders. The Change Tracker system aims to allow the developers to create a list of interesting files and send notifications to the intended stakeholders if any changes are committed and also the committed changes need to be reviewed by the reviewer. The proposed system is intended for use within LGSI to keep track of changes delivered by the LGSI developers. In addition, it is helpful in keeping stakeholders in communication by sending notifications when any new change is committed. This scope of the work can also be extended to other domains, like the TV team managed by the service provider. Users need not have to worry about the services or management. The best example is the web-based email service. The change Tracker system has been developed with Agile methodology. Agile methodology is popular for its nature of incorporating change in the industry with Agile frameworks such as scrum, Kanban, and extreme programming. The main components of the Scrum framework are organizing small teams, daily scrum meetings to know the status, making sprint planning and review, collaboration with other teams, and meetings with stakeholders. At first, the initial requirement of the Change Tracker system is to track the changes made to code by each developer in later stages based on discussions happened with the product manager and the internal stakeholders; the other requirements are

---

N. H. Prasad · S. Kavitha (✉) · L. Narayana · G. R. Sanjay  
Department of Masters of Computer Applications, Nitte Meenakshi Institute of Technology,  
Bangalore, India  
e-mail: [kavitha.s@nmit.ac.in](mailto:kavitha.s@nmit.ac.in); [narayan@nmit.ac.in](mailto:narayan@nmit.ac.in)

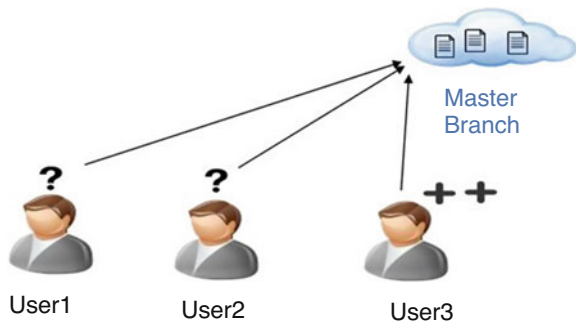
captured such as displaying analytics, exporting permalinks details to excel, and other functionalities. The feedback from various stakeholders play important role in developing a Change Tracker system. The proposed methodology Change Tracker is used in an IT security and compliance management platform. The Change Tracker benefits enterprises track and authenticate changes to the performance of devices, files, and configurations. The important features in Change Tracker are agentless integrity monitoring and intrusion detection.

## 2 Literature Survey

In SDLC, the same file is modified by many developers from different teams and geographical locations [1]. The latest version of the file may undergo several changes by the developers (Fig. 1). It is very difficult to identify who made the changes to a specific file and also difficult to find or recover accidentally lost files. This problem will directly affect project deliverables and may cause delays in identifying the problem and resolving issues [2]. In addition, the current system is not notifying the developers [3]. The advantages of the protractors are diminished when whiskers make contact with objects, according to the author [4], which causes whiskers to have a tendency to only barely touch the environment. There has been research on how this mechanism affects sensory input, but less is known about how sensory input alters the motor pattern [4].

Currently, there is no system to notify other developers who previously modified the same file, and there is no system to track developer’s action when exaggerated past designations file modified by others. Introducing changes accidentally may cause project risk.

Fig. 1 Existing system



### 3 Proposed System

The proposed system is capable of tracking changes delivered by the LGSI developers and enables communication between respective stakeholders. The proposed system is capable of tracking changes delivered by the LGSI developers and enables the communication between respective stakeholders. The Change Tracker tracks the files delivered by all users. It keeps track of the users interested in a file. Once a particular file is modified, Change Tracker notifies all the developers interested in that file. PL can track LGSI applicability on developer interested file. He proposed system facilitate developers to create list of interested Files. The Change Tracker tracks the files delivered by all users. It keeps a track of the users interested in a file. Once a particular file is modified, Change Tracker notifies all the developers interested in that file. The study is conducted before developing the Change Tracker application and analyzed the facts that will affect the project completion. In this study, we identified the technology stack essential for developing this Change Tracker system, such as HTML, CSS, Bootstrap, and JavaScript for the front end and Java, JSP, MySQL, and Java Servlets for the back end and also identified the organization has a strong technical team with essential skills and has the necessary hardware requirements. The development of this application is possible within the threshold budget for the entire project; the costs will include hardware and software costs, development costs, and operational costs. The Change Tracker application can be developed without any extra cost to the organization as this project can be carried out with the existing infrastructure of the organization [5]. This application is beneficial for the organization as it can track changes made to the project and avoids any accidental changes and ensures the software quality which will be beneficial for the organization in terms of finance. The proposed system facilitates developers to create a list of interested files [6].

### 4 Methodology

In the recent years, study has been conducted to map the behavior of components of complex systems and to clarify material flow and mixing properties. Although research groups have established systems to track rodents through video, with the exception of [7], prior approaches all need a secured computer for computation. In software engineering, software engineering researchers have provided many software development methodologies that fit or are adapted by various industries' Software solutions. During the initial stages of the project Change Tracker, there were various methodologies planned, and the Agile methodology was best suited for this project and worked accordingly. In recent years, the organization faced more difficulties as there is no system or software application for tracking permalinks and all their information. Therefore, the organization planned such an application that adapted one of the software development methods, agile development (Fig. 2). The

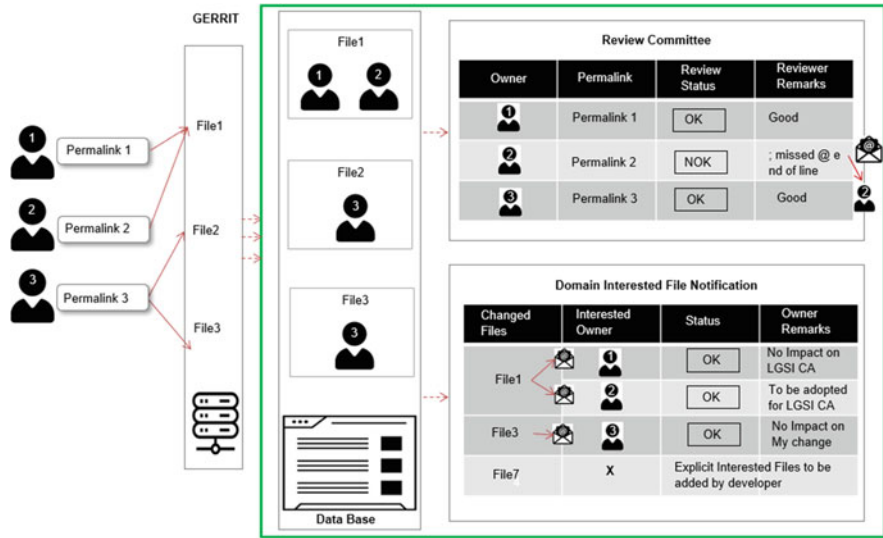


Fig. 2 Architecture of Change Tracker

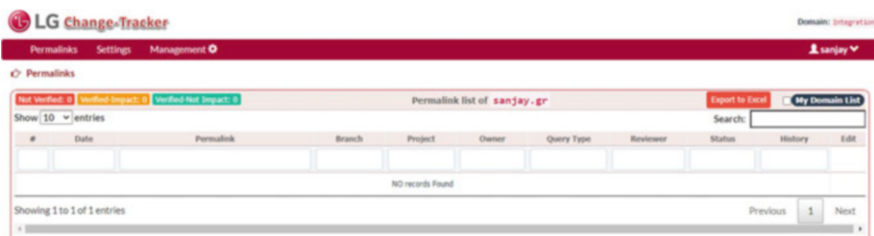


Fig. 3 User Permalink

agile development methodology is the simplest form. As a result of this concern, the advantages of agile software development [8] are that the organization is capable of reducing the overall risk which is associated with the Change Tracker tool.

The user can view records consisting of permalinks (Fig. 3) details created by all the developers. The internal stakeholders of the Change Tracker system can export the record of permalink details, including information such as date, permalink, branch, project, owner, query type, reviewer, status, and history.

In the settings, users can view records consisting of details created by the developers. They include information such as entities branch (Fig. 4), project, file, added type, status, copy query, and add query. Type is maintained to be application and state should be active [9].

The Change Tracker system must enable communication between the admin, reviewer, and user through email notifications (Fig. 5).

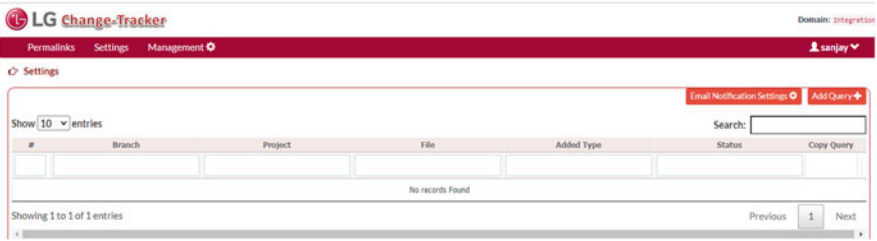


Fig. 4 Record view

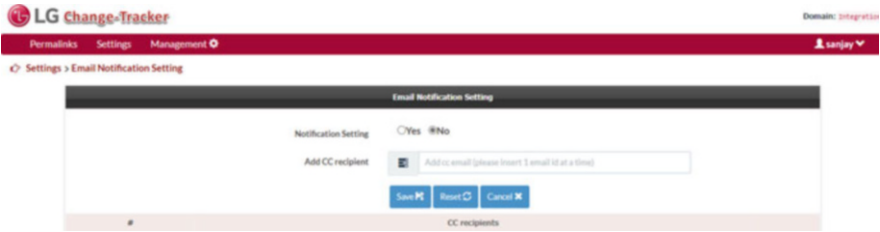


Fig. 5 Email notification setting

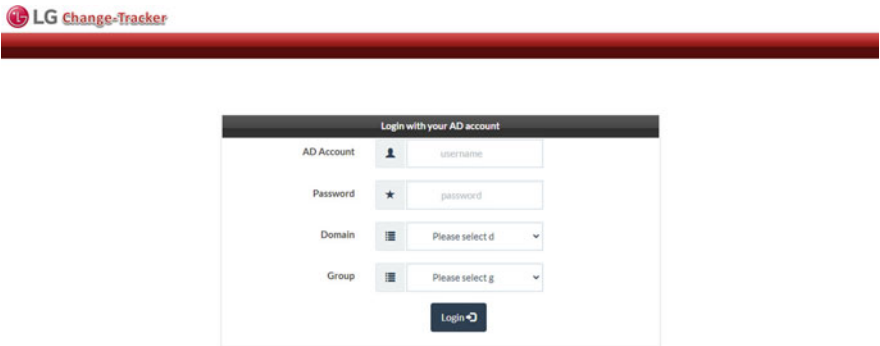


Fig. 6 Login to the module

## 5 Results and Discussions

After deploying the site on the elastic instance, we obtain the URL and visit the website that we deployed [10]. We then enter our credentials (Fig. 6) to log in, as the users are authenticated functions.

Here, the admin (Fig. 7) can post the permalink for other users to access and read them. This is how we can see the permalinks.

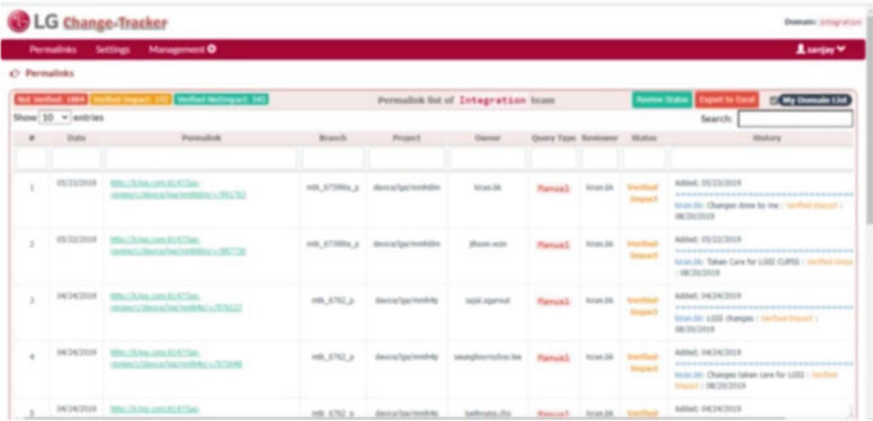


Fig. 7 Dashboard of permalink

## 6 Conclusion

In this paper, we have proposed the Change Tracker system to address one of the major problems faced by the LGSI employees in identifying who modified the code, removed files during software development. The Change Tracker system allows every individual developer to keep track of changes delivered to his/her interested files and creates permalinks delivered by LGSI developers then the permalinks will be reviewed by the review task team. In addition, Change Tracker notifies the respective stakeholders when the changes are made. Thus, this application helps in taking care of any risk associated when a new change is introduced or modified. In the future, the organization plans to enhance the proposed system by incorporating features like all DevOps activities such as tracking android security patches, domain patches also the Change Tracker system into a single application as its future goal.

## References

1. J.M. Vara Mesa, ATL/AMW use case modeling Web applications: Detailed description and user guide. *International Journal of Software Engineering and Its Applications*, 5(2), 2011 (2009)
2. S. Al-Fedaghi, Scrutinizing UML activity diagrams, in *17th International Conference on Information Systems Development*, Paphos, Cyprus, 25–27 Aug 2008
3. W. Huang, R. Li, C. Maple, H. Yang, D. Foskett, V. Cleaver, Web application development lifecycle for small medium-sized enterprises (SMEs), in *Proc. of the Eighth International Conference on Quality Software*, (IEEE, 2008), pp. 247–252
4. J. Voigts et al., Tactile object localization by anticipatory whisker motion. *J. Neurophysiol.* 113(2), 620–632 (2015)

5. J. McCurley, D. Zubrow, C. Dekkers, *Measures and Measurement for Secure Software Development* (Carnegie Mellon University Build Security In, 2008)
6. H. Vikas, Overview of lift web framework, in *Presented at the 4th IndicThreads.com Conference on Java*, (Pune, 2009)
7. M.A. Nashaat, H. Oraby, L.B. Peña, S. Dominiak, M.E. Larkum, R.N.S. Sachdev, Pixying behavior: A versatile real-time and post hoc automated optical tracking method for freely moving and head fixed animals. *eNeuro* **4** (2017). <https://doi.org/10.1523/ENEURO.0245-16.2017>
8. F. Baharom, A. Deraman, A. Hamdan, A survey on the current practices of software development process in Malaysia. *J. Inf. Commun. Technol.* **4**, 57–76 (2006)
9. M.P. Papazoglou, D. Georgakopoulos, Serviced-oriented computing. *Commun. ACM* **46**(10), 25–28 (2003)
10. B. Benatallah, *Web Services: Life Cycle Intelligence, Power Point Presentation, School of Computer Science and Engineering* (The University of New South Wales, 2006)



**Part II**  
**Bigdata and Privacy Preserving Services**

# Design and Development of a Smart Home Management System Based on MQTT Incorporated in Mesh Network



Andrea Antony, Nishanth Benny, Gokul G. Krishnan, Mintu Mary Saju, P. Arun, and Shilpa Lizbeth George

## 1 Introduction

The world is getting more efficient day by day, but it is on the verge of a major climate change. Daily home activities consume a significant amount of resources, and their impact is far greater. Users are always looking for a device that is interoperable and that makes their work easier. Smart home management solutions can cater these requirements. A smart home is a convenient house setting in which appliances and devices may be managed remotely using a smartphone or other networked devices. A smart home's devices allow the user to handle features such as home security, temperature, lighting, and other different features remotely. The smart home concept has been broadened by embedding sensors in quotidian devices and by enabling interoperability with mobile devices. This is due to the progression in home networking and the advent of high-speed Internet technologies [1]. People opt for more smart and green homes; hence, developing a smart home management system that allows controlling almost every aspect of your home is very essential. In current market trends, individual home management units are available. This mostly accounts for door security cameras and electrification controls using Internet of Things (IoT). All smart home management systems require the Internet to function, and they fail in case of disruption. Thus a niche for an all integrated, "internet-free" home management system arises. The future potential of home automation systems involves making homes even smarter. Smart homes will help save money on energy. Smart home systems appear to be gaining more popularity, implying that the way we live our daily lives will change substantially in future [2]. Many factors drive

---

A. Antony (✉) · N. Benny · G. G. Krishnan · M. M. Saju · P. Arun · S. L. George  
Department of Electronics and Communications Engineering, St Joseph's College of Engineering and Technology, Palai, Kerala, India  
e-mail: [arun@sjcetpalai.ac.in](mailto:arun@sjcetpalai.ac.in)

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023  
A. Haldorai et al. (eds.), *5th EAI International Conference on Big Data Innovation for Sustainable Cognitive Computing*, EAI/Springer Innovations in Communication and Computing, [https://doi.org/10.1007/978-3-031-28324-6\\_6](https://doi.org/10.1007/978-3-031-28324-6_6)

the development of smart home systems, but the most important are convenience, security, energy management, connectivity, and luxury. Even though most smart home devices can be controlled via an app and therefore do not require a control system or hub, smart home management systems allow the user to access multiple devices from a single interface and automate tasks and procedures [2]. The proposed system mainly focuses on developing a sustainable green home with reduced water wastage and improved security in all aspects. An overview of system architecture, mesh network, layered architecture, and MQTT communication system is given in Sect. 2, followed by the implementation of the initial prototype in Sect. 3. The final results and discussions are presented in Sect. 4.

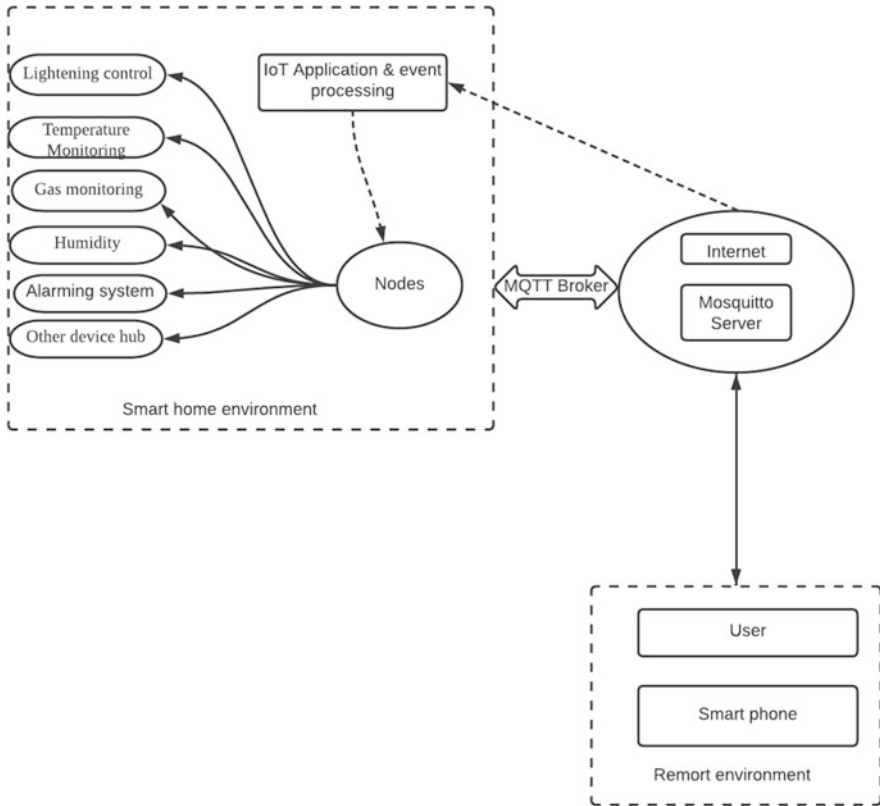
## 2 Overview

### 2.1 System Architecture

The proposed system is an IoT-based mesh network home management system that consists of a garden management system, water tank management system, gas leakage detection system, and a home security system. The system consists of a smart home environment and a remote environment, both of which are connected to a Mosquitto server via a MQTT broker as portrayed in Fig. 1.

The nodes are interconnected as a mesh network. Mesh networking is a typical multi-hop topology in which all wireless sensor nodes communicate with each other to hop data to and from the base station (Fig. 2). Each system has its own set of sensors and reads corresponding data. The data is processed by a single board computer. A corresponding output sequence will be initiated. The master node is a single board computer in which the Raspberry Pi 4 model B is used as the master, and it also consists of the display, speaker, and mic. It can be interfaced with the in-built voice recognition system “Jarvis” or interfaced with Node-Red. The slave node uses the ESP32, as a complete standalone system or as a slave device to a host micro-controller unit (MCU). As the system consists of several subsystems such as a water tank management system, gas leakage management system, etc., each system requires separate nodes to operate. Hence, separate ESP32 is used for each node to work. The messages are received from the local server with the help of the MQTT protocol. When a message is received to take sensor data, the sensor will first publish the data. The subscriber client will listen for incoming messages from the subscribed topic and react to the published action to that topic like “on” or “off.” Clients can subscribe to one topic and publish to another as well. MQTT protocol is used for communication, and ESPNOW protocol is used for the mesh topology.

The home management system inputs temperature and humidity data using a DHT22 sensor. The data is sent to the Mosquitto server. Using electronic switches and relays, coolers or air conditioning is adjusted to the temperature setting required by user. Electrical switches and appliances can be accessed remotely by user and



**Fig. 1** Block diagram

are switched on/off by relays. A display is incorporated to provide information to user. Soil moisture sensor, FC-28, detects water requirements in the plot. It triggers pump to turn on through a relay and other irrigation facilities to maintain ideal water requirements. This is the working of the garden management system. Gas leakages and accidents are a grave but common incident in households. With quick leakage detection, a heavy price can be avoided. The gas leakage detection system comprises a MQ6 sensor that senses butane. Upon sensing, a servo motor is used to close the gas inlet valve. Additionally, a buzzer is activated to alert residents. Theft control and safety is a major requirement in any household. In the home and security management system, when “Night mode” is turned on, any human movement, if detected outside the residence, will be alerted to users. A D203s PIR motion detector sensor detects any abnormal activity, sends this data to the server, and consecutively an alarm rings. The master switch is automatically turned on. Water tank management is vital to reduce water wastage due to overflow. An ultrasonic sensor helps measure water levels, and a relay will automatically activate

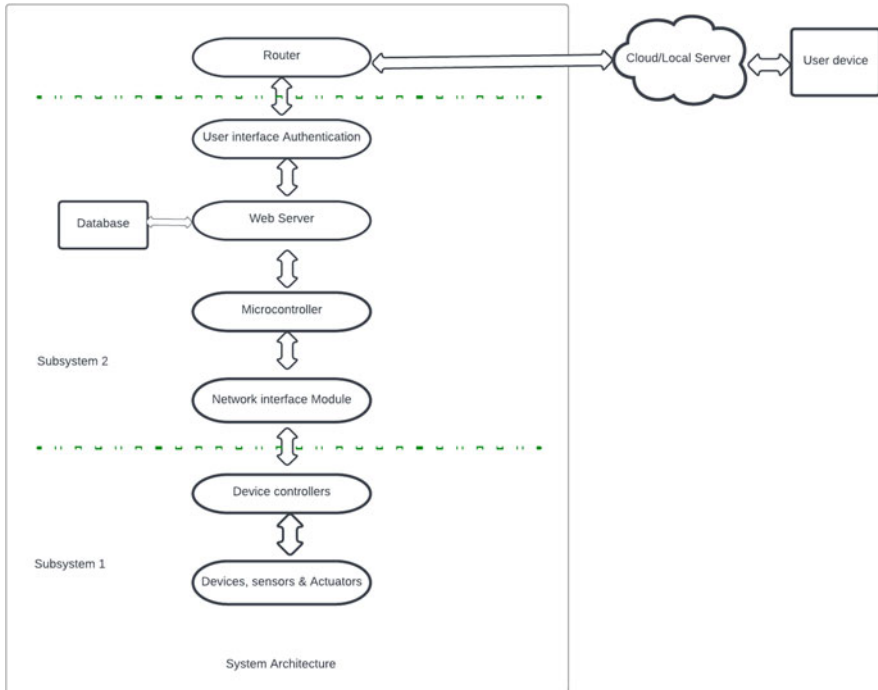


Fig. 2 System architecture

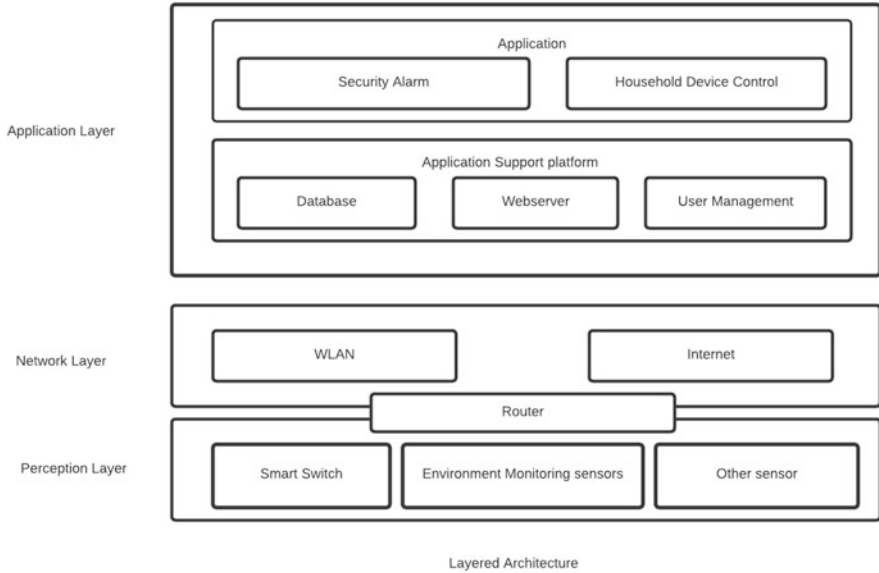
pump. Additionally, a TDS sensor to measure the water quality level and a Ph sensor to measure water Ph level are provided.

## 2.2 Mesh Networks

A wireless mesh network is a communication network made up of radio nodes that communicate with one another in a mesh topology [3]. Each node may connect with other nodes (within range); they essentially constitute an intelligent grid of access points (Aps) that can talk with one another to intelligently route traffic through the network. Mesh networks have a number of advantages as well as some drawbacks [4].

## 2.3 Layered Architecture

Figure 3 shows the layered architecture of the system. It consists of three main layers: application layer, network layer, and perception layer. The application layer processes the data intelligently, and that processed data is used by us. It includes



**Fig. 3** Layered architecture

the household device control and the alarming system based on the information received. Network layer focused on reliable transmission in which it transfers data through Internet and mobile telecommunication networks. The perception layer consists of various collecting and controlling modules that perceive and gather information from the sensors.

### 2.4 MQTT Communication System

MQTT is a low-bandwidth, lightweight publish/subscribe messaging protocol designed for M2M (machine-to-machine) telemetry [5]. A basic MQTT communication system consists of three basic components:

**Publisher:** Publishers generate and submit data to the MQTT broker. The publisher in the case of a smart home may be a weather station that sends temperature and humidity data to the broker every 5 min.

**Broker:** A broker, similar to a server, collects messages from all publishers, saves the data, and distributes it to the appropriate subscribers.

**Subscriber:** A subscriber is a component that receives a specific type of message from publishers. In the smart home use case, a subscriber could be a monitor or panel that displays the various temperatures and/or humidity levels within and outside the home.

The MQTT broker's Internet protocol (IP) address is the same as the Raspberry Pi's in the network. MQTT username and password are specified in the MQTT broker's configuration file (`mosquitto.conf`) [6]. In both the publisher and subscriber scripts, this setting must be identical. A MQTT client ID is only given to the publisher. Thus the publisher is identifiable, and it stops the broker from accepting data from unknown publishers. The publisher requires a service set identifier (SSID) and password. The micro-controller cannot send data over the local network without it.

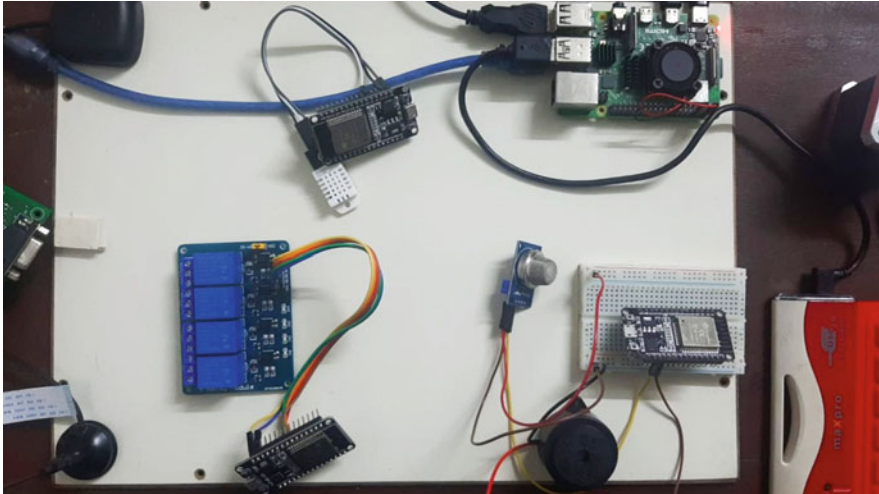
### **3 Implementation**

#### **3.1 Technologies and Platforms used**

- The system uses MQTT protocol by which master and slave node communication will take place.
- WI-FI modules are used for the prototyping. ESP32 WIFI module is used as the slave node controller.
- Speech recognition technologies have been used for the communication of the system with the user.
- For the master node prototyping, Raspberry Pi 4 Model B is used. It takes all the sensor data and necessary actuation takes place.
- Mosquitto server is used as a platform that is used for enabling the MQTT protocol [7].
- Node-RED—An open software of IBM is used for the monitoring of the system.
- Operating system is Linux.
- Arduino IDE platform is used for programming nodes.

#### **3.2 Prototype Design**

An initial prototype (Fig. 4) was set up to demonstrate the working of the proposed system. To imitate the main node, a Raspberry Pi 4 Model B device was used, while to imitate the subnodes, NodeMCU ESP32 was used. First, a mesh network was formed using the above units. Then, another mesh network was formed using the three ESP units. All individual management systems are interconnected using the MQTT protocol in a Mosquitto server. Sensors publish the data collected to the server that is then subscribed by various ESP32s, and the corresponding set of outputs to be taken is initiated. For example, the sensor in gas leakage detection system detects any traces of liquid petroleum gas or butane gas in the atmosphere. If the value exceeds a certain threshold, the buzzer is triggered and a servo is rotated to close the regulator valve. This was realized by using the MQTT protocol. For this implementation, a local Mosquitto MQTT Broker was used. Thus the entire IoT system can thus be controlled, even in the absence of Internet connectivity.



**Fig. 4** Initial prototype

## 4 Results and Discussions

The smart home management system was designed and fabricated. A basic prototype was made with all nodes and sensors. The programming was done in Arduino IDE. Graphs were plotted using Arduino IDE serial plotter. Figure 5 shows a graph plotted with the data received by the MQ6 sensor for gas leakage detection system. In case of butane gas detection, when the threshold level exceeds, the actuator turns off the gas valve that cuts off the gas supply. Figure 6 shows the various inputs received and outputs generated by the home management system, while measuring the humidity and temperature parameters. As an add-on feature, a voice recognition system was developed to enable users to give commands via speech. The voice recognition and control system was named “Jarvis.” It also provides information about the news, time, etc., apart from control of different nodes. The initial design for the formation of mesh networks was by interconnecting different nodes via a MQTT server. A local Raspberry Pi network was formed. This turned out to increase system complexity. On further research, an improved methodology was developed (Fig. 7). In the improved design, ESP module and sensors were implemented as separate modules, instead of being in the same module. It decreases the complexity of the system and operates the system as planned.



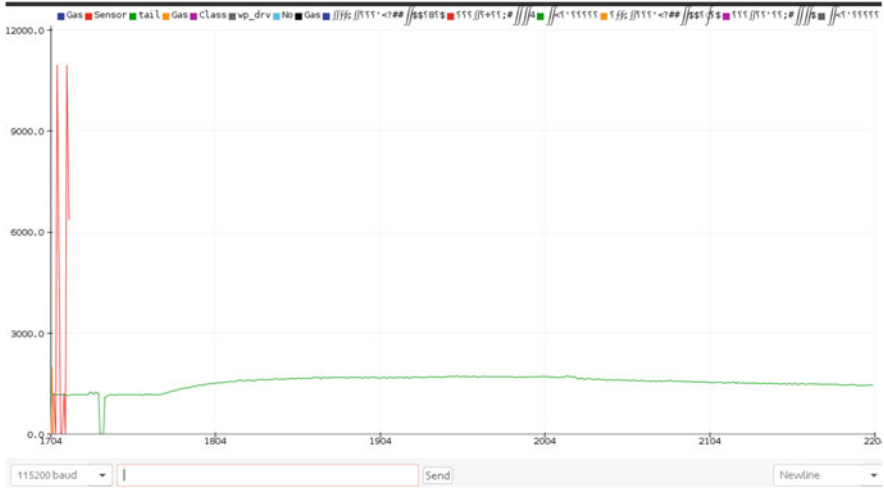


Fig. 5 Graph obtained

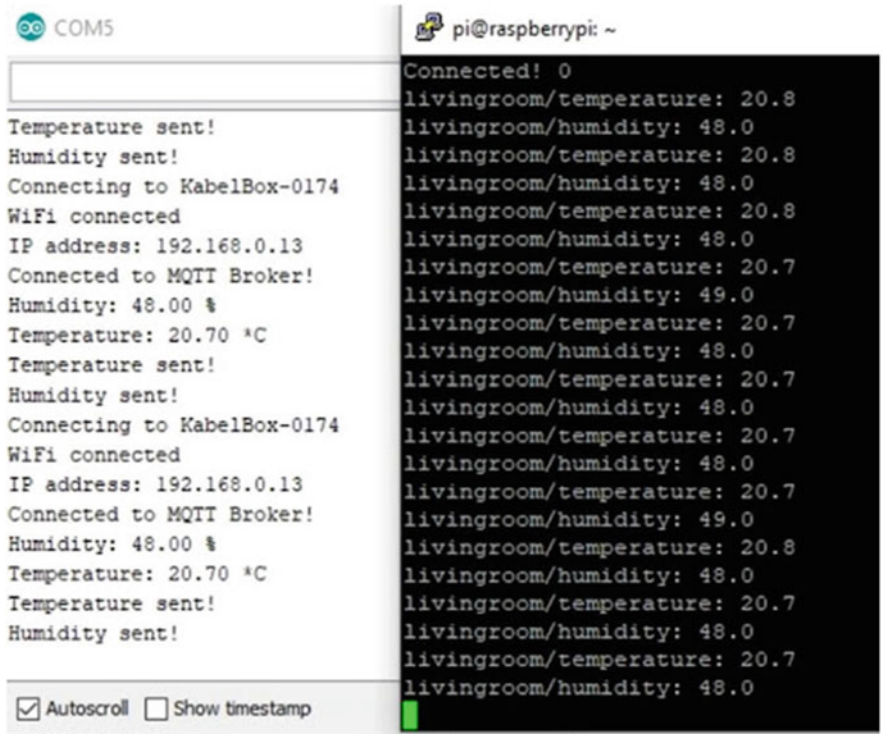
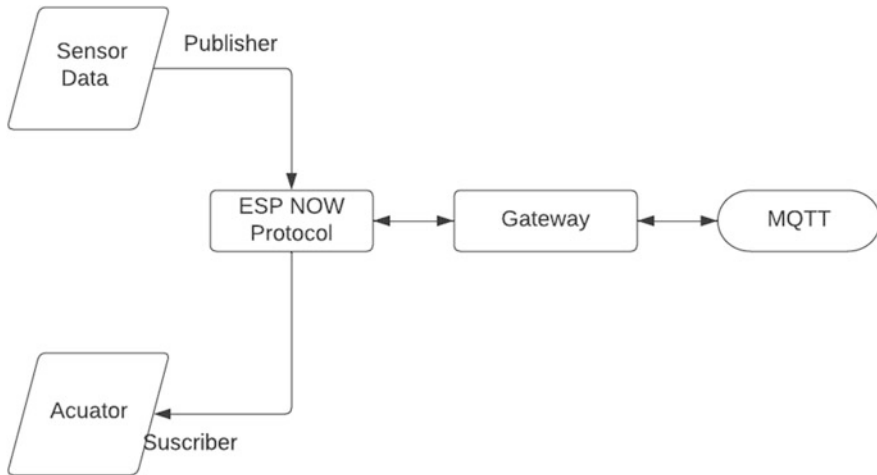


Fig. 6 Home management system output



**Fig. 7** Improved system design

## 5 Conclusion

This chapter proposes the working of a mesh network smart home management system. The design, development, and implementation of smart home management system incorporating mesh network and embedded system was conducted. The proposed system will help users to be efficient in energy and water usage along with providing a safe and secure home even in the absence of Internet connectivity. The system is capable of controlling and monitoring various aspects including:

- Gas leakage management system
- Garden management system
- Home switches control system
- Advanced security system

The existing smart home management devices are available as individual modules, which work independently. Hence, there is no existing system where all the functionalities are integrated together. An initial functioning prototype was executed. The accuracy of the system has been tested for different conditions. The system mainly consists of a master node and slave nodes that are interconnected as a mesh network. These nodes communicate with each other to hop data to and from the base station. Shortcomings identified with initial design were identified and replaced with updated system design. Thus an improved methodology for the effective operation of the system has been proposed by the implementation of ESP module and sensors as separate modules and interconnecting it with a gateway. The market gap of integrated Internet-free smart home management system has been addressed by the proposed system in a cost-effective manner.

## References

1. H. Wei-Dong, Z. Bo-Xuan, Smart home wireless system using ZigBee and IEEE802.15.4, in *2016 Sixth International Conference on Instrumentation and Measurement, Computer, Communication and Control (IMCCC)*, (2016), pp. 858–863. <https://doi.org/10.1109/IMCCC.2016.168>
2. G. Song, Z. Wei, W. Zhang, A. Song, Design of a networked monitoring system for home automation, in *IEEE Transactions on Consumer Electronics*, vol. 53, no. 3 (2007), pp. 933–937. <https://doi.org/10.1109/TCE.2007.4341568>
3. K.C. Karthika, Wireless mesh network: A survey, in *2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)* (IEEE, Piscataway, 2016)
4. R. Kashyap, M. Azman, J.G. Panicker, Ubiquitous mesh: A wireless mesh network for IoT systems in smart homes and smart cities, in *2019 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)* (2019), pp. 1–5. <https://doi.org/10.1109/ICECCT.2019.8869482>
5. G.R. Hiertz et al., Mesh technology enabling ubiquitous wireless networks: Invited paper, in *Proceedings of the 2nd Annual International Workshop on Wireless Internet. WICON '06* (ACM, Boston, 2006). ISBN: 1-59593-510-X
6. R.A. Light, Mosquitto: server and client implementation of the MQTT protocol. *J. Open Source Softw.* **2**(13), 265 (2017)
7. [Online]. Available: <https://mosquitto.org/>. Accessed 28 June 2022

# Novel Machine-Learning-Based Decision Support System for Fraud Prevention



Norman Berezcki, Vilmos Simon, and Bernat Wiandt

## 1 Introduction

Over the past years, there has been a dramatic increase in the amount of data generated by people using the Internet. However, the development of hardware and software was significant, and scientists could not process a big amount of data with a personal computer. Dealing with huge amounts of data or doing calculations on complex problems became slow, and using supercomputers is not accessible for a wide range of projects. The intense growth of the amount of data and the increasing complexity of new computer technology solutions, such as blockchain, and the benefits of services gained interest for cloud computing.

Cloud computing is a service-based solution that allows the user to use services, such as IaaS, PaaS, SaaS, etc., rented from a provider. A major advantage of using cloud technologies is the better performance, the use of customizable hardware that does not require maintenance, and the better support for cooperative workflows. Based on these advantages, cloud service technology has gained large momentum in corporate environments [1]. Cloud services play a key role for a wide range of scientists and industrial processes.

Cloud platforms provide efficient resource allocation and several useful functions, such as creating automated backups. Registering on several major platforms is open for everyone. Thus it is unavoidable that fraudulent registrations will happen. Users are labeled as fraud, if somehow cause harm to the company: not paying their bills or engaging in illegal activities such as storing/streaming child pornography or mining cryptocurrencies.

---

N. Berezcki (✉) · V. Simon · B. Wiandt

Department of Networked Systems and Services, Budapest University of Technology and Economics, Budapest, Hungary

e-mail: [norman.berezcki@edu.bme.hu](mailto:norman.berezcki@edu.bme.hu); [svilmos@hit.bme.hu](mailto:svilmos@hit.bme.hu); [bwiandt@hit.bme.hu](mailto:bwiandt@hit.bme.hu)

**Table 1** Last 3-year complaint loss comparison in millions of \$ [2]

| Crime type           | 2021 | 2020 | 2019 |
|----------------------|------|------|------|
| Credit card fraud    | 173  | 130  | 111  |
| Confidence fraud     | 956  | 600  | 475  |
| Identity theft       | 278  | 219  | 160  |
| Personal data breach | 517  | 194  | 120  |

Table 1 shows that there is a massive increase in the loss because of fraudulent activities on the Internet.

Preventing malicious activity is important to protect the users of online services and to preserve their reputation. The majority of fraud cases can be detected during the registration process based on the recurring patterns coming from the registration information. Data is collected for each and every registration to the cloud platform. The data is mostly collected by the providers, but there are several third-party services providing information about the users. The data collected is used to determine whether a registered user is likely fraudulent or not. A user is considered anomalous, or called anomaly, if it is potentially going to do fraudulent activities. Currently, this work is done manually by analysts; thus, it is non-deterministic, because decisions made by people can be easily biased based on the subjectivity of people, such as experience or the current emotional state.

The aim of this research is to improve the efficiency of the fraud detection process by providing a decision support system for the analysts and implementing an unsupervised anomaly detection algorithm to validate the labeling. Every user that causes damage to the company (such as going against the laws or harming its reputation) is called fraud/fraudulent activity or anomaly.

The paper introduces a novel machine-learning-based approach that can be integrated into the existing decision-making framework performed by analysts. This new process eventuates a more deterministic way of classifying users and higher accuracy. The new method is also capable of giving feedback for already labeled data set thus improving the analysts' decision methodology. Our newly developed system relies on both supervised and unsupervised machine learning algorithms to provide multiple approaches to the problem.

The structure of this chapter is the following: Sect. 2 provides a brief summarizing about existing anomaly detection methods and fraud prevention systems. Section 3 details how the newly developed method works and what advantages does it have instead of using just the analysts' decision-making. Section 4 presents the evaluation of the introduced process on a real industrial data set and examines how it performs. Finally, Sect. 5 summarizes our methodology and its impact and proposes future development possibilities.

## 2 Related Works

Fraudulent activities are probably as old as humanity since people started using computers and telecommunication technologies so started criminals and scammers. The first amendment to the federal computer fraud law was enacted in the early stages of telecommunication and networks in 1986 [3]. Research into fraud detection due to its importance has a long history. The first discussions and analysis of computer fraud emerged during the 1970s from L. I. Krauss [4]. Fraudulent activities have been the subject of many studies in various fields. Since then, there has been an increasing amount of literature on preventing fraudulent activities [5].

There are a plenty of definitions of fraud. The Association of Certified Fraud Examiners defines fraud as “the use of one’s occupation for personal enrichment through the deliberate misuse or misapplication of the employing organization’s resources or assets” [6]. There are two types of fraud against companies: external and internal fraud. Internal fraudulent activities against a company is committed by an employee, for example, a sabotage. External fraud can be the activity of a user that harms the law, for example, not paying for the service, or streaming forbidden contents, such as related to terrorism. This chapter focuses on external fraudulent activities. Surveys such as that published by Abdallah et al. [7] have shown that the definition of fraud highly depends on the domain of the field it is observed on.

An efficient fraud prevention process includes a precise fraud detection. Fraud prevention is a critical aspect of online services because it has a high impact on the service provider’s reputation. An online service can easily lose a large proportion of its customers if it has a reputation for being easily hackable [8].

A good approach to find fraud users is anomaly detection. Anomaly detection refers to the problem of finding patterns in data that do not conform to expected behavior. These nonconforming patterns are often referred to as anomalies or outliers [9]. The survey of Chandola et al. [9] provides a comprehensive review on anomaly detection methods and their usage. It discusses the variety of reasons how anomalies in data from several domains can be found [9]. Anomalous traffic patterns can indicate malicious attempts [9]. For example, anomalies in medical or healthcare data can mark abnormal patient conditions [9]. Anomalous user detection is often performed by graph-based fraud detection methodologies [10].

A large and growing body of literature has investigated user-profiling-based anomaly detection [11, 12]. The common purpose of this is to create user profiles and define distance thus creating clusters. The hypothesis what these algorithms are based on is that users in same clusters act same, so the goal is to identify fraud user clusters.

A common fraud activity is registering fictitious accounts. Marakhtanov et al. proposed a long short-term memory (LSTM) recurrent neural-network-based methodology that can detect fraud users with 0.99 recall [13]. Sharma also investigates an LSTM autoencoder-based user behavior anomaly detection system that performs 0.91 recall [14].

Other neural networks than LSTM can perform well at anomaly detection tasks. In their study, Ding et al. show how neural-network-based technologies can be applied for anomalous user detection [15].

Hodge and Austin [16] identify 3 fundamental approaches to the problem of anomaly detection:

1. Determine the outliers with no prior knowledge of the data. A statistical approach is flagging the most outlying data in a given set based on statistical operations. Another approach is to perform unsupervised clustering with machine learning algorithms. Common algorithms from this field are k-nearest neighbor, connectivity-based outlier factor (COF) [17], one-class support vector machine [18], and neural-networks-based solutions such as the self-organizing map (SOM) [19] and the adaptive resonance theory (ART) [20].
2. Model the normal and abnormal behaviors. These methodologies require a labeled training set. The used methodologies for this approach are the scoring functions, the linear classifiers, the classification trees, and the nearest-neighbor methods [21].
3. Model normal behavior only. It is referred to a semi-supervised recognition task. Technologies utilized for this approach are the SSMBBoost, the Boosting, the ASSEMBLE, and the SemiBoost [22].

A common approach of anomaly prevention in user space is to assign a profile to every user [23]. These profiles (often handled as matrix) contain information about predefined properties of a user. This profile can be compared with the target variable, and machine learning algorithms can detect correlation between the label and certain property values.

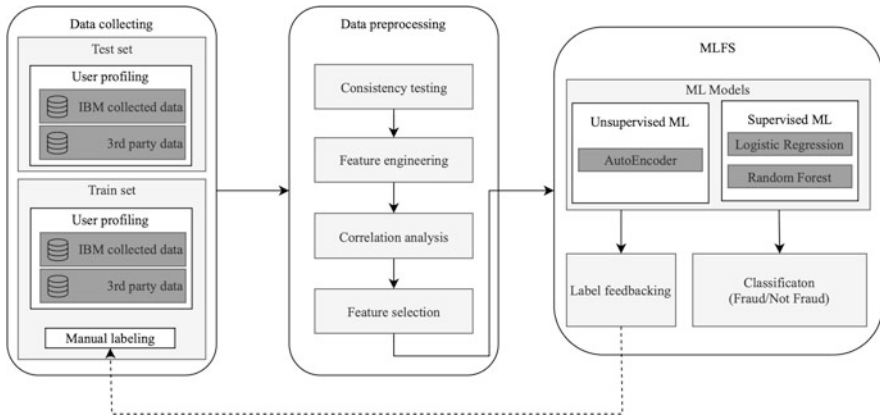
One of the earliest and most cited user-profile-based anomaly detection system has been presented by Lane et al. [24]. Their methodology presents a machine-learning-based approach to detect anomalous user activities, where the user profiles are collected from their UNIX typing sequences.

A popular machine-learning-based classifier is the random forest classifier. There are several papers about the successful use of the algorithm for anomaly detection [25–27].

Kater et al. present a study that uses state-of-the-art classification algorithms to filter out malicious users during registration [28]. They show that a supervised approach with the right features can provide a good basis for fraud prevention.

### 3 Machine-Learning-Based Fraud Prevention System

As Sect. 2 shows, there are a plenty of existing anomaly detection methods. A possible solution could be to implement a general solution, but the domain and the definition of anomaly differ highly in each case, thus more specific, domain-related models tend to have better performance, and moreover, they can be implemented to mimic the decision-making process of analysts more precisely. This section



**Fig. 1** Decision-making framework implementing MLFS

presents the machine-learning-based fraud prevention system (MLFS) developed by us, which uses the exact same features that are used by analysts for decision-making. This newly introduced system handles numerical and categorical features of a registering user as an input and returns a probability that shows how likely is it that the user is fraud. MLFS is integrated into the decision-making process. By making a decision, the analysts take the suggestion made by MLFS into consideration to decide whether a user is labeled as fraud or not. Use of MLFS leads to a more deterministic process, because it is based on exact mathematical models, while human decision-making can be highly influenced by the personal elements, such as the experience level or current emotional status.

Figure 1 shows how the decision-making framework implementing MLFS builds up. It consists of 3 main components: data collecting, data preprocessing, and MLFS. The data collection part collects the user profiles and assigns labels to the training data points. Data preprocessing block is responsible for testing the consistency of the train and test sets, generating new features, and performing correlation analysis. The goal of this block is to enable the process to rely on features that have high descriptive power. After the input is prepared, MLFS is ready to be utilized. Both supervised and unsupervised approaches are implemented and contribute to detect anomalies. Supervised algorithms are trained on the pre-labeled data by analysts. Basically, the supervised approach models and mimics the decision-making of analysts. This indicates a logical limit because if the classification algorithm reaches up to perfect accuracy, it is still only as accurate as the analysts' labeling. There might be misclassified cases in the training data set.

To improve the performance and eliminate this limitation of fraud prevention, an unsupervised approach is also implemented. It determines the outliers based on mathematical operations without labels. The final classification process of a new user is shown in Fig. 2. The pre-processed user profile is passed to the analyst and to the pre-trained ML models. The supervised and unsupervised approaches calculate



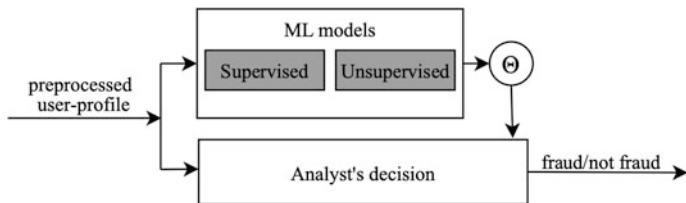


Fig. 2 Decision-making process with MLFS

the probability of fraud case, and then  $\Theta$  calculates the arithmetic mean of the two probabilities and passes it to the analyst. The analyst's decision is highly supported by the implemented machine learning algorithms.

For the supervised approach, the logistic regression (LR) and the random forest classifier (RF) are used. These algorithms have been selected based on several reasons. Both are lightweight algorithms that do not need big computational capacity. Both algorithms are easy to understand and do not operate black-box-like as most of the neural-network-based classifiers do. LR and RF can also prevent overfitting. These algorithms are commonly used and well-studied ML algorithms. These two algorithms are really accurate when performing one-class classification, LR performs better in overall accuracy, but the true positive rate and false positive rate are higher for RF with increased noise variables [29]. By using deep neural-network-based classification solutions, it could potentially improve the effectiveness of the classification; however, the number of available labeled data set is too small to train a supervised neural network model.

**LR** is one of the mostly used statistical models to calculate the probability of one event based on the linear combination of more independent variables. These variables, called predictors, are the normalized values in a user profile. The output of LR is a continuous probability variable bounded between 0 and 1 that shows that the probability of a newly registering user will be fraud. To prevent overfitting, a cross-validation set is used, that is, the 5% of the test set.

**RF** is a robust, very popular classification method. It initializes several independent decision trees and uses the most voted results by the decision trees. Decision trees are decision support systems that represent the final result of successive decision sequences. These decision can be based on probabilities, cost functions, etc. RF is a popular algorithm because it is easy to use, has very high accuracy, and provides solution for overfitting by using bagging. It uses just a subset of the data set during teaching and performs validation with the other data points. It makes unnecessary to perform cross-validation, since this operation corresponds exactly to that.

The unsupervised part of MLFS implements an **autoencoder neural network**.

As discussed earlier in this section, our goal is to implement an anomaly detection process that is independent from the labels made by analysts thus avoiding the logical limitation. Autoencoder is a generative unsupervised neural network that is commonly used for anomaly detection. It consists of two main parts: encoder

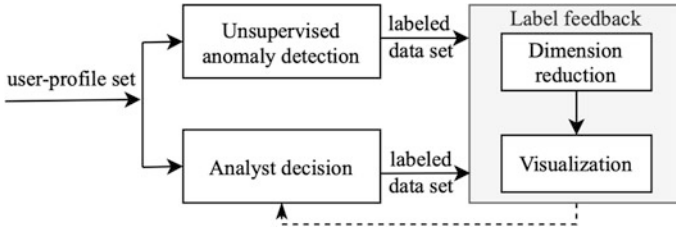


Fig. 3 Feedback to the analysts’ decision-making with MLFS

and decoder. The encoder takes a user profile  $X$  as an input and compresses the given data into lower-dimensional latent subspace  $z$ . After this, the decoder takes the compressed data  $z$  and the decoder reconstructs the original data  $X'$ . Then  $X$  and  $X'$  are compared, and the reconstruction error (mean square error) is calculated. It is assumed that the parameters of a fraudulent user are different from a normal user. The autoencoder learns how to compress and decompress data points that represent users with normal behavior. If an anomalous case occurs, the decompression part can only reconstruct it with high error. If the error crosses the threshold, the case is marked as anomalous.

The MLFS can also re-classify an existing labeled user profile set based on the unsupervised (autoencoder) approach shown in Fig. 3. The unsupervised anomaly detection is performed on the same database that is labeled by the analysts. The user profiles, labeled by both the unsupervised algorithm and analysts, are then transformed into a 3-dimensional latent space to be able to visualize it. In this latent space, the results of the two decision-making methods are compared. This comparison may show cases that are marked as anomalous by the unsupervised learning but not by analysts. These cases are then marked and sent back to the analysts for a deeper analysis. This process gives a feedback for the analysts’ decision-making. If the autoencoder implementation can correct the imperfection of the labeling process that can lead to a labeled training data set with less fault, thus the quality of the training data set improves. A better training set results in more accurate supervised classification. The cooperative use of both supervised and unsupervised approaches leads to a self-improving system.

To be able to visually compare the labels given by the analysts and by the autoencoder, dimension reduction is implemented to overcome the visualization difficulty of user space by its high dimensionality. Dimension reduction is a process that reduces data from a high-dimensional space to a lower-dimensional space with minimal loss of information using mathematical operations (e.g., different projections).

MLFS uses the **UMAP** (Uniform Manifold Approximation and Projection for Dimension Reduction) dimension reduction method. UMAP has several advantages over the popular dimension reduction method t-SNE, but the two most significant are the radical reduction of the running time of dimensional reduction on large sets and the better preservation of local data structures that is an important part

of visual cluster analysis for anomaly detection [30]. MLFS reduces the data into a 3-dimensional latent space and plots each compressed user profile as a data point, where the colors of the points indicate the label of the anomaly detection process. Then the labeling of the autoencoder and the analysts can be visually compared. A big difference between the two labeling indicates that the autoencoder-based method is not able to perform anomaly detection properly, because much of the analysts' labeling can be considered correct. If the comparison shows a slight difference, then the differing users are sent back to the analysts for more observation, because the ground of the difference can be that the analysts did not recognize the anomalous pattern properly.

In summary, MLFS can contribute to the decision-making by:

1. Proposing a classification methodology based on various machine learning approaches
2. Using a mathematical way to decide whether a new registering user is anomalous or not thus eliminating the human intuition and making the process more deterministic
3. Giving feedback for the labeled data set thus improving the labeling mechanism used by analysts

## 4 Experiments and Results

### 4.1 Data Collecting

MLFS uses existing data to determine whether a new user is potentially fraud user or not. It is collected by IBM and third-party services, which are to provide information about the context of a previous occurrence of a user, based on its username or e-mail address.

The data for this study was generated by IBM Budapest Lab for educational purposes. The used databases are synthetic and GDPR compliant, so it is not possible to determine how much of the world of the IBM Cloud they cover or when the data was recorded or how much it reflects the current set of users. This data set was labeled by analysts. The anomalously marked cases were pretty rare, and they accounted for only 20% of the data set making it the data set unbalanced by the two categories (fraud/not fraud). The data consists of 500.000 records with about 300 features. There is also an unlabeled data set available during the project that has not been labeled manually yet containing 2.000.000 records.

For features to be passed to a model, a unified structure is required. Because the data came from multiple sources, their occupancy rates were different, so features that were missing by more than 10% of the entities from the labeled data set were dropped. The categorical variables have been re-coded into a maximum of 4 categories, keeping the most significant 3 and one "other" category. Approximately, 30 properties remained.

## 4.2 Data Preprocessing

To use the labeled data set for teaching the model that will be evaluated on the unlabeled data set, it is important that the features in the two data sets are consistent with one another. Consistency means that the sets of values for the same features in the two databases are statistically identical. Using consistent training and validation data set is essential for good predictions. The machine learning algorithm learns the patterns from the training data set. In case when the validation data set is not similar, the characteristic of these patterns can differ that can intensely bring down the efficiency of the algorithm, because it is trained to recognize patterns in the training set. To determine whether the labeled and unlabeled data sets are similar, the statistical analysis of variables has been performed.

For continuous variables, the goal is to disprove the null hypothesis. The null hypothesis claims that there are no statistical relationship between two sets. The null hypothesis persists until proven otherwise [31]. The alternative hypothesis claims that the values of the two variables are statistically related, making the two data sets similar. To prove this, the *paired t-test* and the *Smirnov test* are used. The *t-test* examines whether the mean of two variables differs statistically [32]. The Smirnov test is a two-sample version of the Kolmogorov–Smirnov test, which is designed to show if two samples are identically distributed. The two samples are statistically similar if the *p*-value of the test is less than 0.05. Both of the statistical tests showed that the labeled and unlabeled data sets are consistent through continuous features.

To examine categorical variables, the distribution of the occurrence of unique values has been tested. The distribution of every categorical feature has been compared. The comparison showed no significant difference between the occurrence of each value in the two data sets thorough the same feature. However, when examining the categorical variables, it was found that the labeled set had a greater fill rate. This may be due to the fact that users in the labeled data set use a paid service, and these users are more likely to provide more data, against those users that will use just the free plan services.

MLFS only performs operations on numerical data, so it is important to convert the categorical text features into numerical. The *one-hot encoding* was used for the conversion. For a feature with *n* category, one-hot encoding generates an  $n - 1$  high vector. The trivial base vectors (unit vectors) and the null vector represent each category. This has the advantage over label encoding, which assigns an integer to each category, to define the same distances between categories.

From the existing properties, new properties are created that can increase the accuracy of the method. The location of the IP address was available from various sources, where the billing takes place, and whether the user is using a VPN or other servers that mask the IP. If this is not used, but the IP address and the billing address do not match, a Location Unsimilar bit is set to 1, indicating that the two locations are different. Otherwise, its value is 0. If there is missing data, its value is also set to 1 as a precaution.

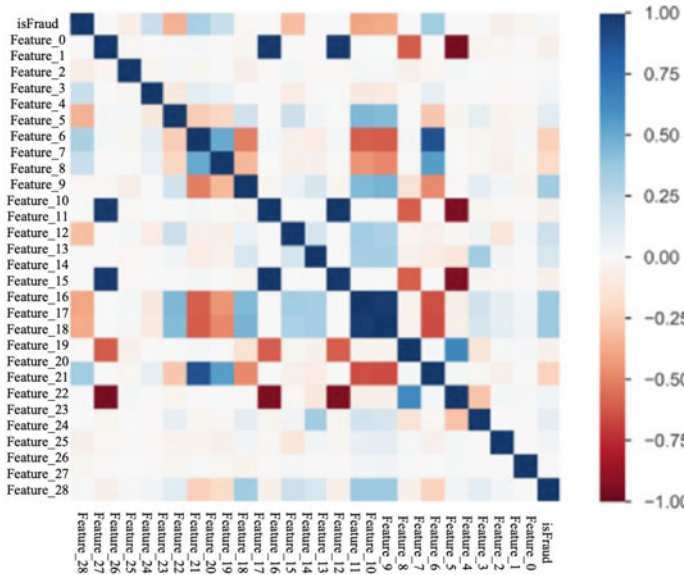


Fig. 4 Pearson’s correlation matrix

In order to select the features with high descriptive power for the model, two correlation coefficients are observed. **Pearson’s correlation coefficient:** It is a frequently used correlation metric that can detect the linear association between two variables. Its coefficient range is from -1 to 1. If the  $r$  coefficient is positive, it represents a positive relationship between the variables; otherwise, if negative, it represents a negative correlation.  **$\phi_k$  correlation coefficient:** The  $\phi_k$  correlation coefficient has several advantages. It can be used between categorical, continuous, and interval variables. Unlike Pearson’s  $r$ , it is also capable of detecting nonlinear relationships. If the two variables are Gauss-distributed, it leads the problem back to the Pearson  $r$  coefficient; otherwise, it calculates the  $\chi^2$  test.

The Pearson’s correlation coefficient and the  $\phi_k$  correlation coefficient showed identical results. Figure 4 shows the Pearson correlation matrix of the features and the top 10 features have been selected to the model.

### 4.3 Performance Evaluation of MLFS

To measure the accuracy of MLFS classification, accuracy, precision, recall, and F1-score metrics are used (TP: True Positive, TN: True Negative, FP: False Positive, FN: False Negative):

– Accuracy:

$$\frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

– Precision:

$$\frac{TP}{TP + FP} \tag{2}$$

– Recall:

$$\frac{TP}{TP + FN} \tag{3}$$

– F1-Score:

$$\frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{2 * TP}{2 * TP + FP + FN} \tag{4}$$

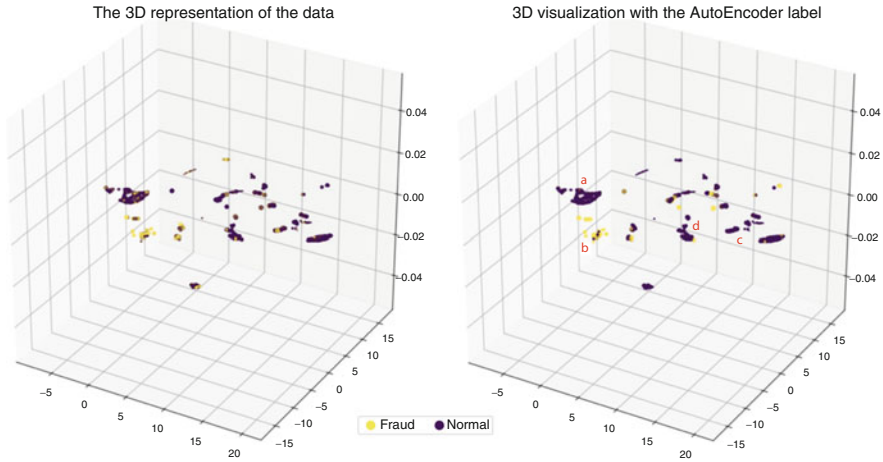
Precision shows how many of the anomalously classified users are truly anomalous. In contrast, recall shows how many anomalous users have been labeled as anomalous out of every anomalous user from the user space. If the algorithm retrieves 60 anomalies out of 100 anomalous cases and marks another 60 users as anomaly that are indeed not, that means 0.5 precision and 0.6 recall. But if the algorithm retrieves 20 anomalies out of 100 truly anomalous cases and retrieves 0 false positive cases, that means 1 precision but only 0.2 recall. It is important to know these metrics, so the MLFS can be optimized for the right usage. The goal of this system is to optimize it to find as many truly anomalous cases as possible instead of optimizing it to label as few normal cases as anomalous as possible.

Table 2 shows the performance measurements of MLFS. As Table 2 shows, RF reached better performances in every aspect than LR; however, the difference is not significant. The results are adequate, and the developed system can recommend labels with high accuracy.

Autoencoder was trained on the labeled data set without passing it the labels. The trained autoencoder model was evaluated by comparing the predicted labels with the output. Figure 5 shows the visualized comparison of the autoencoder labeling and the analysts labeling.

**Table 2** Performance measurements of classification models

|                          | Accuracy | Precision | Recall | F1-score |
|--------------------------|----------|-----------|--------|----------|
| Logistic regression      | 0.800    | 0.741     | 0.664  | 0.684    |
| Random forest classifier | 0.806    | 0.750     | 0.672  | 0.693    |



**Fig. 5** Comparison of labels given by the analysts and the autoencoder

Figure 5 shows that no significant difference can be seen between the labels (left) and the predicted labels (right). This means that the fraud prevention can be successfully solved by unsupervised models that do not rely on domain knowledge to label the data. Normal behavior user grouping (groups a, b, and c) and anomalous grouping (group b) occur in the user space; however, outliers can be found in these groups. Cases classified differently by the autoencoder have been sent back for revision to the analysts to determine whether the classification was correct or not. The results are as expected, autoencoder marked largely the same users fraud, and however, not every label is the same. This unsupervised approach of MLFS can indicate potentially mislabeled users.

## 5 Conclusions

A novel decision support system has been created that can predict with great correctness whether a new user is fraud or not. MLFS uses a logistic regression and random forest classifier to implement a supervised classification-based anomaly detection that relies on the analysts' labeling. RF overperformed LR in every metrics, but the difference was not significant. The MLFS can reach 0.8 accuracy, 0.75 precision, and 0.67 recall on the IBM Cloud Prepared data set. A recommendation system has been proposed that provides a strong support for the analysts. Later on, it can also replace human work, resulting in a much faster registration process, because a new user does not have to wait for the analysts to examine its case.

The implementation of the proposed MLFS system makes the user fraud prevention more deterministic, the human factors of the analysts will not have such an impact on the decision, so the filtering process can be much more consistent.

This system has brought into focus features for the analyst that have not been observed with great emphasis; however, the autoencoder and the correlation analysis showed that they have a great impact on the label. The results obtained during the feature and correlation analysis were demonstrated several times to the analyst teams. Their attention was drawn to several features that showed a high correlation with the label, but so far it has not been observed with great attention, thereby improving the accuracy of their work, and thus even more users who harm IBM Cloud and the company can be filtered out in advance. This information has been implemented into the actual decision-making system.

The developed process can provide feedback to the classification. The decision-making of the analysts can be supervised thus improving its effectiveness. This can improve the quality of the labeling, resulting in more accurate models.

Future possibility is to try out additional machine learning or deep learning models to increase the performance of the novel process. For this, it is necessary to collect more data, both labeled and unlabeled ones.

**Acknowledgments** The research reported in this paper is part of project no. BME-NVA-02, implemented with the support provided by the Ministry of Innovation and Technology of Hungary from the National Research, Development and Innovation Fund, financed under the TKP2021 funding scheme.

## References

1. P. Koehler, A. Anandasivam, D. Ma, Cloud services from a consumer perspective, in *Proceedings of the 16th Americas Conference on Information Systems (AMCIS 2010)*, Lima, Peru (2010)
2. Internet Crime Complaint Centre IC3, *Internet Crime Report 2021*. Federal Bureau of Investigation (2021)
3. D.S. Griffith, The Computer Fraud and Abuse Act of 1986: a measured response to a growing problem. *Vand. L. Rev.* **43**, 453 (1990)
4. L.I. Krauss, A. MacGahan, *Computer Fraud and Countermeasures* (Prentice-Hall, Englewood Cliffs, 1979)
5. A.A.Z. Mansour, A. Ahmi, O.M.J. Popoola, A. Znaimat, Discovering the global landscape of fraud detection studies: a bibliometric review. *J. Financial Crime* **29**(2), 701–720 (2022)
6. Association of Certified Fraud Examiners, *Report to the nations on occupational fraud and abuse*. Association of Certified Fraud Examiners (2002)
7. A. Abdallah, M.A. Maarof, A. Zainal, Fraud detection system: a survey. *J. Netw. Comput. Appl.* **68**, 90–113 (2016)
8. A.O. Hoffmann, C. Birnbrich, The impact of fraud prevention on bank-customer relationships: an empirical investigation in retail banking. *Int. J. Bank Marketing* **30**, 390–407 (2012)
9. V. Chandola, A. Banerjee, V. Kumar, Anomaly detection: a survey. *ACM Comput. Surv.* **41**(3), 1–58 (2009)
10. T. Pourhabibi, K.-L. Ong, B.H. Kam, Y.L. Boo, Fraud detection: a systematic literature review of graph-based anomaly detection approaches. *Decision Support Syst.* **133**, 113303 (2020)
11. M. Chen, A.A. Ghorbani, et al., A survey on user profiling model for anomaly detection in cyberspace. *J. Cyber Security Mob.* **8**(1), 75–112 (2019)



12. R. Ramachandran, R. Nidhin, P. Shogil, Anomaly detection in role administered relational databases—a novel method, in *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)* (IEEE, Piscataway, 2018), pp. 1017–1021
13. A.G. Marakhtanov, E.O. Parenchenkov, N.V. Smirnov, Detection of fictitious accounts registration, in *2021 International Russian Automation Conference (RusAutoCon)* (IEEE, Piscataway, 2021), pp. 226–230
14. B. Sharma, P. Pokharel, B. Joshi, User behavior analytics for anomaly detection using LSTM autoencoder-insider threat detection, in *Proceedings of the 11th International Conference on Advances in Information Technology* (2020), pp. 1–9
15. Z. Ding, L. Liu, D. Yu, S. Huang, H. Zhang, K. Liu, Detection of anomaly user behaviors based on deep neural networks, in *2021 IEEE 20th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)* (IEEE, Piscataway, 2021), pp. 1240–1245
16. V. Hodge, J. Austin, A survey of outlier detection methodologies. *Artif. Intell. Rev.* **22**(2), 85–126 (2004)
17. O. Alghushairy, R. Alsini, T. Soule, X. Ma, A review of local outlier factor algorithms for outlier detection in big data streams. *Big Data Cognit. Comput.* **5**(1), 1 (2020)
18. M. Goldstein, S. Uchida, A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data. *PLOS ONE* **11**, 1–31, 04 (2016)
19. X. Qu, L. Yang, K. Guo, L. Ma, M. Sun, M. Ke, M. Li, A survey on the development of self-organizing maps for unsupervised intrusion detection. *Mob. Netw. Appl.* **26**(2), 808–829 (2021)
20. S. Omar, A. Ngadi, H.H. Jebur, Machine learning techniques for anomaly detection: an overview. *Int. J. Comput. Appl.* **79**(2), 33–41 (2013)
21. E. Carrizosa, D.R. Morales, Supervised classification and mathematical optimization. *Comput. Oper. Res.* **40**(1), 150–165 (2013)
22. J.E. Van Engelen, H.H. Hoos, A survey on semi-supervised learning. *Mach. Learn.* **109**(2), 373–440 (2020)
23. C.S. Hilas, J.N. Sahalos, User profiling for fraud detection in telecommunication networks, in *5th International Conference on Technology and Automation* (2005), pp. 382–387
24. T. Lane, C.E. Brodley, An application of machine learning to anomaly detection, in *Proceedings of the 20th National Information Systems Security Conference*, vol. 377, Baltimore, USA (1997), pp. 366–380
25. R. Primartha, B.A. Tama, Anomaly detection using random forest: A performance revisited, in *2017 International Conference on Data and Software Engineering (ICoDSE)* (IEEE, Piscataway, 2017), pp. 1–6
26. J. Zhang, M. Zulkernine, A. Haque, Random-forests-based network intrusion detection systems. *IEEE Trans. Syst. Man Cyber. Part C (Appl. Rev.)* **38**(5), 649–659 (2008)
27. M.A.M. Hasan, M. Nasser, B. Pal, S. Ahmad, Support vector machine and random forest modeling for intrusion detection system (IDS). *J. Intell. Learn. Syst. Appl.* **2014**, 45–52 (2014)
28. C. Kater, R. Jäschke, You shall not pass: detecting malicious users at registration time, in *Proceedings of the 1st International Workshop on Online Safety, Trust and Fraud Prevention* (2016), pp. 1–6
29. K. Kirasich, T. Smith, B. Sadler, Random forest vs logistic regression: binary classification for heterogeneous datasets. *SMU Data Sci. Rev.* **1**(3), 9 (2018)
30. L. McInnes, J. Healy, J. Melville, UMAP: Uniform manifold approximation and projection for dimension reduction (2018). Preprint arXiv:1802.03426
31. D.R. Anderson, K.P. Burnham, W.L. Thompson, Null hypothesis testing: problems, prevalence, and an alternative. *J. Wildlife Manag.* **64**, 912–923 (2000)
32. T.K. Kim, T test as a parametric statistic. *Korean J. Anesthesiol.* **68**(6), 540–546 (2015)

# Big Data Challenges in Retail Sector: Perspective from Data Envelopment Analysis



Praveen M. Kulkarni , Prayag Gokhale , and Padma S. Dandannavar 

## 1 Introduction

Technology advancement has provided an opportunity to the organizations to collect data related to customers for providing better services to the customers [1]. The data generated through various methods of technology supports in creating competitive advantage for the organization [2].

Research conducted by Aktas Emel and Meng Yuwei, 2017 [3], mentions that there would be an increase in the application of big data in the retail sector by 60% by 2025.

Although big data management implementation would increase in the retail sector, there are challenges in implementing big data applications in the retail sector [4].

Studies related to Indian retail sector show that Indian retail market is expected to reach \$1.3 trillion by 2025; this demands for implementation of latest technology such as big data management in understanding the customer's [5].

This is evitable as Indian retail sector is experiencing tremendous growth in both demographically and economically in India [5]. Further, the online retail market has

---

P. M. Kulkarni (✉)

KLS Institute of Management Education and Research, Hindwadi, Belagavi, India

P. Gokhale

KLE Dr. M. S. Sheshgiri College of Engineering and Technology, Department of MBA, Udyambag, Belgaum, India

P. S. Dandannavar

KLS Gogte Institute of Technology, Department of Computer Science and Engineering, Udyambag, Belgaum, India

e-mail: [padmad@git.edu](mailto:padmad@git.edu)

grown from \$6 billion in the year 2015 to \$70 billion in the year 2019, this indicates that offline retailing and online retailing has increased in India [5].

However, there are challenges related to implementing big data in retail sector; they are (a) data privacy, (b) data credibility, (c) data analysis manpower, (d) management culture of data management, (e) data security, (f) top management support, (g) data decision making, (h) affordability and operational cost, (i) technology infrastructure, and (j) data related to specific customers [2, 6, 7].

In relation to the research with regard to the challenges of implementing big data in retail sector in India, there is limited information to the body of knowledge related to big data management in retail sector of India [8, 9]; hence, this study is undertaken to understand the challenges witnessed by the retail sector in India in implementing big data technology.

The present study has applied data envelopment analysis method which is a linear programming methodology which numerically shows the efficiency of different entities in the study [10, 11]. Hence, the present study has adopted this method of analysis to understand the various level of challenges with regard to implementation of big data management in the retail sector of India.

## 2 Literature Review

In this section, the study presents the information with regard to the literature review from the perspective of big data in retail management.

The present literature review would consist of two sections: firstly, information related to big data in the retail sector, and secondly, challenges of implanting big data in the retail sector of India.

### 2.1 *Big Data in the Retail Sector*

Big data and retailing are almost related, as we find data related to stores which sell thousands of products and have billions of financial transactions [12]. For instance, Walmart has a business operation in more than 10,000 stores in 20 countries and more and have service capacity to offer products to more than 30 million customers on everyday basis through its network of stores across the globe [13].

Further, from the perspective of data from the customers in the retail sector show that large amount of data is produced with regard to purchase, buying behavior, and volume of transactions [14].

These data augment information related to inventory management, supply chain management, and providing offers and discounts to the customers.

For instance, predictive analytics provides information on the real-time data for preparing the stock of products in the retail stores, which can reduce the cost of inventory in the stores [15].

Still, there are considerable areas of development in the retail sector in relation to big data, even though this sector is on forefront for creating and implementing big data in the sector. However, there are areas of development for implementing this technology in retail sector. For instance, there are fewer organizations who have the capability to gather and analyses the data and take complete benefit of data analyzed through big data technology for retail sector.

Further, larger organizations are not eager to invest at the level that would be appropriate to benefit of big data and there are scuffle to gain actionable consumer insights from the increasing data availability from the perspective of retailers, customers, and supply chain management [16].

These explanations propose that retailing, for both practitioners and academicians, is at the epicenter of a storm of big data prospects and challenges, which demands additional work on how to derive extra value from big data.

## ***2.2 Challenges of Implementing Big Data in Retail Sector***

Development in the area of information technology and reduced cost of data collection have provided an opportunity to the organizations to integrate data for business development, especially for the retail sector.

This abundance of data supports the retail sector to develop competitive advantage by understanding the retail buying behavior and mapping the future business development in the retail sector.

Big data technology can collect data and also have the capability to trace the customers buying behavior, however the major concern level of data privacy and integration of data for the gaining meaningful insights about the customers.

Another, challenge with regard to implementation of big data technology in the retail sector is the data credibility; this related to with regard to quality of data which is applied for the process of data analysis.

Another, factor associated with big data analysis is the lack of shortage of talented workforce to conduct the analysis of the data and provide meaningful information for decision makers in the retail sector.

One of the most critical observations is with regard to the development of management culture to integrate big data culture for improving customer satisfaction and management practices in the retail sector.

The study is from the perspective of academic research; however, future research from the marketing and big data in the retail sector needs deeper understanding and provide greater quantitative analysis. The study analysis would benefit the business practices and enhance higher customer satisfaction in retail sector of India.

### 3 Research Methods

The Data Envelopment Analysis (DEA) procedure presented by Abraham Charnes and contemporaries estimate an efficiency frontier by bearing in mind the top performance observations (extreme points) which “envelop” the residual observations by means of mathematical programming techniques. The notion of efficiency can be demarcated as a ratio of outputs to inputs:

$$\text{Efficiency} = \frac{\text{Outputs}}{\text{Inputs}} \tag{1}$$

With the intention that an inefficient unit can turn out to be efficient by increasing products (output) maintaining the equivalent level of employed resources, or by decreasing the used resources and retaining the equivalent production level, or by a blend of both  $j = 1, 2, 3, m$ , DMUs through  $x_i \mid i = 1, 2, 3, n$  inputs to result in  $y_r \mid r = 1, 2, 3$ , outputs and multipliers  $v_i$  and  $u_r$  allied with inputs and outputs, the efficiency expression presented in (1) can be validated as the ratio between weighted outputs to inputs:

$$\text{Efficiency} = \frac{\sum_{r=1}^s u_r y_{jr}}{\sum_{i=1}^n v_i x_{ji}} \tag{2}$$

In Charnes et al. [17], the degree for the technical efficiency and the multipliers, for an explicit DMU, is appraised by explaining the fractional programming:

$$\max \frac{\sum_{r=1}^s u_r y_{jr}}{\sum_{i=1}^n v_i x_{ji}} \mid \sum_{r=1}^s u_r y_{jr} - \sum_{i=1}^n v_i x_{ji} \leq 0 \tag{3}$$

With  $i, j, r$ , and positive  $v_i, u_r$ . The problem denominates the CCR “constant return to scale input-oriented model,” which in contrast is equal to explaining the subsequent linear programming:

$$\min (\theta) \mid \sum_{j=1}^m z_j x_{ji} \leq \theta x_{oi}; \sum_{j=1}^m z_j y_{jr} \geq y_{or}; \sum_{j=1}^m z_j = 1; z_j \geq 0 \tag{4}$$

Consequently, an efficiency score  $\theta$  varies from 0 to 1 entitling the efficiency for respective DMU. Peripheral influence of each input and output can obtained in the “Multiplier model of (3),” the peers of efficiency and particular weights in the envelopment form of (4) and similarly the probable for enhancements and slacks in an extension form of (4).

## 4 Results

Data envelopment analysis can be applied to a linear programming-based procedure and optimizing to evaluate the efficiency of respective unit. By means of refining the efficiency of respective unit, a reference set for an inefficient unit is obtained and the efficiency of several units can be equated to the efficiency frontier.

### 4.1 Project Specifications

In this study, data privacy, data credibility, data security, technology infrastructure, customer data, and cost of data are the part of decision-making unit (DMU), the was evaluated with respect to organizational culture as input variable and decision-making process and sales as output variables. The DEA form adopted in this study is the model basic radial grounded on the model constant return to scale.

### 4.2 Efficiency

The efficiency value found by the defined model is shown in Table 1. Furthermore, to the value of efficiency, its type is also be made known in Table 1.

### 4.3 Reference Set

In every DEA, the resulting method attempts to enhance the efficiency of the target unit to the maximum. This exploration process will end when either of efficiency of target unit or one or more units is = 1. Hence, for an ineffective unit minimum one other unit should have the efficiency = 1, with the identical weightages of the target unit are attained from the result of the model. Therefore, these efficient units are identified as the peer group for the inefficient unit. The peers are illustrated in Table 2.

**Table 1** Efficiency analysis through DEA

| Variables                 | Efficiency | Result      |
|---------------------------|------------|-------------|
| Data privacy              | 0.311      | Inefficient |
| Data credibility          | 0.137      | Inefficient |
| Data security             | 1          | Efficient   |
| Technology infrastructure | 0.479      | Inefficient |
| Customer data             | 1          | Efficient   |
| Cost of data              | 1          | Efficient   |

**Table 2** References

| Parameters                | Peer1         | Peer2         | Peer3        |
|---------------------------|---------------|---------------|--------------|
| Data privacy              | Data security | Cost of data  | –            |
| Data credibility          | Data security | Customer data | Cost of data |
| Data security             | Data security | –             | –            |
| Technology infrastructure | Data security | Cost of data  | –            |
| Customer data             | Customer data | –             | –            |
| Cost of data              | Cost of data  | –             | –            |

**Table 3** Peer frequencies

| Parameters    | Frequencies |
|---------------|-------------|
| Data security | 4           |
| Customer data | 2           |
| Cost of data  | 4           |

**Table 4** Lambdas

|                           | Data privacy | Data credibility | Data security | Technology infrastructure | Customer data | Cost of data |
|---------------------------|--------------|------------------|---------------|---------------------------|---------------|--------------|
| Data privacy              | 0            | 0                | 0.143         | 0                         | 0             | 0.448        |
| Data credibility          | 0            | 0                | 0.026         | 0                         | 0.031         | 0.165        |
| Data security             | 0            | 0                | 1             | 0                         | 0             | 0            |
| Technology infrastructure | 0            | 0                | 0.318         | 0                         | 0             | 0.223        |
| Customer data             | 0            | 0                | 0             | 0                         | 1             | 0            |
| Cost of data              | 0            | 0                | 0             | 0                         | 0             | 1            |

Table 3 also shows the number of the repeated peer units.

#### 4.4 $\lambda$ (Weights for Peer Units)

Altering the value of each input and output in such a mode that the deliberated unit is traced on the efficiency frontier, then the hypothetical unit located on the efficiency frontier can be regarded as the virtual unit.  $\lambda$  denotes the grouping of the peer units utilized to build each virtual unit. The values of  $\lambda$  are shown in Table 4.

#### 4.5 Weights (Values of the Variables for the Primary Model)

Tables 5 and 6 show the values of the variables for the primary model, which  $v_i$  is coefficient or weight assigned by DEA to input and  $u_r$  is coefficient or weight assigned by DEA to output.

**Table 5** Input weights

| Parameters                | Organization culture | Manpower |
|---------------------------|----------------------|----------|
| Data privacy              | 0.001                | 0        |
| Data credibility          | 0.001                | 0        |
| Data security             | 0.003                | 0.001    |
| Technology infrastructure | 0.002                | 0        |
| Customer data             | 0.001                | 0.002    |
| Cost of data              | 0.002                | 0        |

**Table 6** Output weights

| Parameters                | Decision making | Sales |
|---------------------------|-----------------|-------|
| Data privacy              | 0               | 0.001 |
| Data credibility          | 0               | 0.001 |
| Data security             | 0               | 0.002 |
| Technology infrastructure | 0               | 0.002 |
| Customer data             | 0.001           | 0     |
| Cost of data              | 0               | 0.002 |

**Table 7** Input slacks

| Parameters                | Organization culture | Manpower |
|---------------------------|----------------------|----------|
| Data privacy              | 0                    | 0        |
| Data credibility          | 0                    | 0        |
| Data security             | 0                    | 0        |
| Technology infrastructure | 0                    | 0        |
| Customer data             | 0                    | 0        |
| Cost of data              | 0                    | 0        |

**Table 8** Output slacks

| Parameters                | Decision making | Sales |
|---------------------------|-----------------|-------|
| Data privacy              | 8.17            | 0     |
| Data credibility          | 0               | 0     |
| Data security             | 0               | 0     |
| Technology infrastructure | 41.052          | 0     |
| Customer data             | 0               | 0     |
| Cost of data              | 0               | 0     |

### 4.6 *Input and Output Slacks*

The slacks related to respective units are shown respectively in Tables 7 and 8.

### 4.7 *Target Values*

Table 9 presents the actual and target values of each input.

Table 10 presents the actual and target values of each output.



**Table 9** Inputs and target inputs

| Parameters                | Organization culture | Manpower      |
|---------------------------|----------------------|---------------|
| Data privacy              | 825 → 256.226        | 555 → 172.37  |
| Data credibility          | 675 → 92.678         | 427 → 58.628  |
| Data security             | 217 → 217            | 668 → 668     |
| Technology infrastructure | 378 → 181.008        | 523 → 250.443 |
| Customer data             | 127 → 127            | 404 → 404     |
| Cost of data              | 503 → 503            | 172 → 172     |

**Table 10** Outputs and target outputs

| Parameters                | Decision making | Sales     |
|---------------------------|-----------------|-----------|
| Data privacy              | 384 → 392.17    | 346 → 346 |
| Data credibility          | 169 → 169       | 121 → 121 |
| Data security             | 178 → 178       | 413 → 413 |
| Technology infrastructure | 198 → 239.052   | 274 → 274 |
| Customer data             | 936 → 936       | 139 → 139 |
| Cost of data              | 819 → 819       | 641 → 641 |

## 5 Discussion

As per Table 1, it can be observed that Data security in adopting big data customer data and cost of data are the variables which are efficient in a Retail Set-up; this indicates that the adoption of big data in the retail sector would be positively supported by the variables which are 100% efficient. Whereas on other hand, if we look at the efficiency of Privacy of the data, it is just 31.11%, which indicates that the issue with data privacy of the retail customer is significantly inefficient and will have a negative impact on the implementation of big data in the retail sector. Similarly, the credibility of the data is highly inefficient with just 13.7% and may cause instability in the model as the data collected might not be credible to support the results predicted by the defined model. The technology infrastructure available in the retail sector is also found to be inefficient with 47.9%.

## 6 Conclusion

The retail sector is one of the fastest growing sectors and will dominate the economies of the world. So, most of the retailers are aiming to make optimum use of the customers data to deploy predictive models and induce the customers to purchase added products or persuade the consumers for higher unplanned purchases. The analysis done in this paper indicates that the deployment of big data model or analysis in the retail sector would be highly beneficial to the retailers, with variables like data security, customer data, and cost of the data being highly efficient in successfully implementing the defined model, but on the other hand, the issues

like credibility of the data, privacy of the data, and the technology infrastructure available in the retail sector may significantly hamper the effective implementation of the model. Therefore, though the implementation of the big data model in the retail sector carries a lot of benefits in terms of predictability, one should be vigilant while using the data for the effective deployment of the defined model.

## References

1. C. Raddats, P. Naik, A.Z. Bigdeli, Creating value in servitization through digital service innovations. *Ind. Mark. Manag.* **104**, 1–13 (2022)
2. M. Roe, K. Spanaki, A. Ioannou, E.D. Zamani, M. Giannakis, Drivers and challenges of internet of things diffusion in smart stores: A field exploration. *Technol. Forecast. Soc. Chang.* **178**, 121593 (2022)
3. E. Aktas, Y. Meng, An exploration of big data practices in retail sector. *Logistics* **1**(2), 12 (2017)
4. M. Naeem, T. Jamal, J. Diaz-Martinez, S.A. Butt, N. Montesano, M.I. Tariq, E. De-la-Hoz-Franco, E. De-La-Hoz-Valdiris, Trends and future perspective challenges in big data, in *Advances in Intelligent Data Analysis and Applications*, (Springer, Singapore, 2022), pp. 309–325
5. N. Kumar, An overview of emerging technologies in the Indian retail industry, in *Global Challenges and Strategic Disruptors in Asian Businesses and Economies*, (IGI Global, Hershey, 2021), pp. 247–256
6. M.A.E.A. Youssef, R. Eid, G. Agag, Cross-national differences in big data analytics adoption in the retail industry. *J. Retail. Consum. Serv.* **64**, 102827 (2022)
7. M.H. Sazu, S.A. Jahan, How big data analytics impacts the retail management on the European and American markets? *CECCAR Bus. Rev.* **3**(6), 62–72 (2022)
8. S.A. Gawankar, A. Gunasekaran, S. Kamble, A study on investments in the big data-driven supply chain, performance measures and organisational performance in Indian retail 4.0 context. *Int. J. Prod. Res.* **58**(5), 1574–1593 (2020)
9. N. Verma, J. Singh, An intelligent approach to big data analytics for sustainable retail environment using Apriori-MapReduce framework. *Ind. Manag. Data Syst.* **117**, 1503 (2017)
10. S. Brajesh, Big data analytics in retail supply chain, in *Big Data: Concepts, Methodologies, Tools, and Applications*, (IGI Global, Hershey, 2016), pp. 1473–1494
11. W.W. Cooper, L.M. Seiford, K. Tone, *Data Envelopment Analysis: A Comprehensive Text with Models, Applications, References and DEA-Solver Software*, 2nd edn. (Springer, New York, 2007)
12. M.G. Dekimpe, Retailing and retailing research in the age of big data analytics. *Int. J. Res. Mark.* **37**(1), 3–14 (2020)
13. F. Xiong, M. Xie, L. Zhao, C. Li, X. Fan, Recognition and evaluation of data as intangible assets. *SAGE Open* **12**(2), 21582440221094600 (2022)
14. L.L. Har, U.K. Rashid, L. Te Chuan, S.C. Sen, L.Y. Xia, Revolution of retail industry: From perspective of retail 1.0 to 4.0. *Procedia Comput. Sci.* **200**, 1615–1625 (2022)
15. E. Nilsson, Changes in the retail landscape—a Nordic perspective. *Int. Rev. Retail Distrib. Consum. Res.* **32**(4), 349–350 (2022)
16. X. Lyu, F. Jia, B. Zhao, Impact of big data and cloud-driven learning technologies in healthy and smart cities on marketing automation. *Soft. Comput.*, 1–14 (2022). <https://doi.org/10.1007/s00500-022-07031-w>
17. A. Charnes, W.W. Cooper, E. Rhodes, Measuring the efficiency of decision making units. *Eur. J. Oper. Res.* **2**(6), 429–444 (1978)

**Part III**  
**Bigdata and Data Management Systems**

# Restoration of Ancient Kannada Handwritten Palm Leaf Manuscripts Using Image Enhancement Techniques



Parashuram Bannigidad  and S. P. Sajjan 

## 1 Introduction

In the recent days, the research area of handwritten character recognition has got much attention towards ancient inscriptions, since they contain lots of unfolding knowledge in the field of science, literature, astronomy, medicine, etc. The materials used to write these inscriptions are paper, palm leaf, stone rocks, and temple walls, etc., and these materials are now degrading in nature due to climatic condition, ink bleeding, lack of attention, and unscientific storage. In digital image processing, the binarization of document image is often the first stage. Ancient documents are ruined, where extensive noise in the background or lots of changes exists. Hence, it is very difficult to categorize foreground and background pixels. Figure 1 shows a sample manuscript of historical degraded Kannada handwritten manuscripts written on palm leaf.

The goal of this research is to apply local and global thresholding to improve the quality of ancient Kannada handwritten palm leaf manuscripts that have already degraded. There has been very little research work done in this area in the literature. The global threshold, introduced by N. Otsu, defines a universal value for all pixel of the image's intensities in order to distinguish between themselves as foreground and background [1]. Non-uniformly distributed noise in an image cannot be removed with a global threshold. There is no way to use a global threshold to get rid of the noise in an image if it is not equally distributed. In contrast, local thresholding, where the threshold varies dependent on local region, offers an adaptable solution for images with distinct background intensities described by B. Gatos et al. [6]. Niblack thresholding is a local thresholding method that

---

P. Bannigidad · S. P. Sajjan (✉)

Department of Computer Science, Rani Channamma University, Belagavi, India



**Fig. 1** Images of degraded historical Kannada handwritten documents written on palm leaf

calculates a threshold for each pixel based on the mean and standard deviation of the pixel's local neighborhood was explained by W. Niblack [3]. Sauvola used two algorithms to calculate the threshold of each pixel based on the local mean, and standard deviation (SD) has been proposed by J. Sauvola and M. Pietikainen [4]. Neha Kundal and Anantdeep analyzed the performance of a novel historical document restoration algorithm based on Sauvola thresholding, as well as a hybrid method integrating both with certain local filters [2]. N Venkata Rao et al. proposed cleaning outdated document images using a modified Iterative Global Threshold [7]. Degraded Kannada handwritten paper inscriptions were restored using image enhancement techniques (Hastapratī) has been proposed by Parashuram Bannigidad and Chandrashekar Gudada [5]. B. Gatos et al. and E. Kavalieratou proposed native threshold, which provides an adaptive result for images with variable background intensities, with the threshold value varying on the attributes of the entire image [6, 9]. Images of degraded non-uniformly illuminated historical Kannada handwritten documents were restored has been proposed by Parashuram Bannigidad and Chandrashekar Gudada [10]. A combined method for binarizing historical Tibetan document images was described by Han, Yuehui et al. [11]. Sauvolanet is a degraded document binarization adaptive learning sauvola network has been evolved by Li, Deng, Yue Wu, and Yicong Zhou [12]. "Text Line Segmentation with LBP Features for Digitization and Recognition of Historical Kannada Handwritten Manuscripts" has been proposed by Bannigidad, Parashuram, and Chandrashekar Gudada [13, 16]. A framework for improved binarization of degraded historical document images was described by Xiong, Wei et al. [14]. Binarization of non-uniformly illuminated document images using the K-Means clustering algorithm has been proposed by Yang, Xingxin, and Yi Wan [15].

In this paper, the Iterative Global Thresholding (IGT) is used for segmentation and the performance evaluation measures; MSE and PSNR are used as quality measurements for degraded manuscripts. In particular, the results are compared with other standard methods in the literature, such as Sauvola, Niblack, and adaptive thresholds, using our own dataset. We also used AMADI LONTARSET, a standard palm leaf dataset, to evaluate and measure the performance of our algorithm and to demonstrate that the proposed technique is exhaustive.

## 2 Proposed Method

### 2.1 Iterative Global Thresholding (IGT)

The primary goal of this research study is to improve the quality of degraded palm leaf images by using various segmentation techniques, namely, local and global thresholding and IGT. In this paper, the Iterative Global Thresholding (IGT) algorithm is developed. In order to distinguish the image's pixel intensities into text, object, and background categories, the global threshold provides a single, unified value [7]. In each iteration following image equalization, the relative proximity to background intensity is calculated. In each iteration, this method is able to handle a variety of degraded conditions. The intermediate tones are shifted toward the background, making it easy to distinguish between the foreground and background. This method does not completely remove the image's non-uniformly distributed noise. The Iterative Global Thresholding (IGT) algorithm is also implemented with AMADI\_LONTARSET, a benchmark palm leaf dataset. The quality of the image is measured by using performance evaluation measures, i.e., PSNR and MSE. The Iterative Global Thresholding (IGT) with PSNR gives better results as compared to traditional thresholding algorithms, which is discussed below:

Thresholding is an easiest form of image segmentation based on intensity values of pixels which is given in Eq. (1).

$$D(x, y) = \begin{cases} 255, & \text{if } S(S(x, y) > \text{thresh}) \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

Where,

thresh = Thresholding values

$D(x, y)$  = Final pixel values

$S(x, y)$  = Initial pixel values

The most common and significant function used for classification is the pixel intensity value because it is the primary piece of information stored in each pixel. The pixels' intensity value is shown in Eq. (1).

The various steps used for reducing the noise in the image of the manuscripts are as follows: (i) extraction of degraded (noisy) documents; (ii) converting a noisy document to a grayscale image; (iii) average background+object intensity; (iv) pixel intensity shifting towards the background of an image; (v) equalize the image by considering how the intensity of the foreground objects affects the background information; (vi) compute the mean intensity of the image; and (vii) determine the threshold value among recursive mean levels of intensity [8].

The Iterative Global Thresholding (IGT) algorithm first applies to degraded palm leaf images. The degraded palm leaf image, if it still contains noise, then detects that noise and reprocessed separately. The proposed algorithm is made up of the following steps:

**Algorithm 1 Applying Iterative Global Thresholding (IGT)**

Procedure:

Input: Palm Leaf Image

Output: Background Image

1. Applying Iterative Global Thresholding (IGT) to the palm leaf image.
2. Convert image to NumPy array.
3. Calculate the mean of all pixels as a threshold.
4. Subtract threshold from pixel:  $\text{npimage} \leftarrow (1 - \text{threshold}) + \text{npimage}$ .
5. Make it Histogram Equalization.
6. If the number of pixels transformed between 2 iterations  $< 3\%$ , then we terminate.
7. If the pixel is not already background (1), then convert to the foreground (0).

End Procedure.

**Remaining Noise Area Detection**

Apply Iterative Global Thresholding (IGT) to each identified area individually. This technique is both straightforward and efficient. It determines a global threshold for a palm leaf manuscript image iteratively. The following steps are performed in each iteration:

**Algorithm 2 The Calculated Average Pixel Value**

Procedure:

Input: Converted Palm Leaf Image

Output: The final image is binarized.

1. Subtraction of  $T_i$  from each pixel. (i.e.,  $\text{npimage} \leftarrow (1 - \text{threshold}) + \text{npimage}$ ).
2. The gray scale histogram is stretched to distribute the remaining pixels across all grey scale tones.

$\text{min\_pixel\_value} \leftarrow \text{np.min}(\text{npimage})$

$\text{npimage} \leftarrow 1 - ((1 - \text{npimage}) / (1 - \text{min\_pixel\_value}))$

3. Repeat steps 1–3 until the termination condition satisfied.
4. The ultimate image is binarized image with the value of MSE and PSNR for performance evaluation.

End Procedure.

**2.2 Mathematical Analysis**

- (i) The Eq. (2) calculates the  $T_i$  threshold values with the  $i$ th repetition of a  $M \times N$  image.

$$T_i = \frac{\sum \sum I_i(x, y)}{M \times N} \quad (2)$$

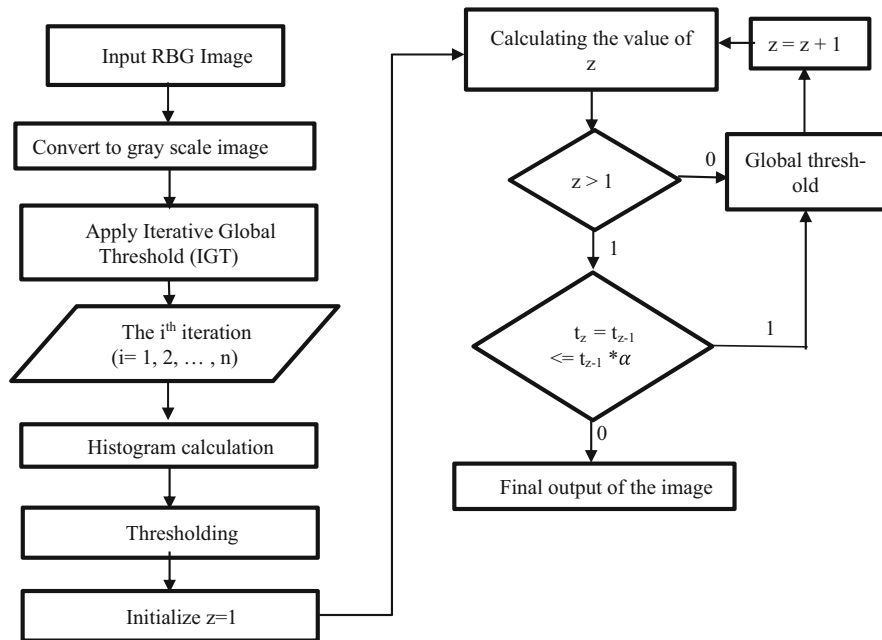


Fig. 2 The flow diagram of the proposed technique

- (ii) Every pixel in the image has its average pixel value calculated by Eq. (3) and subtracted.

$$I_s(x, y) = I_i(x, y) - T_i + 1 \tag{3}$$

- (iii) The histogram is then extended to divide the remaining pixels into all grey scale tones.

It is possible to compute the Iterative global threshold value iteratively. The proposed technique includes several steps, which are represented in the flow diagram in Fig. 2.

### 3 Experimental Results and Discussion

The historical Kannada handwritten palm leaf manuscripts are collected from e-Sahithya Documentation Forum, Bangalore. The implementation is done on a windows system containing AMD processor 8GB RAM, 2.50 GHz speed, on the system using Anaconda3 Distribution, Jupyter Notebook, Python 2.9. Camera captures mediaeval Kannada handwritten palm leaf manuscripts as shown in Fig. 3.



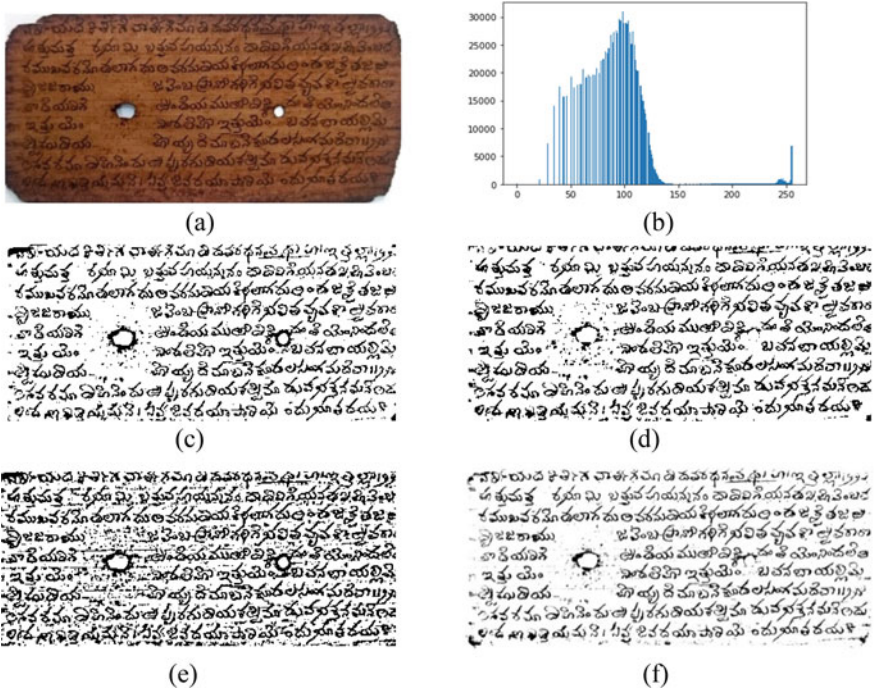
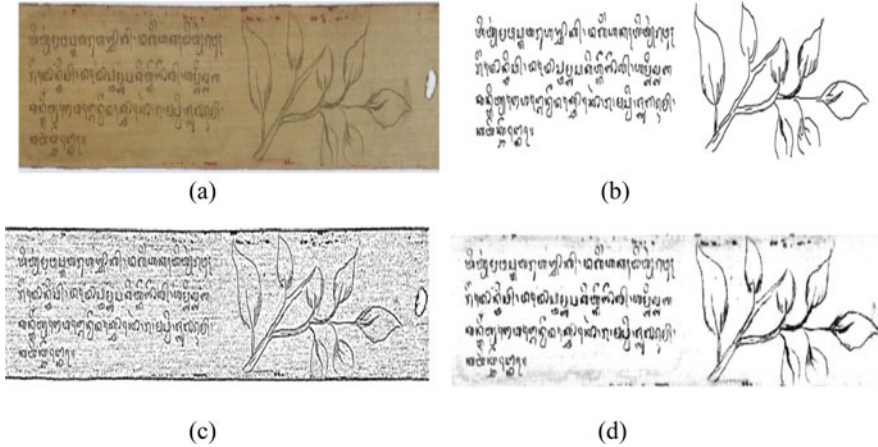


Fig. 3 (a) Sample images of historical Kannada handwritten palm leaf. (b) Histogram of original image. (c) Savoula threshold applied gray images. (d) Niblack threshold applied gray images. (e) Adaptive threshold applied gray images, (f) IGT Threshold applied gray images

Figure 3a shows a typical noisy document with non-uniformly distributed noise. The image’s background is golden brown in color. Computed histogram of the original noisy document is shown in Fig. 3b. The anticipated result was then compared to the Souvola, Niblack, and Adaptive threshold which are shown in Fig. 3c–e. All the methods produce some noise in the manuscript while maintaining image clarity. After repeatedly applying the IGT algorithm to the grayscale image till the requirement is satisfied. Each steps removes some of the noise from the manuscript. Once it is completed all the repetitive steps, the intensity values of the histogram are stretched back to the background, and the resulting IGT manuscript are shown in Fig. 3h.

The proposed algorithm was also tested and implemented on standard palm leaf datasets, such as the AMADI LONTARSET dataset. The AMADI LONTARSET dataset results are shown in Fig. 4. A typical noisy document AMADI LONTARSET dataset image is shown in Fig. 4a, and Ground truth image of AMADI LONTARSET dataset image is shown in Fig. 4b. Then, we compared the results of



**Fig. 4** (a) Sample palm leaf handwritten images of LONTARSET dataset. (b) Original palm leaf handwritten Ground Truth Image. (c) Adaptive method of MSE: 0.19 and PSNR: 7.18 value. (d) Proposed method of MSE: 0.24 and PSNR: 7.18 value

our method with the Adaptive threshold, which are given in Fig. 4c, the resultant proposed image is presented in Fig. 4d.

MSE and PSNR geometric feature values are extracted for statistical performance evaluation and calculated using Eqs. (4) and (5), respectively.

$$MSE = \frac{1}{MN} \sum \sum (g(x, y) - f(x, y))^2 \tag{4}$$

Where  $g(x, y)$  represents the output image and  $f(x, y)$  represents the input image

$$PSNR = 10 \log \left( \frac{MAX_i * MAX_i}{MSE} \right) \text{ dB} \tag{5}$$

The maximum image intensity is 255 when the pixel is represented in 8 bits. The MSE and PSNR are calculated for each of the 50 images, and the sample MSE and PSNR values are shown in the Table 1. The performance of the proposed method is visualized by epigraphists and language experts. PSNR and MSE average values are 6.198 and 0.234, respectively. According to the literature, the image quality is determined by the PSNR and MSE.

**Table 1** The proposed results are compared with other standard methods

| Segmentation methods   | Degraded Kannada handwritten palm leaf image performance evaluation |             |             |             |             |             |              |
|------------------------|---|-------------|-------------|-------------|-------------|-------------|--------------|
|                        | Evaluation methods  | Image 1     | Image 2     | Image 3     | Image 4     | Image 5     | Avg          |
| Niblack method         | PSNR  | 4.46        | 4.16        | 4.7         | 4.84        | 4.16        | 4.464        |
|                        | MSE   | 0.33        | 0.19        | 0.32        | 0.29        | 0.38        | 0.302        |
| Souvola method         | PSNR  | 4.71        | 4.57        | 5.58        | 6.52        | 4.55        | 5.186        |
|                        | MSE   | 0.33        | 0.34        | 0.27        | 0.22        | 0.34        | 0.300        |
| Adaptive threshold     | PSNR  | 5.4         | 5.25        | 6.36        | 7.1         | 5.24        | 5.870        |
|                        | MSE   | 0.28        | 0.35        | 0.29        | 0.38        | 0.23        | 0.306        |
| <i>Proposed method</i> | <i>PSNR</i>   | <i>5.65</i> | <i>5.62</i> | <i>6.61</i> | <i>7.49</i> | <i>5.62</i> | <i>6.198</i> |
|                        | <i>MSE</i>  | <i>0.27</i> | <i>0.27</i> | <i>0.21</i> | <i>0.17</i> | <i>0.27</i> | <i>0.234</i> |

## 4 Conclusion

The digitization and restoration of Kannada handwritten palm leaf manuscripts have a significant role to understand the ancient history and cultural customs, and this also helps in understanding and identifying the age of palm leaf. In this study, the Iterative Global Thresholding is applied for degraded Kannada handwritten palm leaf image and MSE and PSNR were used to evaluate the performance of the proposed techniques. The average values of PSNR and MSE is 6.198 and 0.234, respectively. In the literature, the higher the PSNR and lower the MSE determines the quality of the image. The promising results are achieved as compared to the other standard methods, namely, Souvola, Niblack, and Adaptive threshold (Gaussian+ Binary\_Inverse) in the literature. Iterative global threshold removed non-uniformly illuminated background noise and better accuracy is obtained in the proposed attempt. The proposed algorithm was also tested and implemented on other standard palm leaf benchmark datasets, such as the AMADI LONTARSET dataset, obtaining positive results. In the future, the classification and recognition of Kannada handwritten manuscripts written on palm leaf will be considered.

**Acknowledgements** The authors wish to convey their appreciation to Mr. Ashok Damlur, Chairman of the e-Sahithya Documentation Forum in Bangalore, for providing degraded Kannada handwritten palm leaf document images. The authors wish to convey their appreciation to M. W. A. Kesiman and his research team for making publicly available of AMADI\_LONTARSET series dataset.

## References

1. N. Otsu, A threshold selection method from a gray level histograms. *IEEE Trans. Syst. Man Cybern.* **9**(1), 62–66 (1979)
2. N. Kundal, Anantdeep, Performance evaluation of novel historical documents restoration algorithm. *Int. J. Comput. Sci. Eng. Technol.* **5**(7), 278–282 (2015)
3. W. Niblack, *An Introduction to Digital Image Processing*, 1st edn. (Prentice Hall, Englewood Cliffs, 1986), pp. 115–116
4. J. Sauvola, M. Pietikainen, Adaptive document image binarization. *Pattern Recogn.* **33**, 225–236 (2000)
5. P. Bannigidad, C. Gudada, Restoration of degraded Kannada handwritten paper inscriptions (Hastaprati) using image enhancement techniques, in *2017 International Conference on Computer Communication and Informatics (ICCCI)*, (IEEE, Piscataway), pp. 1–6
6. B. Gatos, I. Pratikakis, S.J. Perantoni, Adaptive degraded document image binarization. *Pattern Recogn.* **39**, 317–327 (2006)
7. N.V. Rao, A. Venkata Srinivasa Rao, S. Balaji, L. Reddy, Cleaning of ancient document images using modified iterative global threshold. *Int. J. Comput. Sci. Issues* **8**(6), 409–417 (2011)
8. B. Priyanka, H.R. Mamatha, A comparative study of binarization techniques for enhancement of degraded documents. *Int. J. Comput. Appl.* **119**(11), 15366–15369 (2015)
9. E. Kavallieratou, A binarization algorithm specialized on document images and photos, in *8th Int. Conf. on Document Analysis and Recognition*, (IEEE, Seoul, 2005), pp. 463–467
10. P. Bannigidad, C. Gudada, Restoration of degraded non-uniformly illuminated historical Kannada handwritten document images. *Int. J. Comput. Eng. Appl.* **12**, 1–13 (2018)
11. Y. Han, W. Wang, H. Liu, Y. Wang, A combined approach for the binarization of historical Tibetan document images. *Int. J. Pattern Recognit. Artif. Intell.* **33**(14), 1–21 (2019)
12. D. Li, W. Yue, Y. Zhou, Sauvolanet: Learning adaptive Sauvola network for degraded document binarization, in *International Conference on Document Analysis and Recognition*, vol. 12824, (Springer, Cham, 2021), pp. 1–36
13. P. Bannigidad, C. Gudada, Digitization and recognition of historical Kannada handwritten manuscripts using text line segmentation with LBP features. *Int. J. Emerg. Technol. Innov. Res.* **6**(4), 657–664 (2019)
14. W. Xiong, L. Zhou, L. Yue, L. Li, S. Wang, An enhanced binarization framework for degraded historical document images. *EURASIP J. Image Video Process.* **1**, 1–24 (2021)
15. X. Yang, Y. Wan, Non-uniform illumination document image binarization using K-means clustering algorithm, in *2021 IEEE 9th International Conference on Information, Communication and Networks (ICICN)*, (IEEE, Piscataway, 2021), pp. 506–510
16. P. Bannigidad, C. Gudada, Age-type identification and recognition of historical Kannada handwritten document images using HOG feature descriptors, in *Computing, Communication and Signal Processing*, vol. 810, (Springer, Singapore, 2019), pp. 1001–1010

# Mutli-Label Classification Using Label Tuning Method in Scientific Workflows



P. Shanthi, P. Padmakumari, Naraen Balaji, and A. Jayakumar

## 1 Introduction

Workflow is defined as a composite collection of tasks. Failure of a single task affects the overall performance of the workflow execution due to the dependency nature of workflows. To execute the workflow without interruption, it is essential to detect task failure proactively. The ensemble approach classifies [1] the task as Success or Failure (binary label). Further, in order to fine tune the detection and recovery mechanism, multi-label classification is proposed. It generates the intermediate labels called Partial Success and Partial Failure apart from primary labels (Success/Failure). The difference between binary label and multi-label classification is illustrated in Figs. 1 and 2.

## 2 Related Works

Existing failure detection methods mainly focus on binary label classification for detecting whether the task has failed or succeeded. If more than one task is detected to be failed, prioritization of the task which needs immediate attention is to be identified for effective rescheduling of the tasks. Naive Bayes classifier is suggested to detect the job as success or failure and job assignment is done accordingly [2]. Intelligent workflow prediction model is proposed by comparing various machine learning methods and stated that Naive Bayes classifier gives higher accuracy in task failure prediction [3]. A control theory based approach is proposed [4] to

---

P. Shanthi · P. Padmakumari (✉) · N. Balaji · A. Jayakumar  
School of Computing, SASTRA Deemed to be University, Thanjavur, Tamilnadu, India

Fig. 1 Binary label

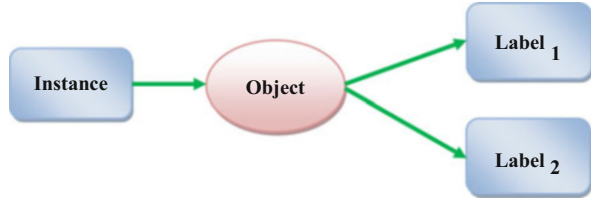
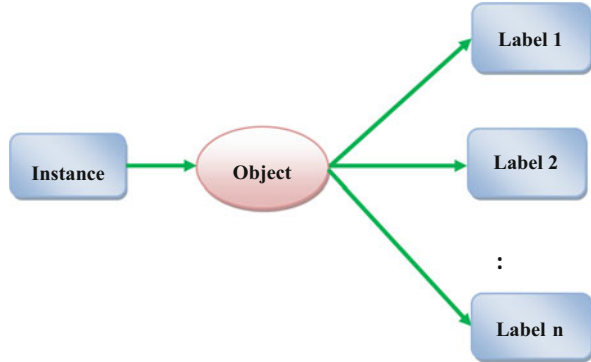


Fig. 2 Multi-label



detect task failure in scientific workflows. This approach is expensive because of the dynamic nature of distributed environments. It needs large number of controllers for individual workflows.

Reinforcement learning method is used to detect workflow task runtime and probability of failure [5]. Based on the detection result, task scheduling is performed. The proposed method reduces the unnecessary assignment of task which may lead to failure. Failure detection and node selection can be accomplished using meta-monitoring systems [6]. Generic failure detection mechanism is proposed using heart beat and event notification method. It is designed to handle the failure at task and workflow level [7]. Planner guide scheduling approach is introduced for proactive failure handling [8]. It decreases the impact of failure in terms of makespan and improves effectiveness of job rescheduling.

Availability predictor mechanism is suggested to forecast the available resources and task behavior for scheduling process [8]. The mechanism reduces average makespan and decreases the task eviction. Proactive fault tolerance using Checkpointing mechanism is recommended [9] to predict the failure based on checkpoint storage. The task failure which has been predicted with binary label classification (Success/Failure) did not consider the intermediate states. This thesis enhances the intermediate states (Partial Success/Partial Failure) using multi-label classification approach for rescheduling of task in a distributed environment.

### 3 Label Tuning in Scientific Workflow

#### 3.1 Euclidean Distance

Euclidean distance is used to measure the distance between Sample A and Sample B with multidimensional attributes in space. If  $(a_1, b_1)$  and  $(a_2, b_2)$  are two-dimensional attributes then the Euclidean distance is calculated using Eq. 1.

$$\text{Euclidean Distance} = \sqrt{(a_2 - a_1)^2 + (b_2 - b_1)^2} \tag{1}$$

Similarity checking is used to find the appropriate label for the test data. Euclidean distance is used to find similarity between the attributes and hence the state of the task by using a constant classifier and an ensemble classifier. Figure 3 shows the detailed view of the similarity function.

Test data contains the attributes of the task assigned in the distributed environment. Based on the attributes, status of the task is identified. Early detection of the task status is useful to reschedule the task accordingly, before the failure happens. Early detection of task failure is done with Training set “TR” and Test set “TS” with same “n” attributes. Similarity function uses TR<sub>n</sub> & TS<sub>n</sub> and finds the Euclidean distance ED using Eq. 2.

$$ED = \sqrt{(TS_1 - TR_1)^2 + (TS_2 - TR_2)^2 + \dots + (TS_n - TR_n)^2} \tag{2}$$

ED is used to find the similarity to assign the label in the decision table. Comparison is made with a constant classifier (Naive Bayes) and an ensemble

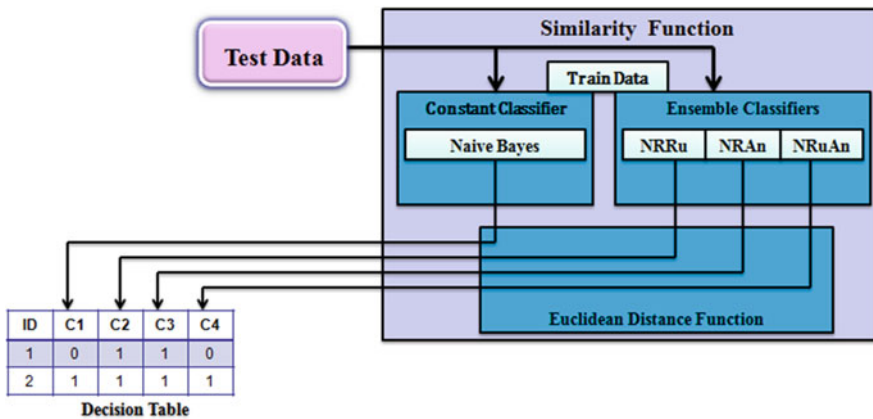


Fig. 3 Similarity function

classifier (NRRu, NRuAn, NRAn) to improve the accuracy of detection method. ED reduces unnecessary testing of the data throughout the dataset. Time taken for similarity function is less when compared to testing the whole dataset. Resultant decision table is used to decide the final status of the task.

### 3.2 Normalization

Normalization is applied for data processing in machine learning techniques. The ultimate aim of normalization is to change the values in the dataset as scaling process without distorting the information. In this work, normalization is used to fine-tune the binary labels into intermediate labels which is used to build strong recovery mechanism. It also finds the bearable task from avoiding resubmission. Min-Max normalization is used for label tuning and it helps to normalize the given data by scaling range within 0–1. Equation 3 is the general formula for Min-Max normalization.

$$\text{Min} - \text{Max} = \frac{\text{Value} - \text{Min}_a}{\text{Max}_a - \text{Min}_a} * (n\text{Max}_a - n\text{Min}_a + n\text{Min}_a) \quad (3)$$

where

Value: Current value in the attributes

$\text{Min}_a$ : Minimum value from the attributes

$\text{Max}_a$ : Maximum value from the attributes

$n\_ \text{Min}_a$ : is considered as 0

$n\_ \text{Max}_a$ : is considered as 1

Label tuning normalization is used to convert binary label (Success/Failure) into multi-label (Success/Partial Success/Failure/Partial Failure). The accuracy of constant and ensemble classifiers are considered to find the minimum and maximum value for Min-Max normalization. In this proposed work, NRAn is found to yield highest accuracy than other ensemble classifiers, so it is considered as maximum value for normalizing and constant classifier as minimum. Normalized value is identified for all the other classifiers. Decision results will reflect in the decision table.

The values obtained will be multiplied with its normalized values and then by adding all the resultant values to find the sum of the result. If the result is greater than 0 and less than or equal to 0.5 and status is assigned with “Partial Failure.” If the result is greater than 0.5 and less than 1 and status is assigned with “Partial Success.” Figure 4 shows the working of normalization in label tuning.



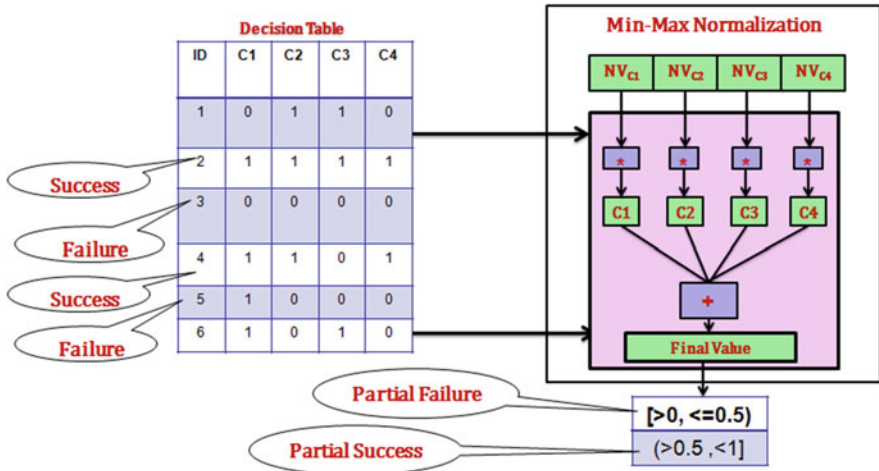


Fig. 4 Normalization in label tuning

### 3.3 Naïve-Bayes Classification

Naïve-Bayes classifiers are a group of classifiers that use probabilistic machine learning techniques to perform classification. It is mainly based on the Bayes theorem, represented using the following equation.

It is a type of supervised machine learning algorithm. It is a simple and most effective classification algorithm.

There are different types of Naïve-Bayes classifiers:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

- Multinomial Naïve Bayes
- Bernoulli Naïve Bayes
- Gaussian Naïve Bayes

### 3.4 Proposed Label Tuning Method

The proposed label tuning method consists of similarity checking and normalization approaches. In similarity checking, the constant and ensemble classification is applied on the test data and are compared. Based on the result, decision table is constructed. If the decision labels from the classifiers are “1,” the task status is considered as Success and if it is “0,” it is considered as Failure. If suppose, different

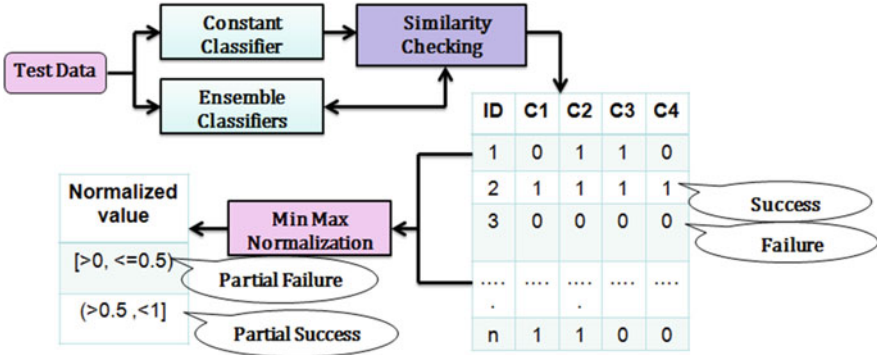


Fig. 5 Flow diagram of label tuning method

results are obtained, then deciding the final state of the task is difficult. This issue can be handled using normalization method. Normalization is used to sub-divide the basic states (Success/Failure) into intermediate states (Partial Success/Partial Failure). Figure 5 shows the overall structure of the proposed label tuning approach.

### 3.4.1 Similarity Checking

The first stage of similarity checking is to train the workflow dataset in constant and ensemble classifiers. In the second stage, the test data obtain from the monitoring layer is compared with the trained data to detect the task status. The steps in similarity checking are as follows:

- (i) Train the workflow data  $X^{tr}$  with constant (Naive Bayes) and ensemble (NRRu, NRuA, NRA) classifier by considering the following attributes:
  - Task ID
  - Datacenter ID
  - Virtual machine ID
  - Bandwidth utilization
  - CPU utilization
  - Memory utilization
  - Disk utilization
  - Depth
  - Status
- (ii) Generate the Labels  $X^{tr}label_{CC}$  and  $X^{tr}label_{ECi}$  by training the dataset  $X^{tr}$ :
  - $X^{tr}label_{CC} \leftarrow$  Constant Classifier ( $X^{tr}$ )
  - $X^{tr}label_{ECi} \leftarrow$  Ensemble Classifier ( $X^{tr}$ )
  - CC – Constant Classifier,  $ECi$  – Ensemble Classifier, where  $i = 3$  represents the combination of classifier (NRRu, NRuA, NRA)

**Table 1** Decision table

| ID | C1 | C2 | C3 | C4 | Final |
|----|----|----|----|----|-------|
| 1  | 0  | 1  | 1  | 0  | N     |
| 2  | 1  | 1  | 1  | 1  | 1     |
| 3  | 0  | 0  | 0  | 0  | 0     |
| 4  | 1  | 1  | 1  | 0  | 1     |
| 5  | 0  | 0  | 0  | 1  | 0     |
| 6  | 1  | 1  | 0  | 0  | N     |

1 Success, 0 failure, N normalization, C1 constant classifier, C2, C3, C4 ensemble classifier

- (iii) Monitor the performance of the tasks executed in the distributed environment by workflow management system periodically. Monitored data are considered as test data  $X^{te}$  to check the status of the task.
- (iv) Perform Similarity checking using Euclidean distance, by comparing the test data with trained data to form a decision table with the label as success or failure.
- Flag  $\leftarrow$  similar ( $X^{tr}_{ki}, X^{te}_i$ )
  - If Flag = 1 then
  - $X^{te}label \leftarrow X^{tr}label_k$   
where  $k = 1$  to total no. of records in train dataset,  $i =$  no. of attributes
- (v) Obtain the final label by comparing the labels of all the classifiers. The sample decision table is represented in the Table 1.

### 3.4.2 Min-Max Normalization

Min-Max Normalization is used to identify the intermediate label and to strengthen the rescheduling processes. Detected labels from similarity checking are entered into the decision table. Let “ $n$ ” be the total number of classifiers. There are six different cases possible to decide the exact status of the task.

**Case 1:** If all the classifiers results “1,” then the final status of the task is Success as given in Fig. 6.

**Case 2:** If all the classifiers results “0,” then the final status of the task is “Failure” as shown in Fig. 7.

**Case 3:** If  $(n - 1)$  number of classifiers are results “1,” and any one of the classifier is results “0,” then the final status is considered “Success” as shown in Fig. 8.

**Case 4:** If  $(n - 1)$  number of classifiers are results “0,” and any one of the classifiers results “1,” then the final status is considered “Failure” as shown in Fig. 9.

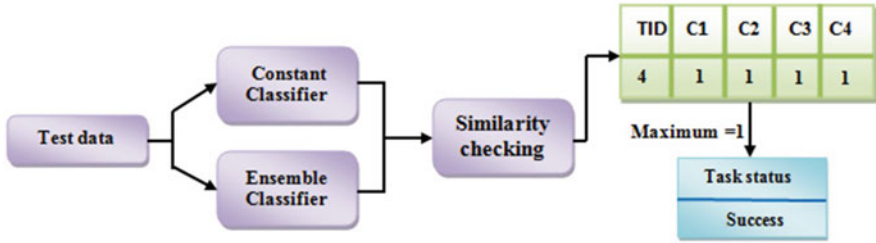


Fig. 6 Case 1 state

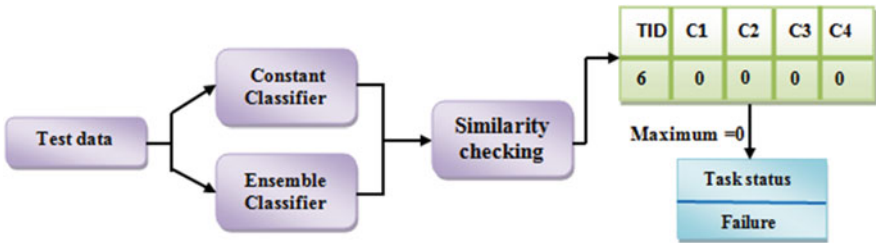


Fig. 7 Case 2 state

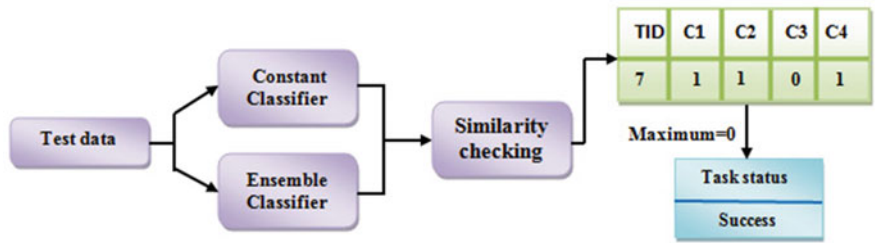


Fig. 8 Case 3 state

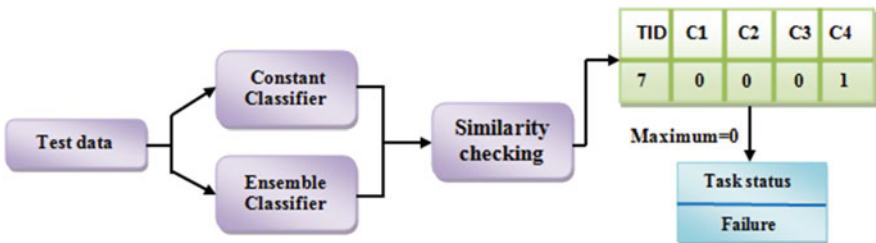


Fig. 9 Case 4 state

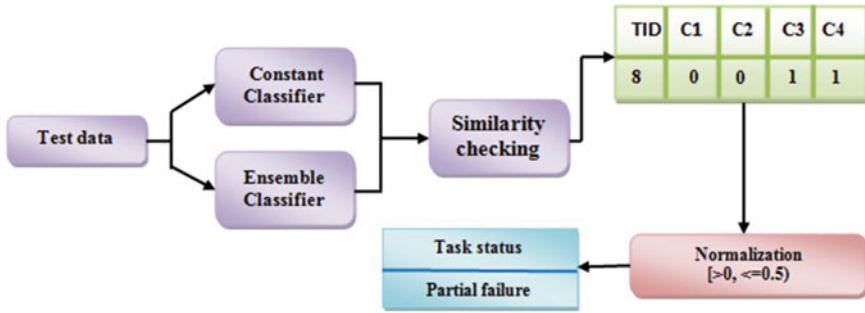


Fig. 10 Case 5 state

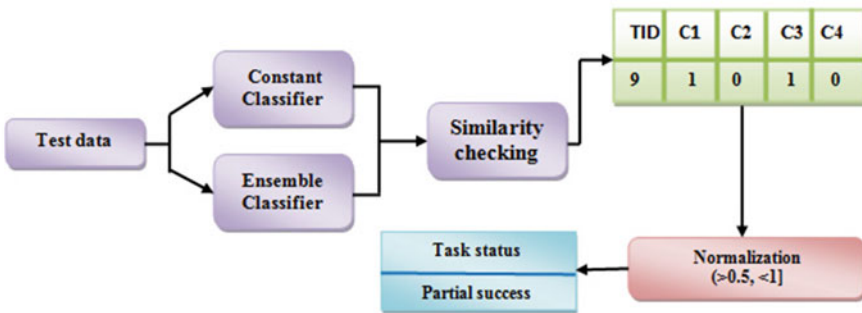


Fig. 11 Case 6 state

**Case 5:** The situation becomes difficult to decide the exact state of the task when some classifiers results “1” and some other “0.” The reasons for such fluctuations in status of the task, is the uncertainty of the task because of dynamic nature of distributed resources. It is because sometimes “Success” tasks may turn as “Failure” and “Failure” tasks may turn as “Success.” However, this problem can be solved using min-max normalization method. If the Normalized value range is in the range  $(0 < \text{Normalized value range} \leq 0.5)$ , then the final status is “Partial Failure,” as represented in Fig. 10.

Case 6: If the value is in the range  $(0.5 < \text{Normalized value range} \leq 1)$ , then the final state is “Partial Success” as given in Fig. 11.

The advantages of the proposed label tuning approach are listed below:

- (i) Strengthens the scheduling mechanism
- (ii) Prioritizes the tasks according to its health status
- (iii) Identifies the task which requires immediate attention

## 4 Results and Discussions

Few papers have discussed the prediction of the failures in the scientific workflow using machine learning techniques. The proposed and existing methods are tested using the scientific workflows such as Broadband, Epigenome, and Montage. Table 2 shows the prediction accuracy of the proposed label tuning method with the accuracy of 97.3% by comparing with the existing techniques

### 4.1 Experimental Analysis of Label Tuning Method

Table 3 shows the comparison of labels in existing and proposed methods. In the existing method, status of task limits to Success or Failure. If the label of the task is success, it continues its execution. Otherwise, resubmission/retry is encountered. In the proposed label tuning method, apart from having success/failure labels, two more states are also introduced: “Partial Success” and “Partial Failure” which is used to make recovery action based on the priority.

Validation of proposed label tuning method is done by executing workflow with 80 tasks using with and without tuning approach. Table 4 shows the prediction label with and without label tuning method. Tasks are detected only as Success or Failure in existing methods [3, 10], but the proposed label tuning method identifies the intermediate labels (Partial Success/Partial Failure), which are used to prioritize the task. Prediction with LT increases 32% of accuracy when compared to without LT.

**Table 2** Accuracy of existing and proposed method

| Methods         | Machine learning techniques | Accuracy (%) |
|-----------------|-----------------------------|--------------|
| Method 1        | Naive Bayes [3]             | 84           |
| Method 2        | Random Forest/Bagging [10]  | 80.1         |
| Method 3        | Decision Tree [10]          | 77.25        |
| Method 4        | Multi-Layer Perceptron [10] | 73.9         |
| Proposed method | Similarity & Normalization  | 97.3         |

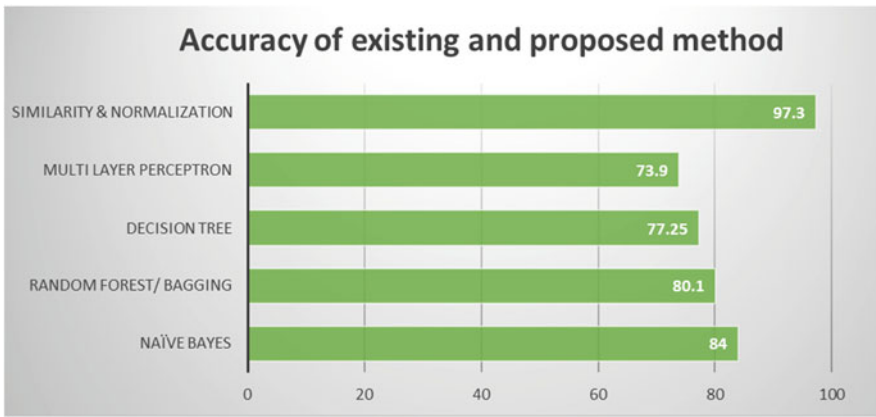
**Table 3** Labels in existing and proposed methods

| Methods         | Label   |                 |                 |         |
|-----------------|---------|-----------------|-----------------|---------|
|                 | Success | partial success | Partial failure | Failure |
| Method1 [11]    | Y       | N               | N               | Y       |
| Method2 [3]     | Y       | N               | N               | Y       |
| Method3 [12]    | Y       | N               | N               | Y       |
| Method4 [13]    | Y       | N               | N               | Y       |
| Proposed method | Y       | Y               | Y               | Y       |

**Table 4** Task failure prediction without and with label tuning (LT)

| Task range | Without LT |   | With LT |    |    |   |
|------------|------------|---|---------|----|----|---|
|            | S          | F | S       | PS | PF | F |
| 1–10       | 8          | 2 | 7       | 1  | –  | 2 |
| 11–20      | 10         | – | 10      | –  | –  | – |
| 21–30      | 6          | 4 | 5       | 1  | 1  | 3 |
| 31–40      | 5          | 5 | 5       | –  | –  | 5 |
| 41–50      | 4          | 6 | 3       | 2  | 3  | 2 |
| 51–60      | 9          | 1 | 9       | –  | 1  | – |
| 61–70      | 10         | – | 9       | 1  | –  | – |
| 71–80      | 7          | 3 | 7       | 1  | 1  | 1 |

*F* failure, *S* success, *PS* partial success, *PF* partial failure



**Fig. 12** Accuracy comparison plot

The performance evaluation of proposed label tuning method is evaluated using precision, recall, Mean Absolute Error (Merr), and Root mean squared error (Rerr). Figure 12 indicates the metric of evaluating the proposed method using precision with 96.9%, Recall with 96.4%, Merr with 19%, and Rerr with 29% for Broadband workflow dataset. For Epigenome the values of evaluating metrics are 95.8% (Precision), 96.3% (Recall), 18.8% (Merr), 22% (Rerr), respectively, and 96.1% (Precision), 96.6% (Recall), 17.2% (Merr), 23.3%, and (Rerr) for Montage dataset. Figure 13 shows the results of proposed label tuning method with accuracy of 97.5% for broadband, 97.2% for epigenome, and 96.2% for montage. The proposed label tuning method gives average of 1.5% more accuracy than existing model [1] (Fig. 14).

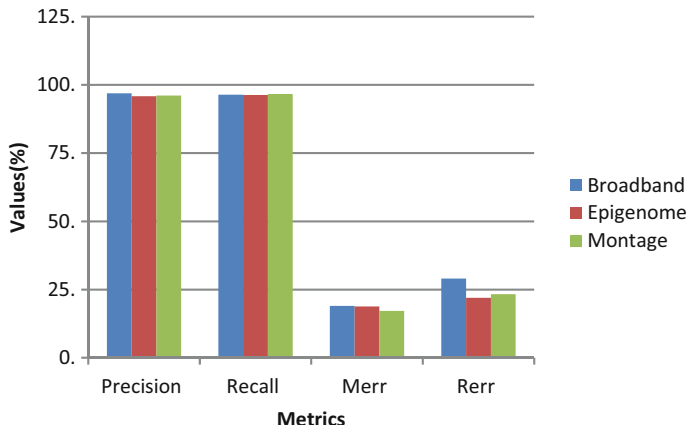
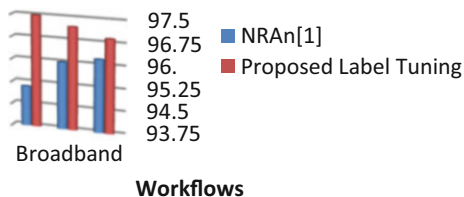


Fig. 13 Metrics

Fig. 14 Accuracy comparison of proposed and existing model



## 5 Conclusion

The main goal of label tuning method is to improve accuracy of failure detection and to apply priority-based rescheduling. The use of multi-label classification improves the accuracy of the failure detection when compared to the binary label classification. It strengthens the rescheduling of resources in distributed environments rather than resubmission. It also provides guideline to submit the tasks for execution. It avoids resubmission and executes constrained based workflow within the deadline. In future, the status of tasks identified by multi-label classification can be used to reschedule the available resources in proactive and an effective manner.

## References


1. P. Padmakumari, A. Umamakeswari, Task failure prediction using combine bagging ensemble (CBE) classification in cloud workflow. *Wirel. Pers. Commun.* **107**(1), 23–40 (2019)
2. G. Yao, Y. Ding, S. Member, K. Hao, Using imbalance characteristic for fault – Tolerant workflow scheduling in cloud systems. *IEEE Access* **9219**, 3671–3683 (2017). <https://doi.org/10.1109/TPDS.2017.2687923>



3. A. Bala, I. Chana, Intelligent failure prediction models for scientific workflows. *Expert Syst. Appl.* **42**, 980–989 (2015). <https://doi.org/10.1016/j.eswa.2014.09.014>
4. R. Ferreira, R. Filgueira, E. Deelman, E. Pairo-castineira, Using simple PID-inspired controllers for online resilient resource management of distributed scientific workflows. *Futur. Gener. Comput. Syst.* **95**, 615–628 (2019). <https://doi.org/10.1016/j.future.2019.01.015>
5. A.M. Kintsakis, F.E. Psomopoulos, P.A. Mitkas, Engineering applications of artificial intelligence reinforcement learning based scheduling in a workflow management. *Eng. Appl. Artif. Intell.* **81**, 94–106 (2019). <https://doi.org/10.1016/j.engappai.2019.02.013>
6. A. Feoktistov, R. Kostromin, I. Sidorov, A. Feoktistov, I. Sidorov, S. Gorsky, Multi-agent algorithm for re-allocating grid-resources and improving fault-tolerance of problem-solving processes. *Procedia Comput. Sci.* **150**, 171–178 (2019). <https://doi.org/10.1016/j.procs.2019.02.034>
7. W. Tan, K. Chard, D. Sulakhe, R. Madduri, I. Foster, S. Soiland-Reyes, C. Goble, Scientific workflows as services in caGrid: A Taverna and gRAVI approach, in *2009 IEEE Int. Conf. Web Serv. ICWS 2009*, (2009), pp. 413–420. <https://doi.org/10.1109/ICWS.2009.19>
8. B. Rood, M.J. Lewis, *Grid Resource Availability Prediction-Based Scheduling and Task Replication* (2009), pp. 479–500. <https://doi.org/10.1007/s10723-009-9135-2>
9. L. Zhu, J. Gu, Y. Wang, T. Zhao, Z. Cai, Optimizing the fault-tolerance overheads of HPC systems using prediction and multiple proactive actions. *J. Supercomput.* **71**, 3668–3694 (2015). <https://doi.org/10.1007/s11227-015-1458-0>
10. Z. Amin, H. Singh, N. Sethi, Review on fault tolerance techniques in cloud computing. *Int. J. Comput. Appl.* **116**, 11–17 (2015). <https://doi.org/10.5120/20435-2768>
11. T. Fahringer, R. Prodan, R. Duan, F. Nerieri, S. Podlipnig, J. Qin, M. Siddiqui, H. Truong, A. Villaz, *ASKALON: A Grid Application Development and Computing Environment* (2005), pp. 122–131. <https://doi.org/10.1109/GRID.2005.1542733>
12. S. Hwang, C. Kesselman, Grid workflow: A flexible failure handling framework for the grid, in *High Performance Distributed Computing, 2003. Proceedings. 12th IEEE International Symposium on. IEEE*, (IEEE, 2003)
13. K. Plankensteiner, R. Prodan, T. Fahringer, A. Kertész, P. Kacsuk, *Fault detection, prevention and recovery in current grid workflow systems*. *Grid and services evolution*, pp. 1–13 (2009)

# A Comparative Analysis of Assignment Problem



Shahriar Tanvir Alam , Eshfar Sagor, Tanjeel Ahmed, Tabassum Haque, Md Shoaib Mahmud, Salman Ibrahim, Ononya Shahjahan, and Mubtasim Rubaet

## 1 Introduction

In the present world, the assignment of items from multiple sources to multiple destinations is critical, and the organization has to make the assignment of an appropriate job to the right machine [1]. Wrong and inappropriate assignments waste approximately 107 hours along with \$2243 per year [2]. Due to a failure to estimate the correct assignment, it costs more than \$20 billion, or \$97 per driver, per year in the United States alone [3]. In a period of dynamic and shifting markets with high-quality products and high demand for quick services, manufacturing organizations have been compelled to maintain low operational expenses and promote customer-focused products with shorter life cycles. The assignment problem occurs frequently in practice and is a basic problem in network flow theory since it can be reduced to a number of other problems, including the shortest path, weighted matching, transportation, and minimal cost flow [4]. Furthermore, a World Bank-funded review estimated that in the “Greater Cai-ro Region” in 2010, the yearly cost of the inappropriate assignment was anticipated to be around 47 billion LE (8 billion USD), or 2400 LE per person (400 USD). This expense is expected to account for almost 15% of the overall per capita GDP [5].

Furthermore, a bottleneck is a congested area in a manufacturing system (such as an assembly line or a computer network) that causes the system to stop or move very slowly [6, 7]. Inefficiencies in the bottleneck frequently result in delays and higher production costs. Bottlenecks can disrupt the flow of the manufacturing process

---

S. T. Alam (✉) · E. Sagor · T. Ahmed · T. Haque · M. S. Mahmud · S. Ibrahim · O. Shahjahan · M. Rubaet  
Military Institute of Science and Technology, Department of Industrial and Production Engineering, Dhaka, Bangladesh  
e-mail: [shahriar@ipe.mist.ac.bd](mailto:shahriar@ipe.mist.ac.bd)

and significantly increase the time and cost of production. Bottlenecks are more likely to occur when a company begins the production process for a new product. This is because the company may encounter process issues that must be identified and resolved; this condition necessitates additional examination and fine-tuning. Controlling the production process, anticipating potential bottlenecks, and devising effective solutions should be the primary concern of any company. One of the most common causes of bottlenecks is the failure to assign the appropriate task to the appropriate machine. This consequence signifies assignment problems that create an obstacle to provide many commodities in the right place on time.

This study is analyzing a comparison of assignment problems while considering the minimization of the overall cost of supplying and fulfilling the demand in order to further understand the mentioned unbearable situation. The primary goal of this research is to develop an optimal result for an assignment problem and to provide an example of how to handle a balanced and unbalanced assignment problem. The literature review is summarized in Sect. 2 of this paper. The proposed methodology is explained in Sect. 3, whereas the conclusion and future recommendations for further research are discussed in Sect. 4.

## **2 Literature Review**

This part illustrates the existing literature relating to assignment problems. Section 2.1 discusses the literature on assignment problems, whereas Sect. 2.2 illustrates the recent study of the unbalanced assignment problem.

### ***2.1 Literature Related to the Assignment Problems***

Many researchers and practitioners in the past implemented the Hungarian method to resolve assignment problems [8–10]. A study on hospital layout design remodeling was undertaken as a Quadratic Assignment Problem (QAP) with geodesic distances, which is a configurational issue that results in inefficient transportation operations for patients, medical personnel, and material logistics [11]. The QAP is intended to allocate a set of resources to a set of locations while minimizing the overall assignment expenditure. When there are  $m$  resources and  $m$  locations, the QAP aims to reduce total assignment costs. Adding resource placement costs and multiplying inter-location distances by resource flow amounts gives the total cost of the assignment [12]. The internal transportation processes between interrelated facilities are minimized by renovating existing hospitals using Computer-Aided Design (CAD), which improves communication by creating documentation and a database. In the same year, Homayouni and Fontes [3] addressed an expansion of the flexible job shop scheduling problem by having to consider that jobs need to be transported across the shop floor by a collection of vehicles, with each production

process being assigned to one of the alternative machines [13]. In this study, the problem was solved using a mixed integer linear programming approach, which demonstrated effectiveness for small-scale instances.

A furthermore study has been conducted on the neural graph-matching network. The matching problem is converted into a constrained vertex classification task using a QAP network, which learns directly from the affinity matrix (or equivalently, the association graph), as presented in another study [14]. The embedding network strategy is utilized to achieve and even surpass the performance of state-of-the-art graph matching and QAP resolvers with a drastically decreased time cost. For the well-known QAP, Dokeroglu, Sevinc, and Cosar introduced hybrid Artificial Bee Colony (ABC) optimization techniques [15]. The robust tabu search approach is used to model bee exploration and exploitation activities. Furthermore, 125 of 134 benchmark issue instances from the *QAPLIB library* are solved optimally, with a 0.27% variation indicated for 9 large problem cases that could not be handled optimally.

Using historical data, Xiang and Liu solved the combined berth allocation and dock cranes assignment problem by accounting for the possibility of ships arriving late and an increase in the number of containers [16]. A robust model with a weighted maximum penalty model has been developed. The problem is decomposed into a deterministic multi-objective optimization problem and a stochastic subproblem, both of which are suitable to the decomposition technique. The applied method proved superior to the robust, deterministic option in terms of total estimated cost, total vessel delays, and berth and dock crane usage rates, thus making it the preferred choice. An additional study has been initiated to look at an assignment problem relating to airport gates. This challenge involves assigning a group of planes to a group of gates [17]. This study determined to make aircraft gate assignments that would work to reduce the overall walking distance of the passengers. With the integer-mixed nonlinear programming model, it had been linearized and developed with the filtered beam search, bound algorithm, and beam search algorithms. In this recent study, the facility layout problem (FLP) dealt with optimal assignments which helped to minimize the transportation cost [18]. It can be added to the problem of hospital facility planning with comprehensive clinics, testing facilities, and radiology units. The proposed method of mixed-type algorithm gathered DDE and tabu search (TS), which were performed with implemented benchmark instances from the *QAPLIB website*. Forty-two optimal and fifty-two instances were found through the proposed method.

A model was created for the assignment problem of scheduling classes of university lecturers in Vietnam [19]. Using a compromise programming approach, the model is transformed into a model with a single objective. Afterward, a genetic algorithm for the model is provided, which can generate a calendar incorporating lecturer schedules while ensuring associated conditions. Based on differential evolution and self-adaptive multi-task particle swarm optimization (SaMTPSO), an effective Evolutionary Multi-task Optimization (EMTO) solver is designed in a study [20]. After that, the algorithm is used to solve the weapon-target assignment problem on two test suites, and it is compared to other relevant algorithms to

demonstrate the algorithm's viability in resolving the identified issues. In another paper, a study is undertaken on a storage assignment problem caused by a shortage of volume in container terminal yards, and a storage-sharing approach between container terminals and dry ports is proposed as a solution [21]. A multiple-objective mixed integer programming model is created, with the goals of lowering travel distance, balancing, and maximizing shared storage, and Non-dominated Sorting Algorithm II (NSGA-II) is used to solve the problem.

## ***2.2 Literature Related to the Unbalanced Assignment Problem***

Rabbani, Khan, and Quddoos proposed a modified "Hungarian method" for resolving unbalanced assignment problems without failing to complete any job [22]. The method works by identifying rows with a single zero and crossing off the other zero in each respective column. The proposed methodology's stepwise algorithm is developed and programmed in *Java SE 11*. Following that, the proposed approach is compared against three alternatives, and it is shown to be better. Furthermore, Kumar proposed a method for resolving the unbalanced assignment problem that is capable of solving unbalanced problem by optimally assigning all jobs to the machine [23]. The unbalanced matrix is divided into some balanced submatrices and those are solved using the Hungarian method. This method has the disadvantage of frequently failing to provide a low total cost. The Hungarian method for resolving unbalanced assignment problems is based on the assumption that some jobs should be assigned to pseudo or dummy machines, but these jobs are left unexecuted by the dummy machines in the Hungarian method. However, it is sometimes impractical in real-world situations.

Moreover, Lampang, Boonjing, and Chanvarasuth introduced a new space-saving and cost-effective approach for unbalanced assignment problems [24]. This approach uses linear space complexity. The proposed method offers a lower optimal cost than [23] according to an experiment with 100,000 cost matrices. For the unbalanced assignment problem, [25] developed a graph-based twin cost matrices technique with an improved ant colony optimization algorithm. It can resolve assignment problems evenly, whether they are balanced or unbalanced, constrained or unconstrained. The twin cost matrices with variable pheromones correlate AP (Assignment problem) and TSP (Travelling Salesman Problem). The mutation method makes it easier to get to the optimal solution by reducing the likelihood that the ant colony may fall in the local optimum. According to experiments, the method produces superior outcomes when compared to other existing methods.

The aforementioned research demonstrates that, despite the fact that an array of knowledge on assignment problems is gradually accruing over time, there is still a lack of research that focuses on assessing the optimal solutions to unbalanced as well as balanced assignment problems. To overcome this gap, we propose a comparative analysis of assignment problems.

### 3 Methodology

The assignment problem is a subset of the transportation problem in which each supply point must be assigned to a demand point and each demand point must be generated. It is used in determining which employee and machine should be assigned to which job. “*Hungarian Method*” can be utilized to answer the assignment problem. The goal is to figure out how all  $n$  tasks should be completed in order to minimize overall expenses.

#### Assumptions

- The quantity of assignees and jobs are both equal.
- Every assignee must be allotted only one job.
- Every job must be submitted by a single assignee.
- $C_{ab}$  is the cost of assignee  $a$  ( $a = 1, 2, 3, 4, \dots, d$ ), accomplishing task  $b$  ( $b = 1, 2, 3, 4, \dots, d$ ).

#### 3.1 Model Formulation

A supply chain is considered where suppliers deliver stuff to companies, which further serves inventory space, which distributes the marketplace. Location and capacity allotment considerations must be made for factories and inventories. Many inventories may be utilized to accommodate consumer needs, and multiple companies may be used to refill inventory. Additionally, it is expected that units have been appropriately adjusted to ensure that one unit from a supply source produces one unit of the final good. The following inputs are needed for the model:

---

$c$  = The number of possible demand stations

---

$d$  = The number of possible plant sites

---

$g$  = The number of possible vendors

---

$k$  = The number of possible inventory sites

---

$D_b$  = Yearly market  $b$  demand

---

$M_a$  = Total production capacity of the factory at the site  $a$

---

$N_s$  = The capacity of supply at the supplier  $s$

---

$L_o$  = The capacity of inventory at the site  $o$

---

$i_a$  = The set price for placing a plant at the site  $a$

---

$i_o$  = The set price for placing an inventory at the site  $o$

---

$J_{sa}$  = Shipping expenses of a single item from a supply source  $s$  to the manufacturer  $a$

---

$J_{ao}$  = Producing and shipping expenses of a single item from the manufacturer  $a$  to the inventory  $o$

---

$J_{ob}$  = Shipping expenses of a single item from the inventory  $o$  to the customer  $b$

---

Define the subsequent decision variables:

---

$y_a = 1$  when a factory is placed at the site  $a$ , 0 otherwise

---

$y_o = 1$  when an inventory is placed at the site  $o$ , 0 otherwise

---

$z_{ob} =$  The amount is transferred from the inventory  $o$  to the market  $b$

---

$z_{ao} =$  The amount is transferred from the factory at the site  $a$  to the inventory  $o$

---

$z_{sa} =$  The amount is transferred from the supplier  $s$  to the factory at the site  $a$

---

The problem is stated as follows:

Min

$$\sum_{a=1}^d i_a y_a + \sum_{o=1}^k i_o y_o + \sum_{s=1}^g \sum_{a=1}^d J_{sa} z_{sa} + \sum_{a=1}^d \sum_{o=1}^k J_{ao} z_{ao} + \sum_{o=1}^k \sum_{b=1}^J J_{ob} z_{ob}$$

The objective function seeks to minimize the total costs of the supply chain network while keeping the following constraints in mind:

$$\sum_{a=1}^d z_{sa} \leq N_s \quad \text{for } s = 1, \dots, g \quad (1)$$

The constraint in Eq. 1 states that the total amount supplied by a supplier cannot be greater than the capacity of the supplier.

$$\sum_{s=1}^g z_{sa} - \sum_{o=1}^k z_{ao} \geq 0 \quad \text{for } a = 1, \dots, d \quad (2)$$

The restriction in Eq. 2 specifies that the total amount exported out of a facility cannot be more than the total amount of raw materials received.

$$\sum_{o=1}^k z_{ao} \leq M_a y_a \quad \text{for } a = 1, \dots, d \quad (3)$$

The restriction in Eq. 3 mandates that the factory's output must not be more than its capacity.

$$\sum_{a=1}^d z_{ao} - \sum_{b=1}^J z_{ob} \geq 0 \quad \text{for } o = 1, \dots, k \quad (4)$$

The restriction in Eq. 4 states that an inventory's sent-out quantity must not be more than its received quantity from factories.

$$\sum_{b=1}^J z_{ob} \leq L_o y_o \quad \text{for } o = 1, \dots, k \quad (5)$$

The limitation in Eq. 5 states that the quantity shipped through an inventory cannot be greater than its capacity.

$$\sum_{o=1}^k z_{ob} = D_b \quad \text{for } b = 1, \dots, J \tag{6}$$

The constraint in Eq. 6 states that the quantity sent to a client should satisfy the demand.

$$y_a, y_o \in \{0, 1\}, z_{ob}, z_{ao}, z_{sa} \geq 0 \tag{7}$$

Every factory or inventory must be either open or closed due to the constraint in Eq. 7.

For minimizing objective function, the term  $J_{ao}z_{ao}$  can play a vital role. Equations 1, 2, 3, 4, 5, 6, and 7 are dependent on the appropriate amount of finished product discharged from the production plant. The “Hungarian Method” is used in this study to fulfill demand from raw materials to finished products by assigning a job to a machine that can reduce production costs. The proper assignment of raw materials (job) to the appropriate machine can confirm the discharge of demand quantities, thereby influencing the mentioned objective function (by maximizing  $z_{ao}$ ).

### 3.2 *Balanced Assignment Problem*

The “Balanced Assignment Problem” is one in which there is the same quantity of machines and jobs. The goal is to delegate tasks to machines in a manner that results in the lowest cost achievable, given that there are “ $n$ ” jobs to complete on “ $m$ ” machines (i.e., one job to one machine). Each machine is capable of performing all tasks, albeit with varying degrees of efficiency. Therefore, the following assumption must be considered:

- $m = n$ , which means jobs and machines are equal in numbers.
- No job can be given to more than one machine.
- There is a cost associated with each machine and job.

A case study related to the “Balanced Assignment Problem” is solved following the “Hungarian Method.” The cost required to set up each machine for completing each job is presented in Table 1.

Tables 2, 3, 4, and 5 present the steps required to determine the appropriate job assignment to the machine.

**Step 1** By taking the minimum element and subtracting it from all the other elements in each row, the new table will be:

Table 2 represents the matrix after completing the 1st step.



**Table 1** Initial table of a “Balanced Assignment Problem”

| Job | Machine |      |      |      |      |
|-----|---------|------|------|------|------|
|     | M 1     | M 2  | M 3  | M 4  | M 5  |
| J 1 | 13.0    | 08.0 | 16.0 | 18.0 | 19.0 |
| J 2 | 09.0    | 15.0 | 24.0 | 09.0 | 12.0 |
| J 3 | 12.0    | 09.0 | 04.0 | 04.0 | 04.0 |
| J 4 | 06.0    | 12.0 | 10.0 | 08.0 | 13.0 |
| J 5 | 15.0    | 17.0 | 18.0 | 12.0 | 20.0 |

**Table 2** Matrix table after step 1

| Job | Machine |      |      |      |      |
|-----|---------|------|------|------|------|
|     | M 1     | M 2  | M 3  | M 4  | M 5  |
| J 1 | 05.0    | 00.0 | 08.0 | 10.0 | 11.0 |
| J 2 | 00.0    | 06.0 | 15.0 | 00.0 | 03.0 |
| J 3 | 08.0    | 05.0 | 00.0 | 00.0 | 00.0 |
| J 4 | 00.0    | 06.0 | 04.0 | 02.0 | 07.0 |
| J 5 | 03.0    | 05.0 | 06.0 | 00.0 | 08.0 |

**Table 3** Matrix table after step 2

| Job | Machine |      |      |      |      |
|-----|---------|------|------|------|------|
|     | M 1     | M 2  | M 3  | M 4  | M 5  |
| J 1 | 05.0    | 00.0 | 08.0 | 10.0 | 11.0 |
| J 2 | 00.0    | 06.0 | 15.0 | 00.0 | 03.0 |
| J 3 | 08.0    | 05.0 | 00.0 | 00.0 | 00.0 |
| J 4 | 00.0    | 06.0 | 04.0 | 02.0 | 07.0 |
| J 5 | 03.0    | 05.0 | 06.0 | 00.0 | 08.0 |

**Table 4** Matrix table after step 3

| Job | Machine |      |      |      |      |
|-----|---------|------|------|------|------|
|     | M 1     | M 2  | M 3  | M 4  | M 5  |
| J 1 | 05.0    | 00.0 | 08.0 | 10.0 | 11.0 |
| J 2 | 00.0    | 06.0 | 15.0 | 00.0 | 03.0 |
| J 3 | 08.0    | 05.0 | 00.0 | 00.0 | 00.0 |
| J 4 | 00.0    | 06.0 | 04.0 | 02.0 | 07.0 |
| J 5 | 03.0    | 05.0 | 06.0 | 00.0 | 08.0 |

**Table 5** Matrix table after step 4

| Job | Machine |      |      |      |      |
|-----|---------|------|------|------|------|
|     | M 1     | M 2  | M 3  | M 4  | M 5  |
| J 1 | 05.0    | 00.0 | 05.0 | 10.0 | 08.0 |
| J 2 | 00.0    | 06.0 | 12.0 | 00.0 | 00.0 |
| J 3 | 11.0    | 08.0 | 00.0 | 03.0 | 00.0 |
| J 4 | 00.0    | 06.0 | 01.0 | 02.0 | 04.0 |
| J 5 | 03.0    | 05.0 | 03.0 | 00.0 | 05.0 |

**Step 2** By taking the minimum element and subtracting it from all the other elements in each column.

Table 3 represents the matrix after completing the 2nd step.

**Table 6** Result of the balanced assignment problem

| Job | Machine | Cost |
|-----|---------|------|
| 1   | 2       | 08.0 |
| 2   | 5       | 12.0 |
| 3   | 3       | 04.0 |
| 4   | 1       | 06.0 |
| 5   | 4       | 12.0 |

**Step 3**

To cover all zeros, draw as few horizontal and vertical lines as possible.

$N$  = least number of lines required to cover all zeros,  $n$  = the order of the matrix

- (a) If  $N = n$ , If  $N = n$ , then the best solution can be determined.
- (b) If  $N < n$ , then go to the next step.

Table 4 represents the matrix after step 3.

**Step 4** Determine the smallest uncovered element  $x$ .

- (a) Write uncovered value = uncovered value  $- x$
- (b) Intersection value = intersection value  $+ x$
- (c) Line values (Other values) as same

This is not the best possible answer because the number of lines that would need to be drawn to cover zero is  $N = 4$  orders of matrix  $n = 5$ .

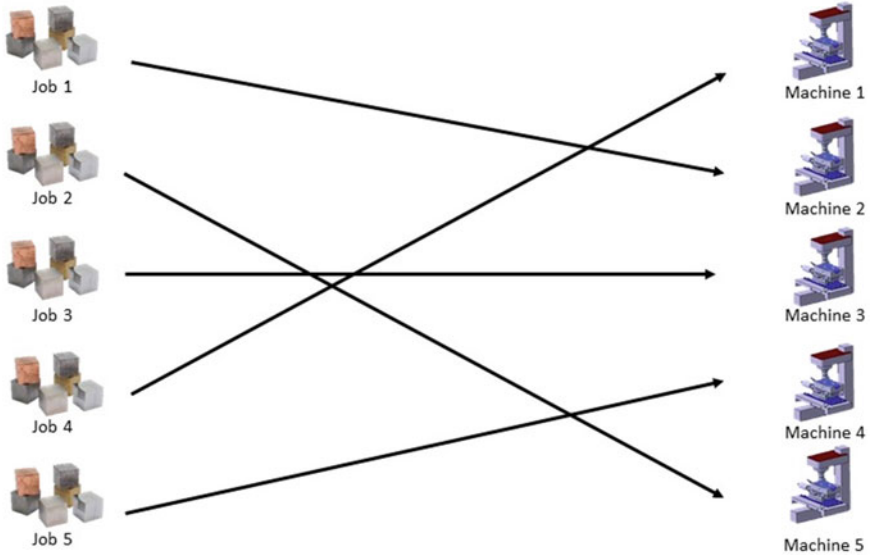
Table 5 indicates that the number of lines sketched to encompass all zeroes is five, and the order of the matrix is five. Therefore, we can formulate an assignment.

Table 6 represents which machine is assigned for which job. Figure 1 is the visual representation of the solution to the balanced assignment problem. To recapitulate, total cost =  $08.0 + 12.0 + 04.0 + 06.0 + 12.0 = 42.0$

**3.3 Unbalanced Assignment Problem**

The ‘‘Hungarian Method’’ for resolving unbalanced assignment problems is founded on delegating some jobs to dummy machines. The tasks assigned to the dummy machines are not completed. In unbalanced assignment problems, the objective is to determine the optimal ratio of the total amount of jobs ( $n$ ) to the total amount of machines ( $m$ ), in which case  $n$  and  $m$  are not equal. In this case, the following inferences are assumed:

- There are more machines than jobs or  $m > n$ .
- No job can be given to more than one machine.
- There is a cost associated with each machine and job.



**Fig. 1** Illustration of a balanced assignment problem solution

**Table 7** Initial table of an “Unbalanced Assignment Problem”

| Job | Machine |      |      |      |      |
|-----|---------|------|------|------|------|
|     | M 1     | M 2  | M 3  | M 4  | M 5  |
| J 1 | 13.0    | 08.0 | 16.0 | 18.0 | 19.0 |
| J 2 | 09.0    | 15.0 | 24.0 | 09.0 | 12.0 |
| J 3 | 12.0    | 09.0 | 04.0 | 04.0 | 04.0 |
| J 4 | 06.0    | 12.0 | 10.0 | 08.0 | 13.0 |

The following case study of the “Unbalanced Assignment Problem” is solved using “Hungarian Method.” Table 7 shows the cost of setting up each machine to complete each job.

The matrix in Table 7 is not a square, so the problem is not balanced. To make it a square matrix, a dummy job (job 5) has been added with corresponding entities zero. Tables 9, 10, and 11 present the steps required to assign the appropriate job to the machine.

Table 8 is the new modified matrix which is a balanced matrix after adding the dummy row to Table 7.

**Step 1** By taking the minimum element and subtracting it from all the other elements in each row, the new table will be:

Table 9 represents the matrix after completing step 1.

**Step 2** Take the lowest value and subtract it from each value in the perspective column.

Table 10 is representing the matrix after completing step 2.

**Table 8** The new matrix after adding a dummy row

| Job | Machine |      |      |      |      |
|-----|---------|------|------|------|------|
|     | M 1     | M 2  | M 3  | M 4  | M 5  |
| J 1 | 13.0    | 08.0 | 16.0 | 18.0 | 19.0 |
| J 2 | 09.0    | 15.0 | 24.0 | 09.0 | 12.0 |
| J 3 | 12.0    | 09.0 | 04.0 | 04.0 | 04.0 |
| J 4 | 06.0    | 12.0 | 10.0 | 08.0 | 13.0 |
| J 5 | 00.0    | 00.0 | 00.0 | 00.0 | 00.0 |

**Table 9** Matrix table after step 1

| Job | Machine |      |      |      |      |
|-----|---------|------|------|------|------|
|     | M 1     | M 2  | M 3  | M 4  | M 5  |
| J 1 | 05.0    | 00.0 | 08.0 | 10.0 | 11.0 |
| J 2 | 00.0    | 06.0 | 15.0 | 00.0 | 03.0 |
| J 3 | 08.0    | 05.0 | 00.0 | 00.0 | 00.0 |
| J 4 | 00.0    | 06.0 | 04.0 | 02.0 | 07.0 |
| J 5 | 00.0    | 00.0 | 00.0 | 00.0 | 00.0 |

**Table 10** Matrix table after step 2

| Job | Machine |      |      |      |      |
|-----|---------|------|------|------|------|
|     | M 1     | M 2  | M 3  | M 4  | M 5  |
| J 1 | 05.0    | 00.0 | 08.0 | 10.0 | 11.0 |
| J 2 | 00.0    | 06.0 | 15.0 | 00.0 | 03.0 |
| J 3 | 08.0    | 05.0 | 00.0 | 00.0 | 00.0 |
| J 4 | 00.0    | 06.0 | 04.0 | 02.0 | 07.0 |
| J 5 | 00.0    | 00.0 | 00.0 | 00.0 | 00.0 |

**Table 11** Matrix table after 1st scanning

| Job | Machine |      |      |      |      |
|-----|---------|------|------|------|------|
|     | M 1     | M 2  | M 3  | M 4  | M 5  |
| J 1 | 05.0    | 00.0 | 08.0 | 10.0 | 11.0 |
| J 2 | 00.0    | 06.0 | 15.0 | 00.0 | 03.0 |
| J 3 | 08.0    | 05.0 | 00.0 | 00.0 | 00.0 |
| J 4 | 00.0    | 06.0 | 04.0 | 02.0 | 07.0 |
| J 5 | 00.0    | 00.0 | 00.0 | 00.0 | 00.0 |

Table 11 is the matrix after 1st scanning. The order of the matrix is 5, and the number of lines drawn to cover all the zeros is 5 (In this case, the matrix’s order is equal to the number of lines required to cover all zeros.). As a result, we may create an assignment using this table.

Table 12 is the final matrix table after completing all steps. Table 13 represents which machine is assigned for which job. Figure 2 shows the unbalanced assignment problem’s illustrated solution. Total cost = 08.0 + 09.0 + 04.0 + 06.0 + 00.0 = 27.0

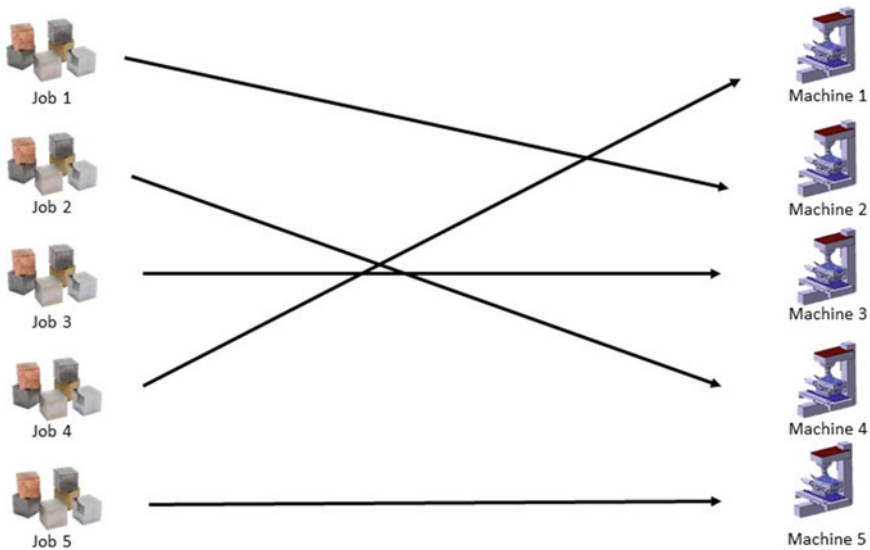
If “Number of lines drawn to cover all zeroes” ≠ “Order of matrix” after **step 2** of the unbalanced assignment problem, the following steps should be followed:

**Table 12** Final table matrix

| Job | Machine |      |      |      |      |
|-----|---------|------|------|------|------|
|     | M 1     | M 2  | M 3  | M 4  | M 5  |
| J 1 | 05.0    | 00.0 | 08.0 | 10.0 | 11.0 |
| J 2 | 00.0    | 06.0 | 15.0 | 00.0 | 03.0 |
| J 3 | 08.0    | 05.0 | 00.0 | 00.0 | 00.0 |
| J 4 | 00.0    | 06.0 | 04.0 | 02.0 | 07.0 |
| J 5 | 00.0    | 00.0 | 00.0 | 00.0 | 00.0 |

**Table 13** Result of the Unbalanced Assignment Problem

| Job | Machine | Cost |
|-----|---------|------|
| 1   | 2       | 08.0 |
| 2   | 4       | 09.0 |
| 3   | 3       | 04.0 |
| 4   | 1       | 06.0 |
| 5   | 5       | 00.0 |



**Fig. 2** Illustrates the visual representation of the solution to the unbalanced assignment problem

**Step 1** When rows are smaller than columns, the matrix is not square. The “Unbalanced Assignment Problem” will be converted to a “Balanced Assignment Problem” by adding a dummy row (job) with corresponding entities (cost) zero.

**Step 2** By taking the minimum element and subtracting it from all the other elements in each row. The matrix after subtraction is known as the row reduction matrix.

**Table 14** Initial table of an “Unbalanced Assignment Problem”

| Job | Machine |      |      |      |      |
|-----|---------|------|------|------|------|
|     | M 1     | M 2  | M 3  | M 4  | M 5  |
| J 1 | 04.0    | 03.0 | 06.0 | 02.0 | 07.0 |
| J 2 | 10.0    | 12.0 | 11.0 | 14.0 | 16.0 |
| J 3 | 04.0    | 03.0 | 02.0 | 01.0 | 05.0 |
| J 4 | 08.0    | 07.0 | 06.0 | 09.0 | 06.0 |

**Step 3** By taking the minimum element and subtracting it from all the other elements in each column. After subtraction, the matrix that is left is called the column reduction matrix.

**Step 4** Sketch the fewest horizontal and vertical lines ( $N$ ) that will cover all the zeros.

- An optimal solution can be found, if the number of lines ( $N$ ) is equal to the order of the matrix ( $n$ ).
- If  $N$  is less than  $n$ , then move on to Step 5.

**Step 5** Find the smallest element that is not covered ( $x$ ).

- Write uncovered element = uncovered value  $- x$
- Write intersection values = intersection values  $+ x$
- Write line values (other values) the same as before.  
Therefore, perform **step 4** again.

**Step 6** After finding the optimal matrix for the assignment, the subsequent steps should be performed.

- If any row has a single zero, then an assignment should be made and the corresponding column should be crossed out.
- Then, scan each column, and if any column has a single zero, make an assignment and cross out the corresponding column.
- If no single zero appears in any row or column, choose any zero at random to cross out the corresponding row and column of that zero.
- Finally, to finish the remaining assignment, repeat the row and column scanning process described earlier.

Another case study to illustrate the mentioned case (“Number of lines drawn to cover all zeroes”  $\neq$  “Order of matrix” after **step 2** of the unbalanced assignment problem) is presented below:

The smallest uncovered element  $x = 1$ .

$N = 5$  lines are needed to cover all 0s, and  $n = 5$  is the order of the matrix.

So,  $N = n$ . This matrix, then, represents the optimal set of solutions.

Tables 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, and 24 present the mentioned case study. The total cost =  $03.0 + 10.0 + 01.0 + 06.0 + 00.0 = 20.0$

**Table 15** The new matrix after adding a dummy row (step 1)

| Job | Machine |      |      |      |      |
|-----|---------|------|------|------|------|
|     | M 1     | M 2  | M 3  | M 4  | M 5  |
| J 1 | 04.0    | 03.0 | 06.0 | 02.0 | 07.0 |
| J 2 | 10.0    | 12.0 | 11.0 | 14.0 | 16.0 |
| J 3 | 04.0    | 03.0 | 02.0 | 01.0 | 05.0 |
| J 4 | 08.0    | 07.0 | 06.0 | 09.0 | 06.0 |
| J 5 | 00.0    | 00.0 | 00.0 | 00.0 | 00.0 |

**Table 16** Minimum element of rows

| Job | Machine |      |      |      |      |
|-----|---------|------|------|------|------|
|     | M 1     | M 2  | M 3  | M 4  | M 5  |
| J 1 | 04.0    | 03.0 | 06.0 | 02.0 | 07.0 |
| J 2 | 10.0    | 12.0 | 11.0 | 14.0 | 16.0 |
| J 3 | 04.0    | 03.0 | 02.0 | 01.0 | 05.0 |
| J 4 | 08.0    | 07.0 | 06.0 | 09.0 | 06.0 |
| J 5 | 00.0    | 00.0 | 00.0 | 00.0 | 00.0 |

**Table 17** Row reduction matrix (step 2)

| Job | Machine |      |      |      |      |
|-----|---------|------|------|------|------|
|     | M 1     | M 2  | M 3  | M 4  | M 5  |
| J 1 | 02.0    | 01.0 | 04.0 | 00.0 | 05.0 |
| J 2 | 00.0    | 02.0 | 01.0 | 04.0 | 06.0 |
| J 3 | 03.0    | 02.0 | 01.0 | 00.0 | 04.0 |
| J 4 | 02.0    | 01.0 | 00.0 | 03.0 | 00.0 |
| J 5 | 00.0    | 00.0 | 00.0 | 00.0 | 00.0 |

**Table 18** Minimum element of columns

| Job | Machine |      |      |      |      |
|-----|---------|------|------|------|------|
|     | M 1     | M 2  | M 3  | M 4  | M 5  |
| J 1 | 02.0    | 01.0 | 04.0 | 00.0 | 05.0 |
| J 2 | 00.0    | 02.0 | 01.0 | 04.0 | 06.0 |
| J 3 | 03.0    | 02.0 | 01.0 | 00.0 | 04.0 |
| J 4 | 02.0    | 01.0 | 00.0 | 03.0 | 00.0 |
| J 5 | 00.0    | 00.0 | 00.0 | 00.0 | 00.0 |

**Table 19** Column reduction matrix (step 3)

| Job | Machine |      |      |      |      |
|-----|---------|------|------|------|------|
|     | M 1     | M 2  | M 3  | M 4  | M 5  |
| J 1 | 02.0    | 01.0 | 04.0 | 00.0 | 05.0 |
| J 2 | 00.0    | 02.0 | 01.0 | 04.0 | 06.0 |
| J 3 | 03.0    | 02.0 | 01.0 | 00.0 | 04.0 |
| J 4 | 02.0    | 01.0 | 00.0 | 03.0 | 00.0 |
| J 5 | 00.0    | 00.0 | 00.0 | 00.0 | 00.0 |

**Table 20** Matrix table after 1st scanning (checking step 4)

| Job | Machine |      |      |      |      |
|-----|---------|------|------|------|------|
|     | M 1     | M 2  | M 3  | M 4  | M 5  |
| J 1 | 02.0    | 01.0 | 04.0 | 00.0 | 05.0 |
| J 2 | 00.0    | 02.0 | 01.0 | 04.0 | 06.0 |
| J 3 | 03.0    | 02.0 | 01.0 | 00.0 | 4.00 |
| J 4 | 02.0    | 01.0 | 00.0 | 03.0 | 00.0 |
| J 5 | 00.0    | 00.0 | 00.0 | 00.0 | 00.0 |

**Table 21** Matrix table after performing step 5

| Job | Machine |      |      |      |      |
|-----|---------|------|------|------|------|
|     | M 1     | M 2  | M 3  | M 4  | M 5  |
| J 1 | 01.0    | 00.0 | 03.0 | 00.0 | 04.0 |
| J 2 | 00.0    | 02.0 | 01.0 | 05.0 | 06.0 |
| J 3 | 02.0    | 01.0 | 00.0 | 00.0 | 03.0 |
| J 4 | 02.0    | 01.0 | 00.0 | 04.0 | 00.0 |
| J 5 | 00.0    | 00.0 | 00.0 | 01.0 | 00.0 |

**Table 22** Matrix table after 2nd scanning (step 6)

| Job | Machine |      |      |      |      |
|-----|---------|------|------|------|------|
|     | M 1     | M 2  | M 3  | M 4  | M 5  |
| J 1 | 01.0    | 00.0 | 03.0 | 00.0 | 04.0 |
| J 2 | 00.0    | 02.0 | 01.0 | 05.0 | 06.0 |
| J 3 | 02.0    | 01.0 | 00.0 | 00.0 | 03.0 |
| J 4 | 02.0    | 01.0 | 00.0 | 04.0 | 00.0 |
| J 5 | 00.0    | 00.0 | 00.0 | 01.0 | 00.0 |

**Table 23** Final assignment matrix table

| Job | Machine |      |      |      |      |
|-----|---------|------|------|------|------|
|     | M 1     | M 2  | M 3  | M 4  | M 5  |
| J 1 | 01.0    | 00.0 | 03.0 | 00.0 | 04.0 |
| J 2 | 00.0    | 02.0 | 01.0 | 05.0 | 06.0 |
| J 3 | 02.0    | 01.0 | 00.0 | 00.0 | 03.0 |
| J 4 | 02.0    | 01.0 | 00.0 | 04.0 | 00.0 |
| J 5 | 00.0    | 00.0 | 00.0 | 01.0 | 00.0 |

**Table 24** Result of the Unbalanced Assignment Problem

| Job | Machine | Cost |
|-----|---------|------|
| 1   | 2       | 03.0 |
| 2   | 1       | 10.0 |
| 3   | 4       | 01.0 |
| 4   | 3       | 06.0 |
| 5   | 5       | 00.0 |



## 4 Conclusion and Future Recommendations

This study considers both balanced and unbalanced assignment problems for assigning a task to a specific machine. The total cost of assigning those jobs is computed in both cases to determine which approach is more feasible. The balanced assignment problem divides five jobs among five different machines. Because the number of machines and jobs is equal, it was simple to assign five distinct jobs to five distinct machines. However, because there are more machines than jobs in the unbalanced problem, a dummy row for the machine is inserted at no cost. The “Hungarian Method” can be used to demonstrate that the one task assigned to machine 5 for the unbalanced problem has no cost. As a result, the unbalanced problem has a lower overall cost, whereas the balanced problem has a higher cost.

Furthermore, for both the balanced and unbalanced methods, Jobs 1, 3, and 4 are assigned to the same machines, namely, Machines 2, 3, and 1, respectively. Due to the false row, Job-2 and Job-5 have different machines assigned to both techniques. The second method is regarded as being more practical than the first since the overall cost in the unbalanced assignment problem is 27, which is much less than the total expenditure in the balanced assignment problem. Finding the best approach to an unbalanced assignment problem where the extra task can be delegated to a real machine necessitates utilizing a different technique than the “Hungarian Method” presented by the solved problem. However, one machine will be idle because of adding a dummy job. Therefore, an improved method for resolving unbalanced problems without converting them to balanced problems that can be solved without the use of dummies or pseudo-jobs/machines should be developed.

Nonetheless, the Multi-Objective Grey Wolf Optimization (MOGWO) algorithm and the Greedy Algorithm can be explored in any complex scenario of assignment problems in future studies. Another practical approach is possible that disregards the constraint of only one machine being able to do one job at a time. In this case, a technologically improved machine capable of handling multiple jobs at once will be introduced as a replacement for one of the existing machines.

## References

1. Z. Xiang, J. Yang, X. Liang, M.H. Naseem, Application of discrete Grey Wolf Algorithm in balanced transport problem, in *2021 3rd International Academic Exchange Conference on Science and Technology Innovation, IAECST 2021*, (2021), pp. 1312–1318. <https://doi.org/10.1109/IAECST54258.2021.9695827>
2. C. Woodyard, *New York City Is Costliest Place to Park in USA* (2018). <https://content.usatoday.com/communities/driveon/post/2011/07/new-york-city-costliest-place-to-park-your-car/1#.WWUoFoQrJdg>. Accessed 23 Apr 2022
3. K. McCoy, Drivers spend an average of 17 hours a year searching for parking spots. USA Today (2017). <https://www.usatoday.com/story/money/2017/07/12/parking-pain-causes-financial-and-personal-strain/467637001/>. Accessed 23 Apr 2022

4. W. Ho, P. Ji, A genetic algorithm for the generalised transportation problem. *Int. J. Comput. Appl. Technol.* **22**(4), 190–197 (2005). <https://doi.org/10.1504/IJCAT.2005.006959>
5. Z. Nakat, S. Herrera, Y. Cherkaoui, *Cairo Traffic Congestion Study* (World Bank, Washington, DC, 2013)
6. S. Bussmann, K. Schild, An agent-based approach to the control of flexible production systems, in *IEEE International Conference on Emerging Technologies and Factory Automation, ETFA*, vol. 2, (2001), pp. 481–488. <https://doi.org/10.1109/etfa.2001.997722>
7. S. Emde, M. Gendreau, Scheduling in-house transport vehicles to feed parts to automotive assembly lines. *Eur. J. Oper. Res.* **260**(1), 255–267 (2017). <https://doi.org/10.1016/j.ejor.2016.12.012>
8. S. Chopra, G. Notarstefano, M. Rice, M. Egerstedt, A distributed version of the Hungarian method for multirobot assignment. *IEEE Trans. Robot.* **33**(4), 932–947 (2017). <https://doi.org/10.1109/TRO.2017.2693377>
9. H.A. Hussein, M.A.K. Shiker, Two new effective methods to find the optimal solution for the assignment problems. *J. Adv. Res. Dyn. Control Syst.* **12**(7), 49–54 (2020). <https://doi.org/10.5373/JARDCS/V12I7/20201983>
10. M. Chen, D. Zhu, A workload balanced algorithm for task assignment and path planning of inhomogeneous autonomous underwater vehicle system. *IEEE Trans. Cogn. Develop. Syst.* **11**(4), 483–493 (2018)
11. C. Cubukcuoglu, P. Nourian, M.F. Tasgetiren, I.S. Sariyildiz, S. Azadi, Hospital layout design renovation as a quadratic assignment problem with geodesic distances. *J. Build. Eng.* **44**, 102952 (2021). <https://doi.org/10.1016/j.jobe.2021.102952>
12. U. Tosun, A new tool for automated transformation of quadratic assignment problem instances to quadratic unconstrained binary optimisation models. *Expert Syst. Appl.* **201**, 116953 (2022). <https://doi.org/10.1016/j.eswa.2022.116953>
13. S.M. Homayouni, D.B.M.M. Fontes, Production and transport scheduling in flexible job shop manufacturing systems. *J. Glob. Optim.* **79**(2), 463–502 (2021). <https://doi.org/10.1007/s10898-021-00992-6>
14. R. Wang, J. Yan, X. Yang, Neural graph matching network: Learning Lawler’s quadratic assignment problem with extension to hypergraph and multiple-graph matching. *IEEE Trans. Pattern Anal. Mach. Intell.* **44**(9), 5261–5279 (2022). <https://doi.org/10.1109/TPAMI.2021.3078053>
15. T. Dokeroglu, E. Sevinc, A. Cosar, Artificial bee colony optimization for the quadratic assignment problem. *Appl. Soft Comput. J.* **76**, 595–606 (2019). <https://doi.org/10.1016/j.asoc.2019.01.001>
16. X. Xiang, C. Liu, An almost robust optimization model for integrated berth allocation and quay crane assignment problem. *Omega (United Kingdom)* **104**, 102455 (2021). <https://doi.org/10.1016/j.omega.2021.102455>
17. Ö. Karsu, M. Azizoğlu, K. Alanlı, Exact and heuristic solution approaches for the airport gate assignment problem. *Omega (United Kingdom)* **103**, 102422 (2021). <https://doi.org/10.1016/j.omega.2021.102422>
18. A.S. Hameed, M.L. Mutar, H.M.B. Alrikabi, Z.H. Ahmed, A.A. Abdul-Razaq, H.K. Nasser, A hybrid method integrating a discrete differential evolution algorithm with tabu search algorithm for the quadratic assignment problem: A new approach for locating hospital departments. *Math. Probl. Eng.* **2021** (2021). <https://doi.org/10.1155/2021/6653056>
19. S.T. Ngo, J. Jaafar, I.A. Aziz, B.N. Anh, A compromise programming for multi-objective task assignment problem. *Computers* **10**(2), 1–16 (2021). <https://doi.org/10.3390/computers10020015>
20. X. Zheng, D. Zhou, N. Li, T. Wu, Y. Lei, J. Shi, Self-adaptive multi-task differential evolution optimization: With case studies in weapon–target assignment problem. *Electronics* **10**(23), 2945 (2021). <https://doi.org/10.3390/electronics10232945>
21. X. Hu, C. Liang, D. Chang, Y. Zhang, Container storage space assignment problem in two terminals with the consideration of yard sharing. *Adv. Eng. Inform.* **47**, 101224 (2021). <https://doi.org/10.1016/j.aei.2020.101224>

22. Q. Rabbani, A. Khan, A. Quddoos, Modified Hungarian method for unbalanced assignment problem with multiple jobs. *Appl. Math. Comput.* **361**, 493–498 (2019). <https://doi.org/10.1016/j.amc.2019.05.041>
23. A. Kumar, A modified method for solving the unbalanced assignment problems. *Appl. Math. Comput.* **176**(1), 76–82 (2006). <https://doi.org/10.1016/j.amc.2005.09.056>
24. A. Iampang, V. Boonjing, P. Chanvarasuth, A cost and space efficient method for unbalanced assignment problems, in *IEEM2010 – IEEE International Conference on Industrial Engineering and Engineering Management*, (2010), pp. 985–988. <https://doi.org/10.1109/IEEM.2010.5674228>
25. L. Wang, Z. He, C. Liu, Q. Chen, Graph based twin cost matrices for unbalanced assignment problem with improved ant colony algorithm. *Results Appl. Math.* **12**, 100207 (2021). <https://doi.org/10.1016/j.rinam.2021.100207>

**Part IV**  
**Bigdata in Medical Applications**

# A Survey on Memory Assistive Technology for Elderly



N. Shikha and Antara Roy Choudhury

## 1 Introduction

Dementia, which usually occurs in older adults, refers to the progressive cognitive decline due to the death of brain cells, which indicates a growing need for guidance and assistance in carrying out regular activities. Since it can be fatal in its later phases, it causes a significant social and economic burden, as well as a serious impact on a person's personality and family. The number of people with Alzheimer's alone is predicted to hit 115 million by 2050. The very thought that somebody must be dependent on caretakers for performing tasks they were once good at in the past can affect them emotionally and mentally as there is always a risk of loneliness.

Reminiscence therapy (RT) can change the outlook of old people towards life since it has benefits on emotional and mental well-being, social relationships and interactions, and protecting personal identity. RT works by encouraging people to revisit moments from their past and is the process of collecting and recalling memories through pictures, stories, and other mementos [1]. It is likely to stimulate the hippocampus and prefrontal cortex, the parts of the brain that are responsible for long- and short-term memory, respectively. Lifelogging is a method of digitally recording a person's everyday activities in varied degrees of detail. Recently, Wearable cameras are used to automatically capture moments that can be reviewed later. This can help in reminiscence therapy for elderly.

There is no effective remedy for dementia till now, but with the advancements in technology, artificial intelligence (AI)-based solutions can help improve one's qual-

---

N. Shikha (✉) · A. R. Choudhury  
Department of Computer Science and Engineering, B.M.S. College of Engineering, Bangalore,  
India  
e-mail: [antararc.cse@bmsce.ac.in](mailto:antararc.cse@bmsce.ac.in)

ity of living. One such solution is Life Bio Memory [2], a speech automated platform which helps in reminiscence and stores user's stories, photos, and videos using which summarized life stories are generated by keyword and phrase extraction. Also, as the experimentation conducted in [3] shows, using virtual reality saw three and two times more improvement in cognitive and execution skills, respectively, compared to pen-and-paper-based approaches in elderly persons. In addition, the results in [4] suggest that dialogue agents can elicit more information and provide valuable support to older adults when the conversation topics are more personal, such as life objectives and the obstacles they faced in the process of growing old and as the conversation progresses, people tend to provide longer responses. The various approaches in the development of intelligent chatbots [5] are Semantic Parsing, Pattern Matching, AI Mark-up Language, Chat Script, Domain Ontologies, Markov Chain probabilistic model, and Neural Networks (Recurrent neural networks, long short-term memory, sequence-to-sequence).

Our motivation is to promote healthy aging by helping the aged have carefree conversations and pursue their hobbies and live an independent life. The proposed companion will be a combination of modified reminiscence therapy and visual lifelogging to aid the elderly in recalling memories whenever necessary. Along with safe navigation to avoid wandering.

This paper is organized into sections as follows: Sect. 2 for previous related work, Sect. 3 for proposed design, and Sect. 4 for conclusion and future work.

## 2 Literature Survey

This section summarizes assistive technologies catered to elderly people, categorized into conversational agents used for reminiscence in Table 1, the Internet of Things (IoT) in Table 2, virtual, augmented reality (VR, AR) systems in Table 3, and finally navigation-based systems.

For lifelogging, which is fundamental for capturing experiences, a novel wearable emotion-based lifelogging system, Memento is proposed in [6] which uses: Electroencephalogram (EEG), motion sensor data and Global Positioning System (GPS) to detect emotions via signal processing (band pass filter, non-brain activity removal) and is integrated with a smart glass. The lifelogging engine auto selects the method: audio, image, or video. For emotion recognition, EEG segmentation and fractal dimension-based feature extraction is applied. The intermediate results of the feature extractor are tagged and stored in SQLite database which acts as a lifelog collector. Furthermore, in [7], classification of informative lifelogs is done based on cities using K means or DB-scan algorithm, and the domains are converted to digraphs, using which stories are generated based on templates defined for each domain/cluster.

*Location Technology* can be utilized to provide security, provide independence to the elderly, and reduce stress and burden on the caretakers [26].

**Table 1** Conversational agents

| Purpose   | Methodology   | Notes  |
|---|---|--|
| [8] Voice automated, news entertainment chatbot to monitor cognitive impairment | Questions are generated from news items. Used similarity metric and binary classification model trained using textual analysis features for detection of cognitive impairment   | Decision tree algorithm: Accuracy, F1 score and recall >83%  |
| [9] Care chatbot  | Questions based on selected biographical topic and provision to provide descriptive and multiple-choice answer  | Limitations:<br>Does not use natural language processing techniques (NLP)  |
| [10] Humanoid robot for memory exercise   | Robot performed tasks like asking questions post reading stories, song matching and helped recall associated/non associated words<br>Visual attention detection based on face expression and head pose estimation   | Achieved improved memory and reduced level of anxiety during interaction   |
| [11] Rule-based virtual caregiver system  | Question (7 predefined), answer (choice + descriptive)-based mind sensing technique to store inner emotions, thoughts. The score for each Q-A pair is calculated using: $Stotal = (SAnswer + SObservation + SSentiment)/3$<br>Weekly feedback notification generated contains, greeting + reflection + advice + conclusion for the worst Stotal in the week   | LINE chatbot used to assess physical, mental, and social well-being of the elderly and provide graph analytics   |
| [12] Charlie, personality-based chatbot   | Rule-based bot and responds by detecting the intent of latest user utterance using Dialog Flow  | Functionalities:<br>Setting and receiving reminders, recommend healthy tips and anecdotes to trigger compassion, discuss and ask questions on selected topics for entertainment, quiz, and riddles |
| [13] Chatbot to aid in medicine intake  | Pre-processing using MATLAB Text analysis toolbox: Drug package insert is segmented into paragraphs, tokenized, stop words, and punctuation is removed, normalized, and converted to word embeddings<br>Word size and frequency is used to select relevant words, distance based on their relative positions and frequency of occurring together is used to form words clusters<br>Unsupervised, Subtractive Mountain cluster algorithm applied to associate keywords in query with clusters and provide predefined responses | Information used by chatbot to give responses:<br>Doctor prescription, medication package inserts, color of box, and pill  |

(continued)

Table 1 (continued)

| Purpose   | Methodology   | Notes  |
|---|---|--|
| [14] Remi: To reduce the effect of cognitive decline and memory | Has four modules:<br>1. Chatbot: built using Dialog Flow ES to ask predefined questions on elderly's past, present<br>2. Shuffle memory game, themed quiz: answer intent detection and validation using dialog-flowttrter package<br>3. Stay informed: Google news API, search using date and keyword<br>4. Local data persistence: Sembast document-based dB of text, images, and location   | The question based chatbot can be extended to verify and provide feedback on the answers to improve user experience  |
| [15] AI-based reminiscence therapy                              | Storage: Materials on historic cultural heritage like dance, song, proverb, tongue-twister, patient information, user preference, and reactions collected from therapy<br>Face emotion recognition is based on eye and mouth features (smile index)<br>Optimal pathfinding reinforcement learning algorithm: Model learns from past reactions to improve user reminiscence every moment with reward/penalty based on emotion and updates Q-table  | Average scores (out of 10):<br>Usability – $7.86 \pm 1.98$ .<br>Satisfaction – $8.43 \pm 1.59$ .<br>Information – $6.04 \pm 0.93$ .<br>Interface – $6.22 \pm 0.87$ .                                   |
| [16] Calendar assistant   | Understands speech commands to add, modify, delete, and reminds a calendar event. Chatbot to validate, clarify, and give appropriate response if event does not exist using APIAI   | Provides efficient querying and sync flexibility with google calendar  |
| [17] Zenbo, photo reminiscence and voice automated robot        | Extraction of event, scene from the selected image using VGG16 model trained on USED, Places365 dataset.<br>The metadata for the selected picture is used to pick questions from pool of 406 questions. Keyword matching is applied on yes/no replies user responses<br>Dynamic and flexible interactive PhotoShow (D-FLIP) platform displays similar images based on time, place, or people  | Image understanding performance:<br>Events – Training: 88%, validation: 84%, soft test set: 82%, hard test set: 59%<br>Scenes – Training: 80%, validation: 80%, soft test set: 76%, hard test set: 60% |
| [18] Pepper: Episodic memory assistance robot                   | The system observes daily events and extracts people, activities, times, places, and objects (called categories) from each episode. Stores them in a dynamic graph dB allowing merging and clustering. Keyword extraction applied on query to extract categories and Naïve Bayes algorithm used to match the query categories and cluster. Based on the cluster/event cues are generated to trigger memory<br>Reinforcement learning is used to efficiently learn the best mapping of cue and event type for a particular user using feedback | Memory assistance confidence: 90% and 19.63% improvement in recalling  |



**Table 2** The Internet-of-Things-based systems

| Purpose   | Methodology   |
|---|---|
| [19] Low-cost smart medicine box                            | Separate boxes for disease specific medicines<br>The setup mainly consists of node-microcontroller unit controlled by a webpage sending triggers, light dependent resistor sensors inside boxes to detect medicine intake, motor to open and close the lid after 30 s timer, magnet reed sensor that activates the motor if it's out of magnetic field for security, and buzzer for reminder. The data is pushed to a web server using Wi-Fi    |
| [20] Low cost and power, wearable device for fall detection | Monitors body temperature, heart rate using peak detection algorithm and oxygen saturation during daily activities to detect fall with 98% accuracy   |
| [21] Smart mirror for elderly mental well-being             | The emotion (CNN model trained on FER2013 dataset), and voice (MLP model trained on RAVDESS dataset) of the elderly is monitored and the mirror/chatbot greets, asks simple questions when a lonely or depressed feeling is detected  |
| [22] 3D character model chatbot for elderly                 | Hardware consists of two SG90 Servos for speaking and neck rotation (step) to track user position and microcontroller unit, smartphone contains Text-to-Speech and Speech-to-Text (android function in Unity) and fetches data from Google app engine. The user input undergoes semantic analysis using NLTK library and naive Bayes algorithm (60% accuracy) is used to predict the response. The client chatbot undergoes continuous training |

Reference [27] discusses methods for live tracking, fall detection, and early warnings. This study utilizes IoT devices, GPS, and Bluetooth based on signal loss in the receiver from the transmitter [28] to track the old person's location and caretakers can check the patient status and get alerts. The movement of patients can be virtually restricted by using Geofencing technology to create boundaries around the patient's region, crossing which the contacts will receive a notification, preventing them from getting lost.

To prevent wandering: Geofencing area analysis, Haversine formula calculations, and Firebase cloud messaging (FCM) server-based notification system is proposed in [29]. The Haversine method is used to calculate the relative distance between two objects based on latitude and longitude, for example, finding individual distance between the patient and each of the caretakers.

Reference [30] discusses a Navigation based approach where GPS is used to learn the standard outdoor routes taken by the elderly (is an overhead) and is used to predict likely destinations. In case a navigational error is detected, it alerts the person with a sound, and then re-routes him to the right path. Combination of Ultra-Wideband for short range, large data radio transmission and iBeacon proximity sensors for indoor navigation and commercial systems like Bluewater Security (wrist worn), smart sole, pocket finder was suggested for wandering detection.

**Table 3** Augmented/virtual reality systems

| Purpose   | Methodology   |
|---|---|
| [23] Virtual reality application to improve memory                              | Consists of following built using Unity 3D:<br>Environment to perform repetitive daily chores, professional tasks, relative name, kinship recognition (based on labelled database mapping), and weekend games (like tennis, dancing) with scores<br>Three-layer development: Creating 3D objects in computer-aided design software, grouping and pivoting, creating scenes, layout, texturization, and scripting process (to control objects) along with evaluation and help module   |
| [24] Bidirectional projection-based augmented reality system with deep learning | Projection of App UI: 3D space map reconstruction, optimal plane selection: Plane surface close to the user coordinates is selected<br>Scenarios: Monitoring, smart IoT intercom, daily medication alarm, weather condition, and home control<br>Contextual awareness:<br>(1) Object detection (YOLO v3 model trained on MS COCO, Open Image dataset) for localization of medicine bottle, window, door<br>(2) Face recognition (FaceNet model) to classify visitors into family, undefined, unknown<br>(3) Pose estimation (PoseNet model) to detect user state (sit, stand, hide, lie)<br>Performance:<br>(1) Pose estimation Avg precision – stand: 87%, sit: 87.5%, hide: 93%, lie: only upper body joints detected<br>(2) Face recognition detection rate –<br>Known (30 images): 87.14% for $\pm 90$ -degree angle, 95.22% with glasses, 49% if half covered<br>Undefined (15 images): 75% and 36% when occluded, unknown: 96% precision<br>(3) Object detection – detection rate decreased with increase in distance for medicine, precision increased when distance $> 2$ m and $> 5$ m for window, door respectively |
| [25] Virtual reminiscence therapy system  | Software: After effects, Photoshop, 3ds Max, Fader VR for scene switching Vuforia, Maker, and Unity engine<br>Historic items, songs, stories, and interior of houses recreated using the VR headset with audio to trigger memories<br>AR is used as a navigation interface to preview scenes based on the image captured by the user  |

## 2.1 Analysis

It is evident that most of the existing dialog systems developed specifically for elderly focus on single modality reminiscence therapy either visual or textual or are only for entertainment purposes. Moreover, automated reminiscence therapy is mostly collecting the information or lifelogs and using it for generating questions sometimes without giving feedback on the responses. This can only help trigger certain pieces of memory at a time. Hence there is a dearth of personalized systems for elderly, which combines entertainment and triggering, recalling of specific memories or experiences (both visual and textual) instantaneously to uplift their

mood. For example, suggesting topics to start a conversation based on the situation and finding missing piece of information while conversing with peers. Even some of the best performing systems use multiple AI models, making management and integration with mobile application cumbersome. In a Geofencing-based approach, the system detects wandering when the patients go beyond some defined safe zones which indirectly restricts the independence of the aged person.

### 3 Proposed Design

In order to promote a happy and holistic well-being for the elderly, the future implementation of the proposed system (see Fig. 1) will be a mobile application and be integrated with mainly three modules namely chatbot which provides both textual and image-based responses, lifelogging or memory collector, and safe navigation and will include the following features to assist the aged:

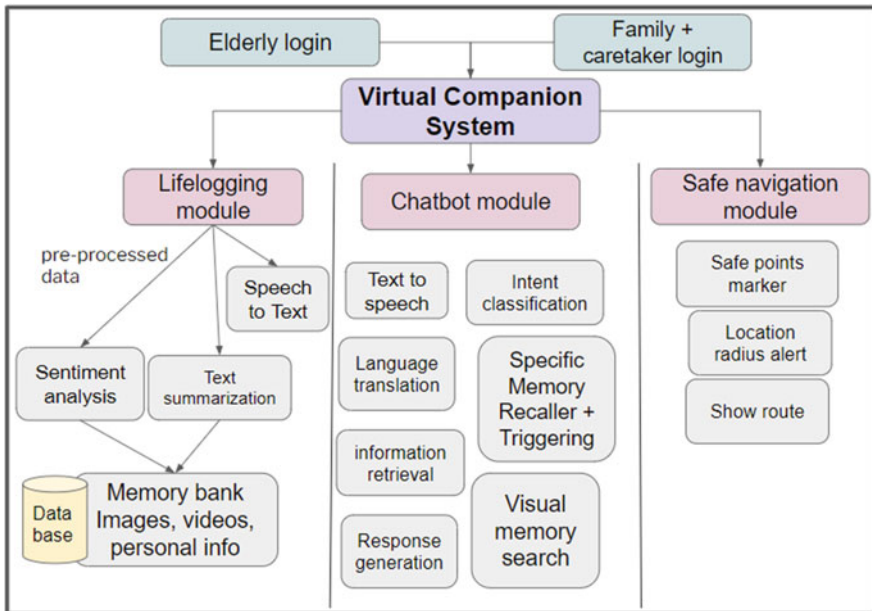


Fig. 1 High level design of the virtual companion system

- Provision to converse in the user's native language. By including native language speech to English text conversion in the Natural language understanding module of the chatbot.
- Dynamic addition and retrieval of daily experiences and important facts in the form of captioned visuals or utterances to the knowledge base by family and elderly:

Important, personal details, memories and conversations with doctor, friend or relatives can be recorded and post extraction of semantic, lexical, and syntactic features can be stored in a NoSQL database like firebase or graph DB accordingly for easy retrieval. In addition, the experiences marked as important can be reviewed on daily basis to enhance memory or to avoid performing daily tasks, repetitively.

- Modified Reminiscence, using which the old and fresh memories can be recalled when required and memory triggering:

A chatbot trained using Generative Pre-trained Transformer (GPT) model that tries to bring up relevant memory as part of the conversation to aid in triggering and answer's questions in a humanly manner by accessing right cues from the memory bank rather than giving exact retrieved responses can be implemented.

- Display related images or videos in case of Visual memory search to trigger memories: As purely image-based search requires domain specific annotated dataset to train; the image/video lifelogs can be retrieved based on the tags and description added during data storage. Even if the visuals retrieved are not accurate, it will aid in triggering other similar memories.
- Provide Personal touch by analyzing sentiments in speech and bringing up mood specific moments from the memory bank and remind happy forgotten memories frequently to provide entertainment through their own memory.
- Intent classification enables having a single chat screen for both visual and textual recalling. The model can be trained to perform a binary classification to identify if the user intent is visual or textual memory retrieval based on certain keywords in the query.
- Safe navigation (see Fig. 2). with live location, deviation tracking and alerting to family and caretakers:

Frequently visited locations and known residences called safe points are marked. The system will prompt the elderly to lock the destination (a subset of safe points) before leaving the house. If not done, an arbitrary radius of 10 m is considered around which the destination needs to be selected. Predefined routes added by family members or shortest route shown through google maps can be chosen if the destination is closer. If public transport is the mode of transport, then regular alerts are sent to the elderly and family to ensure proper deboarding. This will enforce outdoor safety and the elderly can be reminded in case of wandering without compromising on independence.

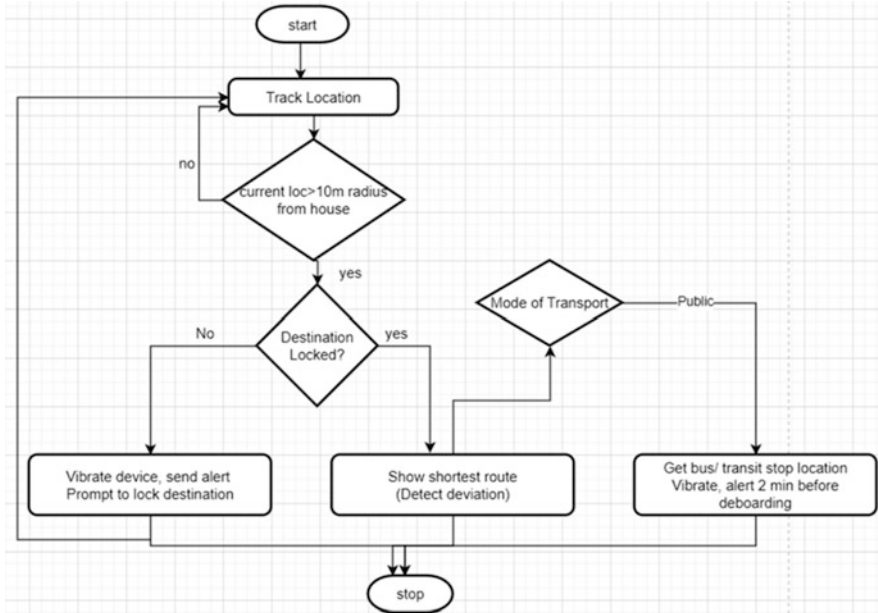


Fig. 2 Flowchart of safe navigation module

## 4 Conclusion

The objective of our research is to provide comprehensive support to the aged so that they are not detached from the world around them. After a detailed survey of different systems for the elderly, comprising of AR, VR, IOT, AI, and navigation technologies, and understanding the various techniques and systems, it can be noted that there were many variants of chatbots, starting from the ones not using NLP, focusing on a single use case like memory games, calendar reminders, simple rule-based bots achieving significant results, and then the advanced ones. The implementation of the proposed design will be an application supporting multimodality (textual and visual format) reminiscence through triggering and instantaneous memory recalling by means of a chatbot along with safe navigation system to provide outdoor independence. However, the key challenges for building the memory recaller are firstly, integrating multiple deep learning models in a mobile application and secondly, processing personalized sparse data in image, video, audio, or textual format and grouping, linking them in such a way that it is retrievable quickly based on the user query.

## References

1. GoodTherapy. Reminiscence therapy, <https://www.goodtherapy.org/learn-about-therapy/types/reminiscence-therapy>. Last accessed 25 Oct 2022
2. B. Sanders, B. Williams, LifeBio memory's technology advancements for capturing life stories and seeing the whole person. *Alzheimers Dement.* **17**, e052237 (2021)
3. P. Gamito, J. Oliveira, C. Alves, N. Santos, C. Coelho, R. Brito, Virtual reality-based cognitive stimulation to improve cognitive functioning in community elderly: A controlled study. *Cyberpsychol. Behav. Soc. Netw.* **23**(3), 150–156 (2020)
4. S.Z. Razavi, L.K. Schubert, K. van Orden, M.R. Ali, B. Kane, E. Hoque, Discourse behavior of older adults interacting with a dialogue agent competent in multiple topics. *ACM Trans. Interact. Intell. Syst.* **12**(2), 1–21 (2022)
5. S. Hussain, O. Ameri Sianaki, N. Ababneh, A survey on conversational agents/chatbots classification and design techniques, in *Web, Artificial Intelligence and Network Applications 2019, Advances in Intelligent Systems and Computing*, ed. by L. Barolli, M. Takizawa, F. Xhafa, T. Enokido, vol. 927, (Springer, Cham, 2019), pp. 946–956
6. S. Jiang, Z. Li, P. Zhou, M. Li, Memento: An emotion-driven lifelogging system with wearables. *ACM Trans. Sens. Netw.* **15**(1), 1–23 (2019)
7. G. Liu, M.U. Rehman, Y. Wu, Toward storytelling from personal informative lifelogging. *Multimed. Tools Appl.* **80**(13), 19649–19673 (2021)
8. F. de Arriba-Pérez, S. García-Méndez, F.J. González-Castaño, E. Costa-Montenegro, Automatic detection of cognitive impairment in elderly people using an entertainment chatbot with Natural Language Processing capabilities. *J. Ambient Intell. Humaniz. Comput.*, 1–16 (2022). <https://doi.org/10.1007/s12652-022-03849-2>
9. C. Müller, R. Paluch, A.A. Hasanat, Care: A chatbot for dementia care, in *Mensch und Computer 2022. Workshop-Band*, ed. by K. Marky, U. Grünefeld, T. Kosch, (Gesellschaft für Informatik e.V., Bonn, 2022)
10. O. Pino, G. Palestra, R. Trevino, B. De Carolis, The humanoid robot NAO as a trainer in a memory program for elderly people with mild cognitive impairment. *Int. J. Soc. Robot.* **12**(1), 21–33 (2020)
11. C. Miura, S. Chen, S. Saiki, M. Nakamura, K. Yasuda, Assisting personalized healthcare of elderly people: Developing a rule-based virtual caregiver system using mobile chatbot. *Sensors* **22**(10), 3829 (2022)
12. S. Valtolina, L. Hu, Charlie: A chatbot to improve the elderly quality of life and to make them more active to fight their sense of loneliness, in *14th Biannual Conference of the Italian SIGCHI Chapter*, (Association for Computing Machinery, New York, 2021), pp. 1–5
13. N. Clar, P.A. Salgado, T.P. Perdicoulis, Subtractive mountain clustering algorithm applied to a chatbot to assist elderly people in medication intake. arXiv preprint arXiv:2110.00933 (2021)
14. H. Nóbrega Grigolli, K. Takashi Yoshida, R. De Oliveira Rocha, C. Amato, V. Farinazzo Martins, REMI: Working on the memory and logical reasoning of the elderly, in *22nd International Conference on Human Computer Interaction*, (Association for Computing Machinery, New York, 2022), pp. 1–4
15. À. Nebot, S. Domènech, N. Albino-Pires, F. Mugica, A. Benali, X. Porta, O. Nebot, P.M. Santos, LONG-REMI: An AI-based technological application to promote healthy mental longevity grounded in reminiscence therapy. *Int. J. Environ. Res. Public Health* **19**(10), 5997 (2022)
16. L. Ferland, Z. Li, S. Sukhani, J. Zheng, L. Zhao, M. Gini, Assistive AI for coping with memory loss, in *32nd AAAI Conference on Artificial Intelligence*, (AAAI, 2018), pp. 1–4
17. E. Gamborino, A. Herrera Ruiz, J.F. Wang, T.Y. Tseng, S.L. Yeh, L.C. Fu, Towards effective robot-assisted photo reminiscence: Personalizing interactions through visual understanding and inferring, in *International Conference on Human-Computer Interaction*, (Springer, Cham, 2021), pp. 335–349

18. C.Y. Yang, E. Gamborino, L.C. Fu, Y.L. Chang, A brain-inspired, self-organizing episodic memory model for a memory assistance robot. *IEEE Trans. Cogn. Dev. Syst.* **14**(2), 617–628 (2021)
19. P.H. Vardhini, M.S. Harsha, P.N. Sai, P. Srikanth, IoT based smart medicine assistive system for memory impairment patients, in *12th International Conference on Computational Intelligence and Communication Networks*, (IEEE, 2020), pp. 182–186
20. J.D.T. Retamosa, A. Araujo, Z.M. Wawrzyniak, Low power wearable device for elderly people monitoring, in *Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments*, vol. 10808, (SPIE, 2018), pp. 987–996
21. P. Silapa Suphakorn Wong, K. Uehira, Smart mirror for elderly emotion monitoring, in *3rd IEEE Global Conference on Life Sciences and Technologies*, (IEEE, 2021), pp. 356–359
22. W.D. Liu, K.Y. Chuang, K.Y. Chen, The design and implementation of a chatbot’s character for elderly care, in *International Conference on System Science and Engineering*, (IEEE, 2018), pp. 1–5
23. F.A. Chicaiza, L. Lema-Cerda, V. Marcelo Álvarez, V.H. Andaluz, J. Varela-Aldás, G. Palacios-Navarro, I. García-Magariño, Virtual reality-based memory assistant for the elderly, in *International Conference on Augmented Reality, Virtual Reality and Computer Graphics*, (Springer, Cham, 2018), pp. 269–284
24. Y.J. Park, H. Ro, N.K. Lee, T.D. Han, Deep-care: Projection-based home care augmented reality system with deep learning for elderly. *Appl. Sci.* **9**(18), 3897 (2019)
25. Y.C. Tsao, C.C. Shu, T.S. Lan, Development of a reminiscence therapy system for the elderly using the integration of virtual reality and augmented reality. *Sustainability* **11**(17), 4792 (2019)
26. S.D. Freiesleben, H. Megges, C. Herrmann, L. Wessel, O. Peters, Overcoming barriers to the adoption of locating technologies in dementia care: A multi-stakeholder focus group study. *BMC Geriatr.* **21**(1), 1–17 (2021)
27. C.W. Lee, H.M. Chuang, Design of a seniors and Alzheimer’s disease caring service platform. *BMC Med. Inform. Decis. Mak.* **21**(Suppl 10), 273 (2021)
28. W. Chantaweesomboon, Bluetooth geo-fence for elderly and patient care, in *25th International Computer Science and Engineering Conference*, (IEEE, 2021), pp. 252–255
29. E.R. Pratama, F. Renaldi, F.R. Umbara, E.C. Djamal, Geofencing technology in monitoring of geriatric patients suffering from dementia and Alzheimer, in *3rd International Conference on Computer and Informatics Engineering*, (IEEE, 2020), pp. 106–111
30. A. Hammoud, M. Deriaz, D. Konstantas, Wandering behaviors detection for dementia patients: A survey, in *3rd International Conference on Smart and Sustainable Technologies*, (IEEE, 2018), pp. 1–5

# An Experimental Investigation on the Emotion Recognition Using Power Spectrum Density and Machine Learning Algorithms in EEG Signals



Nirmal Varghese Babu and E. Grace Mary Kanaga

## 1 Introduction

### 1.1 Human Brain

The brain, a highly developed organ, regulates every bodily function, including intellect, memory, emotion, touch, motor skills, vision, respiration, temperature, and hunger. The central nervous system, sometimes known as the CNS, includes the spinal cord that extends from the brain. The brain, a remarkable three-pound organ, regulates all bodily processes, analyzes information from the outside world, and houses the intellect and soul. Memory, emotion, creativity, and intellect are just a few of the operations that the brain controls. The brain, which is shielded by the skull, is made up of the brainstem, cerebellum, and cerebrum. Each of our five senses—sight, smell, touch, taste, and hearing—receives information to the brain, usually all at once. By fusing the messages in a meaningful fashion, it aids in memory retention. The functioning of various body organs, as well as our thoughts, memories, and speech, is controlled by the brain. It also directs the motion of our arms and legs.

---

N. V. Babu (✉) · E. G. M. Kanaga

Department of Computer Science and Engineering, Karunya Institute of Technology and Sciences, Coimbatore, Tamil Nadu, India

e-mail: [nirmalvarghese@karunya.edu.in](mailto:nirmalvarghese@karunya.edu.in); [grace@karunya.edu](mailto:grace@karunya.edu)



## ***1.2 Emotions***

Emotions are mental states brought on by neurophysiological changes. They are associated in various ways with thoughts, feelings, behavioral responses, and a degree of pleasure or displeasure. In science, there is currently no accepted definition. There are several connections between emotions, mood, temperament, personality, disposition, and creativity. Emotion is a type of subjective mental state. Emotions can be a reaction to internal or external stimuli (such thoughts or memories). Emotions and moods are distinct from one another. A mood is a mental state that increases the likelihood that we will act in a certain way.

Emotions have a big impact on how we connect with others and make decisions, but they also have a big impact on how we see the world around us. Because of the scientific community's present interest in creating emotional links between humans and computers, it was necessary to determine the former's emotional state. Numerous evaluation methods, such as subjective self-reports, autonomic, and neurophysiological testing, may be used to accomplish this. Electroencephalography (EEG) has received a lot of interest from researchers lately since it can be utilized in an easy, economical, portable, and user-friendly way to identify emotions.

Emotion recognition is a method for identifying human emotions. People's ability to accurately predict others' emotions varies tremendously. Technology's application to help people recognize emotions is a relatively recent area of research. The technique often works best when it incorporates a variety of modalities into the setting. The most recent focus has been on automating the detection of physiology as measured by wearables, spoken expressions from audio, written expressions from text, and facial emotions from video.

## ***1.3 Electroencephalography Signals***

An electrogram of the electrical activity on the scalp is captured using the electroencephalography (EEG) method. This electrogram has been proven to accurately depict the macroscopic activity of the brain's surface layer. It is non-invasive since the electrodes are often inserted along the scalp. The EEG measures changes in neuronal voltage caused by ionic current in the brain. An EEG is a procedure used in medicine to continuously monitor electrical activity in the brain using a few scalp electrodes. Diagnostic software typically concentrates on the content of the EEG spectrum or potentials associated with events. The first looks at possible changes related to a certain event, such as "stimulus initiation" or "button push." In the latter, a variety of cerebral oscillations—often referred to as "brain waves"—that may be identified in the frequency range of EEG recordings are investigated. The effective technique of electroencephalography (EEG) permits the collecting of brain signals related to various states from the surface of the scalp. These signals are typically

divided into the five categories of delta, theta, alpha, beta, and gamma based on signal frequencies that vary from 0.1 to more than 100 Hz.

## 2 Previous Works

Chunmeni Qing et al. [1] suggested a new interpretable machine learning and EEG inputs, an approach taken to emotion recognition with an activation mechanism. It suggests using the emotional activation curve to illustrate how emotions are activated. The algorithm first extracts characteristics from EEG data and then classifies emotions using machine learning methods. Various trial portions are utilized to train the proposed model and evaluate its effects on the outcomes of emotion detection. The classification findings and two emotion coefficients, namely the correlation coefficients and entropy coefficients, are used to generate a unique activation curve [2] of emotions. The emotional activation process can be partially revealed by the activation curve, which also can categorize emotions. In [3], the characteristics that CNN has retrieved are initially transferred to SAE for encoding and decoding. Then, for a classification job, the data with less redundancy is utilized as the input characteristics of a DNN. Testing is conducted using the DEAP and SEED public databases. According to experimental findings, the suggested network performs emotion identification better than traditional CNN approaches. According to Prashant Lahane et al. [4], to identify the emotional state of the test participant, features from the EEG signals are extracted using the Kernel Density Estimation (KDE) method and then categorized using an artificial neural network classifier. Results are produced to demonstrate that the suggested improved KDE provides more accurate results. The notion of cluster kernels is also used in the proposed technique to provide superior subject emotion prediction from streaming EEG data.

In [5], Electroencephalography (EEG) data are used to create an emotion recognition system based on the valence/arousal paradigm. Spectral characteristics are derived from each frequency band after discrete wavelet transform (DWT) decomposes EEG signals into the gamma, beta, alpha, and theta frequency bands. By maintaining the same dimensionality as a transform, principle component analysis (PCA) is used on the extracted features to make the features uncorrelated with another machine that supports vector. Emotional states are categorized using SVM, K-nearest neighbor (KNN), and artificial neural networks (ANN). According to Ahmad Tauseef Sohaib et al. [6], five different machine learning methods were assessed for their ability to categorize EEG data linked to affective/emotional states. IAPS (International Affective Picture System) database images were used to elicit the individuals' feelings. After the artifacts in the raw EEG data were removed, a selection of features was made to feed into the classifiers. The findings demonstrated that it is challenging to train a classifier to be accurate over big datasets (15 subjects), but KNN and SVM with the suggested features were relatively accurate over smaller datasets (5 subjects), recognizing the emotional states with an accuracy up to 77.78%.

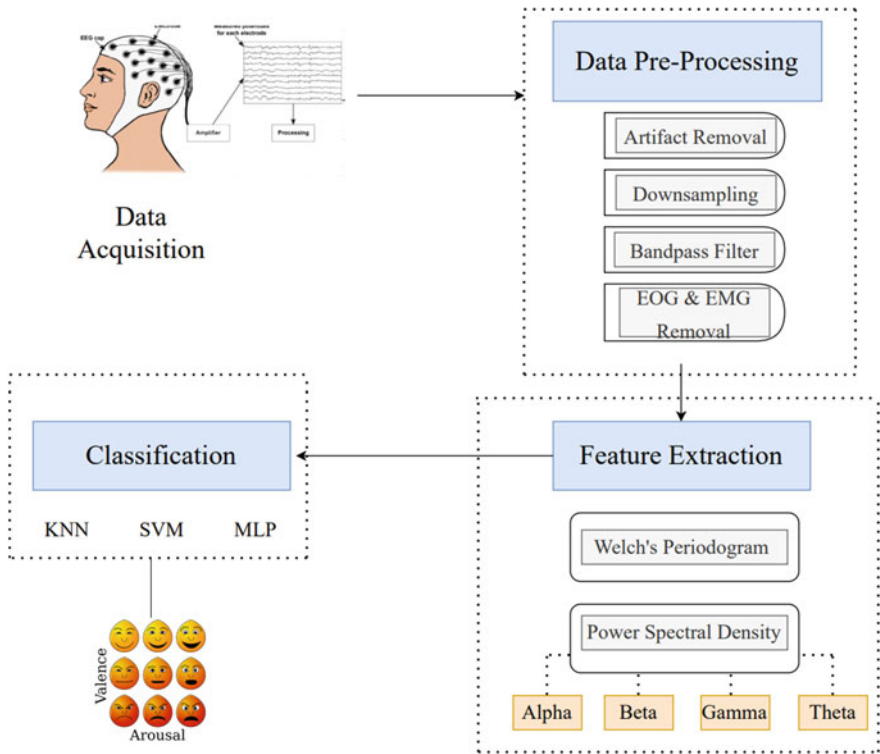
In [7], to discern emotion from unprocessed EEG information, a deep learning approach is suggested. The thick layer categorizes the learned information into low/high arousal, valence, and liking using the Long-Short Term Memory (LSTM) [8] technique. This method's average accuracy for the arousal, valence, and liking classes is 85.65%, 85.45%, and 87.99%, respectively, according to the DEAP dataset. According to Fabian Parsia George et al. [9], a technique for emotion identification based on time-frequency domain statistical data has been developed. The DEAP dataset, which includes 32 people from various gender and age categories, is used to train and test the SVM classifier after the best features have been chosen using a box-and-whisker plot. The experimental findings reveal that the suggested strategy had a 92.36% accuracy rate for the dataset we studied. The suggested method also surpasses state-of-the-art techniques by being more accurate.

T. D. Kusumaningrum et al. [10] state that EEG signal is utilized and examined. The subject's emotional state was then classified using the support vector machine [11] and Random Forest methods, and the outcomes were compared to those of other machine learning techniques. The experiment's findings indicate that the system with the best recognition accuracy is 62.58% of the data were generated using the Random Forest approach [12]. In [13], human's mood can be assessed using an electroencephalogram (EEG), which records input from brain signals and studies impulses. In most cases, the evaluation involved evaluating a person's mental state after being exposed to a stimulus with uncertain immediate outcomes. The linked categories in this investigation were normal, focused, depressed, and startled. Fifty individuals' raw brainwave data were captured using the Neurosky Mindwave, a single-channel EEG. In the meanwhile, the evaluations were completed while reading, watching, and listening to music kept the applicants' thoughts active. The Fast Fourier Transform (FFT) [14] technique was used to extract features, and the K-nearest Neighbors (K-NN) algorithm was used to categorize brain impulses. In [15], developed EmotioNet, which is a CNN based on 3-D covariance shift adaption, obtains classification rates of with a deep prediction layer of considering arousal and valence, the top values were 73.3% and 72.1% performance of a few earlier investigations. Significantly, our findings rely on automated feature extraction, in contrast to earlier handmade elements. Consequently, EmotioNet [16, 17] offers a novel EEG-based technique for emotion recognition.

### 3 Proposed System

Electroencephalogram signals from the DEAP dataset and machine learning techniques are used in the proposed method to identify emotions. Data Acquisition, Data pre-processing, feature extraction, and classification are the system's components. Figure 1 shows the proposed system for the emotion recognition using EEG Signals and machine learning algorithms in DEAP dataset.

Thirty-two individuals' electroencephalograms (EEG) and peripheral physiological data were monitored while they viewed 41-min-long music video snippets.



**Fig. 1** Emotion recognition system using electroencephalography signals

Participants assigned ratings to each video based on its arousal, valence, like/dislike, dominance, and familiarity levels. Frontal face footage was also taken for 22 of the 32 participants. Utilizing retrieval by emotional tags from the website last.fm, video highlight recognition, and an online evaluation tool, a unique strategy for selecting stimuli was applied.

### 3.1 Data Acquisition

Data acquisition is the process of taking samples of signals that represent actual physical conditions and converting those samples into digital numerical values that a computer can manipulate. We made use of the DEAP dataset for this method.

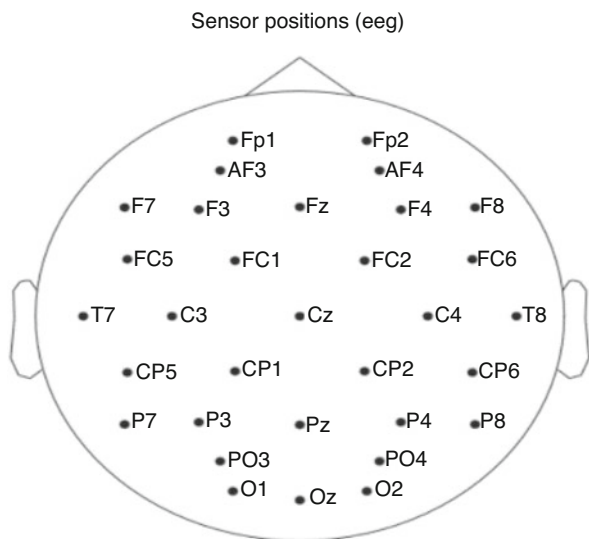
### 3.1.1 DEAP Dataset

The DEAP [18] database comprises the EEG physiological signals of 32 individuals (18 men and 18 women, ages 19–37; average: 26.7), which were captured as they watched 40 music videos with a total runtime of 1 min each on various topics of contention. Each subject relaxedly recorded a 2-min-long EEG signal while seeing a fixation cross on the screen prior to the viewing. The EEG signals were captured at 512 Hz from the following 32 places (as per the worldwide 10–20 positioning system): Fp1, AF3, F3, F7, FC5, FC1, C3, T7, CP5, CP1, P3, P7, PO3, O1, Oz, Pz, Fp2, AF4, Fz, F4, F8, FC6, FC2, Cz, C4, T8, CP6, CP2, P4, P8, PO4, and O2.

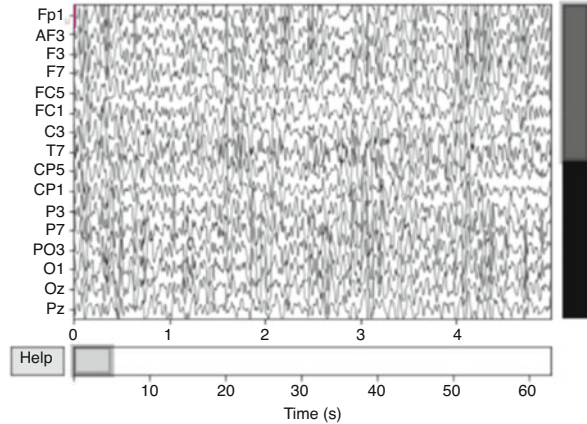
The valence-arousal scale was used to illustrate how the suggested music videos affected the emotions of various viewers. The degree of valence and arousal was evaluated by utilizing the self-assessment questionnaire, and the participants were required to score each video on its arousal, valence, like/dislike, dominance, and familiarity. There was an online assessment of the same videos that could be compared. All the subjects saw the identical movies; however, each subject’s visualization was done in a different order at random. In this study, just the aspects of valence and arousal were considered initially in Figs. 2 and 3.

Four categories [19]—Positive Valence, Negative Valence, High Arousal, and Low Arousal—can be used to group the data from the Deep dataset. Hedonic tone, also known as valence, is an emotive property that describes the inherent appeal or “goodness” (positive valence) or averseness or “badness” (negative valence) of a situation, an item, or an event. Additionally, the phrase describes and groups particular feelings. High arousal, also known as activation, is characterized by sensations of vigor for positive situations (such as joy) or tension for negative states

**Fig. 2** Electrode positions based on 10–20 electrode placement system



**Fig. 3** Sample EEG signals from for subject 1 showing arousal category



(e.g., fear). Between these and low arousal states like tranquilly and sadness, there are clear differences.

### 3.2 Data Pre-processing

Pre-processing, in general, is the act of converting raw data into a format that is more suited for additional analysis and user interpretable. Pre-processing in the context of EEG data often refers to the removal of noise to bring actual brain signals closer to the surface.

The EEG Analysis contains several pre-processing processes, including the following:

1. *Artifact removal*—Unwanted signals known as artifacts are typically caused by environmental noise, experimental mistake, and physiologic artifacts. The EEG data's quality may be harmed by these artifacts. To remove the artifacts or noise effectively, it is necessary to have a thorough understanding of the many forms of artifacts.
2. *Bandpass filter*—When examining EEG data, digital filtering is frequently used as a pre-processing step. Applying a high-pass filter to remove slow frequencies below 0.1 Hz and frequently even 1 Hz and a low-pass filter to remove frequencies over 40 or 50 Hz is the standard procedure in EEG data processing. The signals that span from 4 to 64 Hz were filtered in this case using a bandpass filter.
3. *Downsampling*—Reducing a signal's sample rate is the process of downsampling. By choosing one out of N samples, downsampling lowers the sampling rate of the input AOs by an integer factor. Note that the original data has not been subjected to an anti-aliasing filter.

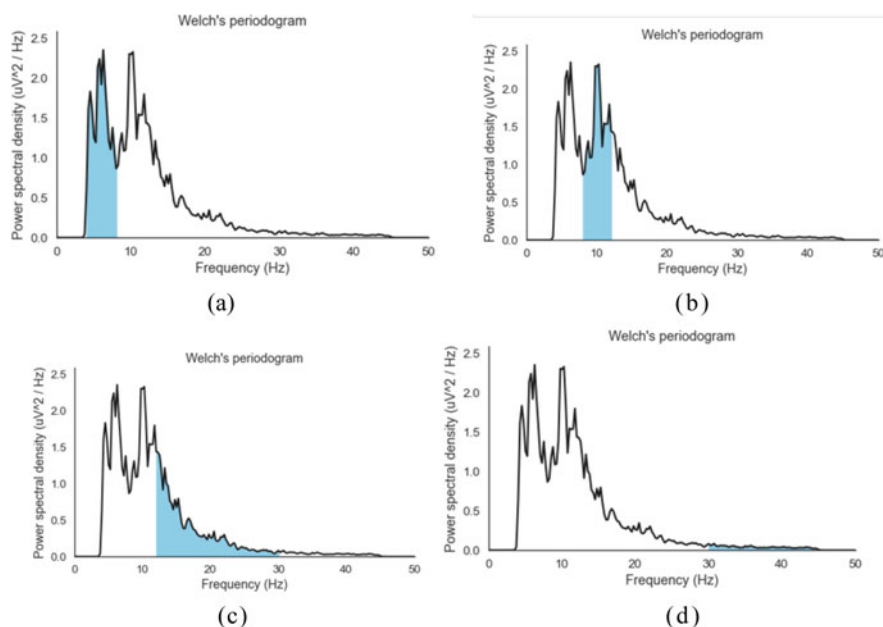
4. *EOG and EMG removal*—The action of the muscles of the face next to the electrode produces electrical “noise” called EMG. Electrical noise produced by eye movement is known as EOG. The wicked relatives of EEG, according to some, are EMG and EOG.

### 3.3 Feature Extraction

In the categorization of electroencephalogram (EEG) signal, feature extraction is a critical step. Extracted features are designed to reduce the loss of significant signal-embedded information.

Power spectral density (PSD), a feature extraction method, was used in this study. PSD is a helpful stationary signal processing method that performs well with narrowband signals. Standard signal processing methods include the distribution of signal power across frequencies and the visualization of energy intensity as a function of frequency. The Welch technique and the PSD were both applied in this experiment.

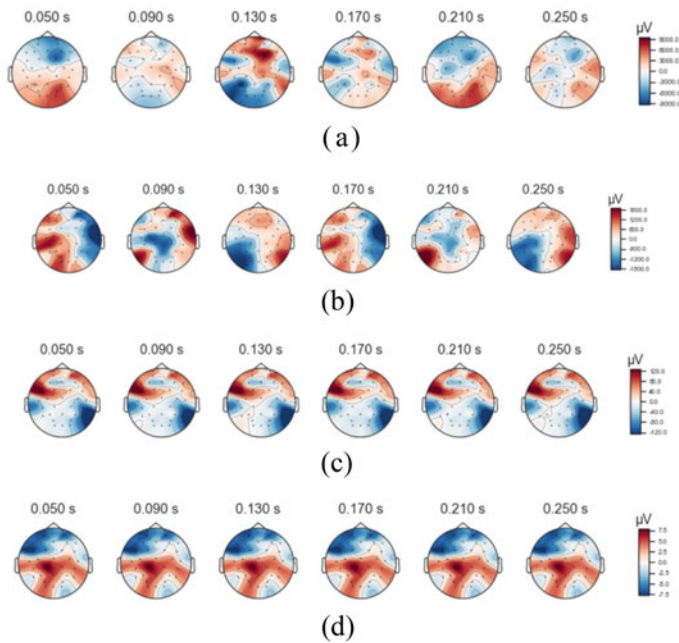
The Welch technique, a modified segmentation strategy, is employed to assess the typical periodogram. The power spectra density equation is defined first. The mean average of the periodogram for each interval is then represented using the Welch Power Spectrum. Figure 4 shows the Welch’s periodogram, defining the upper and lower limits of theta, alpha, beta, and gamma bands.



**Fig. 4** Welch’s periodogram defining the upper and lower limits—(a) theta (4–8 Hz), (b) alpha (8–12 Hz), (c) beta (12–30 Hz), and (d) delta (30–64 Hz)

**Table 1** Valence and arousal values based on the EEG bands and positions of the electrode

| Emotions    | EEG bands | Positions of the electrode |         |       |         |          |           |
|-------------|-----------|----------------------------|---------|-------|---------|----------|-----------|
|             |           | Left                       | Frontal | Right | Central | Parietal | Occipital |
| Valence (%) | Theta     | 58.54                      | 64.23   | 56.91 | 67.48   | 57.72    | 54.47     |
|             | Alpha     | 59.35                      | 51.22   | 60.16 | 65.04   | 62.60    | 55.28     |
|             | Beta      | 69.11                      | 65.04   | 66.67 | 60.98   | 65.85    | 64.23     |
|             | Gamma     | 60.16                      | 60.98   | 69.11 | 61.79   | 60.98    | 62.60     |
| Arousal (%) | Theta     | 52.03                      | 55.28   | 51.12 | 57.72   | 60.98    | 54.47     |
|             | Alpha     | 52.03                      | 54.47   | 52.85 | 58.54   | 60.98    | 51.22     |
|             | Beta      | 49.59                      | 49.59   | 47.97 | 52.85   | 60.98    | 47.15     |
|             | Gamma     | 60.16                      | 50.41   | 44.72 | 48.78   | 59.35    | 47.15     |



**Fig. 5** Topographies—(a) Theta, (b) alpha bands, (c) beta, and (d) gamma bands

EEG signals will be further categorized as alpha, beta, gamma, and theta using the Welch Technique. Table 1 shows the different EEG Signals with their range in Hz. Figure 5 shows the topographies of theta, alpha, beta, and gamma bands.



### 3.3.1 Classification

Following the feature extraction process, the extracted features are sent to the classifier, which then predicts the class of approach. Three types of classification algorithms were used in the classification step: SVM, KNN, and MLP.

SVM functions by mapping the data to a high-dimensional feature space so that data points may be categorized even when the data are not otherwise linearly separable. In contrast to KNN, which calculates the distances between a query and each example in the data, selects the K instances that are closest to the query, and then votes for the label with the highest frequency or averages the labels, a separator between the categories is discovered, and the data are then converted so that the separator may be represented as a hyperplane. A multilayer perceptron (MLP) is a fully connected type of feedforward artificial neural network. The word “MLP” is unclear; in some cases, it might refer to any feedforward ANN or especially to networks made up of several feedforward ANNs.

In this study, we focused on the channels or electrodes that gathered alpha, beta, gamma, and theta waves and were positioned on the left, right, central, parietal, and occipital locations of the brain. Different kernels (linear, sigmoid, rbf, and poly) were tested using SVC to see which was the best. Different k (odd) numbers, algorithms (auto, ball tree, kd tree), and weights (uniform, distance) are tested in KNN to determine which is the best, whereas MLP tests learning rate (alpha), solvers (adam, sgd, lbfgs), and activations (relu, tanh, logistic) to determine the same.

In the proposed work, KNN scored comparatively good accuracy scores when compared to the other two machine learning algorithms in all the corresponding situations as it is a robust technique for large noisy data. The samples are classified by the majority vote of neighbor's class.

## 4 Experimental Results

This section highlights the findings from the classification of human emotions using classifiers like SVM, KNN, and MLP. The results were compared using twofold cross validation to select the classifier. By training several ML models on subsets of the available input data and assessing them on the complementary subset of data, a technique known as cross-validation is used to evaluate ML models. Detect overfitting, or the inability to generalize a pattern, using cross-validation.

EEG band theta received scores of 58.54% on the left side, 64.23% on the frontal, 56.91% on the right, 67.48% on the central, 57.72% on the parietal, and 54.47% on the occipital. The average score for alpha was 59.35% on the left side, 51.22% on the frontal, 60.16% on the right, 65.04% on the central, 62.20% on the parietal, and 55.28% on the occipital. Beta band scored 69.11% in the left side, 65.04% in the frontal, 66.67% in the right, central by 60.98%, parietal by 65.85%, and occipital by 64.23% in the case of valence, while gamma scored 60.16% in the left side,

**Table 2** Valence and arousal values based on the EEG bands and positions of the electrode,  $k$  fold = 1

| Emotions    | EEG bands | Positions of the electrode |         |       |         |          |           |
|-------------|-----------|----------------------------|---------|-------|---------|----------|-----------|
|             |           | Left                       | Frontal | Right | Central | Parietal | Occipital |
| Valence (%) | Theta     | 61.65                      | 69.44   | 61.31 | 71.84   | 61.76    | 58.82     |
|             | Alpha     | 62.12                      | 53.12   | 64.23 | 68.61   | 65.67    | 58.65     |
|             | Beta      | 69.84                      | 68.61   | 71.72 | 63.64   | 68.66    | 68.57     |
|             | Gamma     | 62.60                      | 63.64   | 73.24 | 66.67   | 65.22    | 68.49     |
| Arousal (%) | Theta     | 62.75                      | 62.50   | 62.96 | 64.47   | 67.11    | 62.58     |
|             | Alpha     | 62.89                      | 61.64   | 63.80 | 65.33   | 63.77    | 64.67     |
|             | Beta      | 51.33                      | 61.33   | 56.76 | 64.03   | 55.65    | 56.60     |
|             | Gamma     | 57.39                      | 57.75   | 52.41 | 55.85   | 55.05    | 54.05     |

60.98% in the frontal, 69.11% in the right, central by 61.79%, parietal by 65.85%, and occipital by 64.23 in case of valence.

EEG band theta received scores of 62.75% on the left side, 62.50% on the frontal region, 62.96% on the right, 64.47% on the central region, 67.11% on the parietal region, and 62.58% on the occipital. Alpha received scores of roughly 62.89% on the left side, 61.64% on the frontal, 63.80% on the right, 65.33% on the central, 63.77% on the parietal, and 64.67% on the occipital region. Gamma band scored 57.39% on the left side, 57.75% on the frontal region, 52.41% on the right side, 55.85% on the central region, 55.05% on the parietal region, and 54.05% on the occipital region in case of valence, while beta band scored 51.33% on the left side, 61.33% on the frontal region, 56.67% on the right side, 64.03% on the central region, 55.65% on the parietal region, and 56.60% on the occipital region in case of arousal. Table 2 shows the valence and arousal values based on the EEG bands and positions of the electrode in cross validation 0.

EEG band theta scored 61% on the left side, 69.44% on the frontal region, 61.31% on the right side, 71.84% on the central region, 61.76% on the parietal region, and 58.82% on the occipital region. The average score for alpha was 68.61% on the central region, 65.57% on the parietal region, and 58.65% on the occipital region. It was 62.12% on the left side, 53.12% on the frontal region, 64.23% on the right side, and 68.1% on the central region. Beta band scored 69.84% on the left side, 68.61% on the frontal region, 71.72% on the right side, 63.64% on the central region, 68.66% on the parietal region, and 68.49% on the occipital region in the case of valence, while gamma scored 62.60% on the left side, 63.64% on the frontal region, 73.24% on the right side, 66.67% on the central region, 65.22% on parietal region, and 68.49% on the occipital region in case of valence.

EEG band scores for the theta were 62.75% on the left side, 62.50% on the frontal region, 62.96% on the right side, 64.47% on the central region, 67.11% on the parietal region, and 62.58% on the occipital region. The average score for alpha was 62.89% on the left side, 61.64% on the frontal region, 63.80% on the right side, 65.33% on the central region, 63.77% on the parietal region, and 64.67% on the occipital region. In terms of valence, the beta band scored 51.33% on the

**Table 3** Accuracy comparison with other respective parameters based on the positions of the electrode, EEG band, classifier, and emotions,  $k$  fold = 0

| Emotion     | Position of the electrode | EEG band | Classifier | Precision | Recall | Support | F1 score | Accuracy |
|-------------|---------------------------|----------|------------|-----------|--------|---------|----------|----------|
| Valence (%) | Central                   | Theta    | KNN        | 63.0      | 60.0   | 53.0    | 62.0     | 67.0     |
|             | Left                      | Beta     |            | 61.0      | 77.0   | 53.0    | 68.0     |          |
|             | Right                     | Gamma    |            | 65.0      | 62.0   | 53.0    | 63.0     |          |
| Arousal (%) | Central                   | Alpha    | MLP        | 64.0      | 38.0   | 61.0    | 47.0     | 59.0     |
|             | Parietal                  | Theta    |            | 67.0      | 39.0   | 61.0    | 49.0     | 60.0     |

**Table 4** Accuracy comparison with other respective parameters based on the positions of the electrode, EEG band, classifier, and emotions,  $k$  fold = 1

| Emotion | Position of the electrode | EEG band | Classifier | Precision | Recall | Support | F1 score | Accuracy |
|---------|---------------------------|----------|------------|-----------|--------|---------|----------|----------|
| Valence | Central                   | Theta    | KNN        | 71.0      | 73.0   | 70.0    | 72.0     | 67.0     |
|         | Left                      | Beta     |            | 79.0      | 63.0   | 70.0    | 70.0     |          |
|         | Right                     | Gamma    |            | 72.0      | 74.0   | 70.0    | 73.0     |          |
| Arousal | Central                   | Alpha    | MLP        | 56.0      | 79.0   | 62.0    | 66.0     | 59.0     |
|         | Parietal                  | Theta    |            | 57.0      | 81.0   | 62.0    | 67.0     | 60.0     |

left side, 61.33% on the frontal region, 56.76% on the right side, 64.03% on the central region, 55.65% on the parietal region, and 56.60% on the occipital region, while the gamma band scored 57.39% on the left side, 57.75% on the frontal region, 52.41% on the right side, 55.85% on the central region, 55.05% on the parietal region, and 54.05% on the occipital region in case of arousal. Table 3 shows the valence and arousal values based on the EEG bands and positions of the electrode in cross validation 1.

Theta band, beta band, and gamma band all achieved cross validation 0 scores of 67.0%, 61.0%, and 65.0%, respectively, using KNN classifier in valence. However, when it came to arousal, the theta band and alpha band, respectively, scored 60.0% and 59.0% using the MLPC classifier. Table 4 shows the accuracy comparison with other respective parameters based on the positions of the electrode, EEG band, classifier, and emotion using cross validation 0 in Fig. 6.

Using the KNN classifier in valence, the theta, beta, and gamma bands each earned cross validation 0 scores of 67.0%, 68.0%, and 69.0%, respectively. However, using the MLP classifier, the theta band and alpha band, respectively, scored 59.0% and 60.0% when it comes to arousal. Using cross validation 0 the accuracy comparison with other pertinent metrics depending on the placements of the electrode, EEG band, classifier, and emotion in Fig. 7.

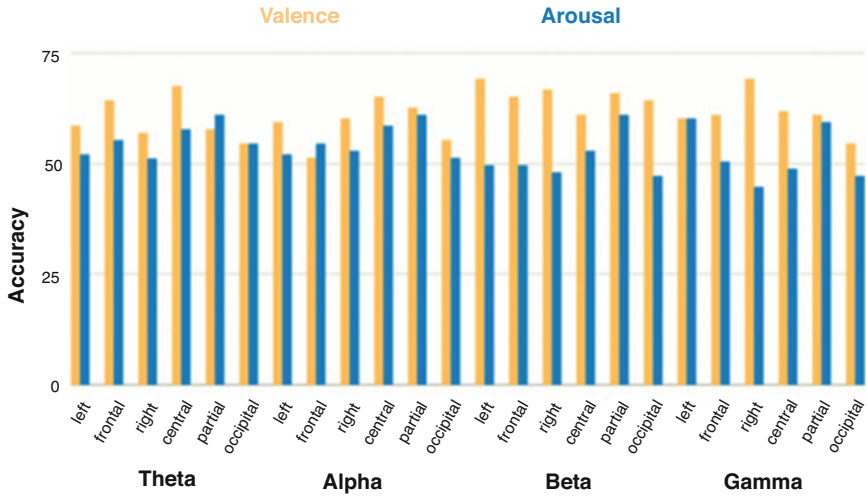


Fig. 6 Accuracy comparison based on the position of the electrodes and the EEG bands,  $k$  fold = 0

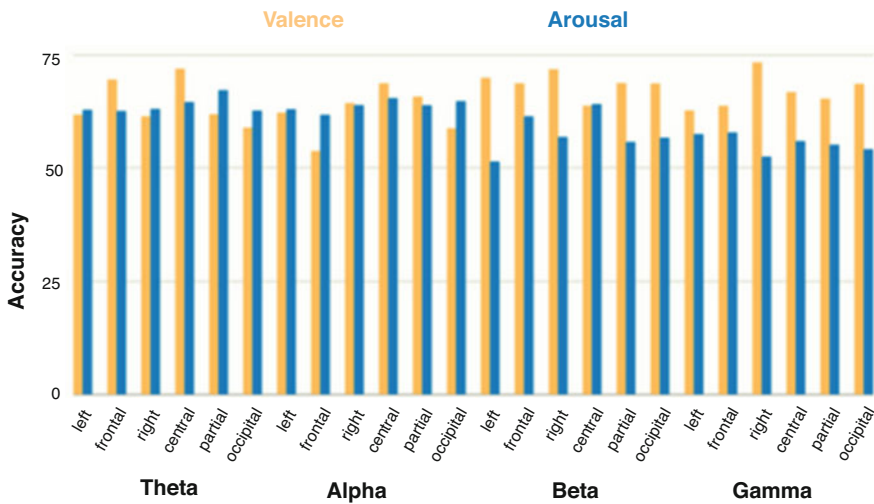


Fig. 7 Accuracy comparison based on the position of the electrodes and the EEG bands in,  $k$  fold = 1

## 5 Summary and Conclusion

In psychology, the term “emotion recognition” describes the process of attributing emotional states based on the observation of nonverbal visual and aural clues. Nonverbal signals are those given by a sender a person expressing an emotional response and include facial, vocal, postural, and gestural indicators.

Using biological brain signals to identify human emotions is becoming more and more appealing. Brain activity may be measured using electroencephalography (EEG), a trustworthy and affordable technique. To meet the criteria of a brain-computer interface, certain procedures must be carried out to detect emotion using EEG data (BCI). These procedures usually involve cleaning out artifacts from the EEG signals, extracting temporal or spectral information from the signal's time or frequency domain, and, eventually, creating a multi-class classification approach. The accuracy of the emotion classification approach is significantly improved by feature quality.

In this study, we employed the DEAP dataset, which consists of EEG signals that were divided into valence and arousal categories, for the emotion recognition task. Data pre-processing, which involves removing undesired or noisy data, came after data acquisition. Important features were chosen for the classification stage during feature extraction. Theta, beta, alpha, and gamma signals were separated from the EEG data during the feature extraction stage using the power density spectrum. KNN, SVM, and MLP were a few of the machine learning algorithms employed for the classification.

**Acknowledgment** This work is partially supported by the Department of Science and Technology—ICPS, Ministry of Science and Technology, Government of India (grant number: DST/ICPS/CLUSTER/Data Science/2018/General).

## References

1. C. Qing, R. Qiao, X. Xu, Y. Cheng, Interpretable emotion recognition using EEG signals. *IEEE Access* **7**, 94160–94170 (2017)
2. N.S. Suhaimi, J. Mountstephens, J. Teo, EEG-based emotion recognition: A state-of-the-art review of current trends and opportunities. *Comput. Intell. Neurosci.* **2020**, 8875426 (2020)
3. Y. Luo, G. Wu, S. Qiu, Y. Luo, S. Yang, W. Li, Y. Bi, EEG-based emotion classification using deep neural network and sparse autoencoder. *Front. Syst. Neurosci.* **14**, 43 (2020)
4. P. Lahane, A.K. Sangaiah, An approach to EEG based emotion recognition and classification using kernel density estimation. *Procedia Comput. Sci.* **48**, 574–581 (2015)
5. O. Bazgir, Z. Mohammadi, S.A.H. Habibi, Emotion recognition with machine learning using EEG signals, in *25th National and 3rd International Iranian Conference on Biomedical Engineering (ICBME)*, (IEEE, 2018)
6. A.T. Sohaib, S. Qureshi, J. Hagelback, O. Hilborn, P. Jerici, Evaluating classifiers for emotion recognition using EEG, in *International Conference on Augmented Cognition AC 2013: Foundations of Augmented Cognition*, ed. by D.D. Schmorow, C.M. Fidopiastis, (Springer, 2013), pp. 492–501
7. S. Gannouni, A. Aledaily, K. Belwa, H. Aboal, Emotion detection using electroencephalography signals and a zero-time windowing-based epoch estimation and relevant electrode identification. *Sci. Rep.* **11**, 7021 (2021)
8. S. Alhagry, A.A. Fahmy, R.A. El-Khoribi, Emotion recognition based on EEG using LSTM recurrent neural network. *Int. J. Adv. Comput. Sci. Appl.* **8**(10), 355–358 (2017)
9. F.P. George, I.M. Shaikat, P.S. Ferdawoos, M.Z. Parvez, J. Uddin, Recognition of emotional states using EEG signals based on time-frequency analysis and SVM classifier. *Int. J. Electr. Comput. Eng.* **9**(2), 1012–1020 (2019)

10. T.D. Kusumaningrum, A. Faqih, B. Kusumoputro, Emotion recognition based on DEAP database using EEG time-frequency features and machine learning methods. *J. Phys. Conf. Ser.* **1501**, 012020 (2020)
11. X.-W. Wang, D. Nie, B.-L. Lu, EEG-based emotion recognition using frequency domain features and support vector machines, in *International Conference on Neural Information Processing ICONIP 2011: Neural Information Processing*, ed. by B.L. Lu, L. Zhang, J. Kwok, (Springer, Berlin, Heidelberg, 2011), pp. 734–743
12. C. Akalya devi, D. Karthika Renuka, S. Soundarya, An EEG based emotion recognition and classification using machine learning techniques. *Int. J. Emerg. Technol. Innov. Eng.* **5**(4), 744–750 (2019) ISSN: 2394-6598
13. A. Yudhana, A. Muslim, D.E. Wati, I. Puspitasari, A. Azhari, M.M. Mardhia, Human emotion recognition based on EEG signal using fast Fourier transform and K-nearest neighbor. *Adv. Sci. Technol. Eng. Syst. J.* **5**(6), 1082–1088 (2020)
14. X. Li, S. Dawei, Z. Peng, Z. Yazhou, H. Yuexian, H. Bin, Exploring EEG features in cross-subject emotion recognition. *Front. Neurosci.* **12**, 162 (2018)
15. Y. Wang, Z. Huang, B. McCane, P. Neo, EmotioNet: A 3-D Convolutional Neural Network for EEG-based emotion recognition, in *2018 International Joint Conference on Neural Networks (IJCNN)*, (IEEE, 2018), pp. 1–7
16. P. Lahane, M. Thirugnanam, Human emotion detection and stress analysis using EEG signal. *Int. J. Innov. Technol. Explor. Eng.* **8**(4S2), 96–100 (2019) ISSN: 2278-3075
17. S. Thejaswini, K.M. Ravikumar, L. Jhenkar, A. Natraj, K.K. Abhay, Analysis of EEG based emotion detection of DEAP and SEED IV databases using SVM. *Int. J. Recent Technol. Eng.* **8**(1C), 1–6 (2019) ISSN: 2277-3878
18. S. Koelstra, C. Muehl, M. Soleymani, J.-S. Lee, A. Yazdani, T. Ebrahimi, T. Pun, A. Nijholt, I. Patras, DEAP: A database for emotion analysis using physiological signals (PDF). *IEEE Trans. Affective Comput.* **3**(1), 18–31 (2012)
19. K.R. Scherer, What are emotions? And how can they be measured. *Soc. Sci. Inf.* **44**(4), 695–729 (2005)

# Detection and Classification of Pneumonia and COVID-19 from Chest X-Ray Using Convolutional Neural Network



L. Swetha Rani, J. Jenitta , and S. Manasa

## 1 Introduction

As per World Health Organization (WHO) as of August 2022, there are 603,711,760 confirmed cases and 6,484,136 confirmed deaths [1]. The virus that is responsible for the Corona Virus Disease (COVID-19) spreads from person to person at a very high rate. The COVID-19 virus can cause lung complications which also looks similar as pneumonia. If a person gets affected by pneumonia or COVID-19, his lungs are filled with fluid and inflamed which leads to breathing difficulty. If breathing problems becomes severe then the patient may require hospitalization and ventilator treatment is required [1]. To avoid all such severe condition and hospitalization, early diagnosis of pneumonia and COVID-19 is very much necessary. It is always better to diagnose the diseases at an early stage to treat it in a better way. The pneumonia and COVID-19 can be diagnosed by blood test, chest X-ray, pulse oximetry, computed tomography Images (CT), and sputum test [2]. CT scanning has its own drawback. From chest X-radiation (X-ray), if the disease is diagnosed early then, we can treat the patient accordingly, and by isolating the patient, we can stop speeding of pandemic disease, too. The programmed identification framework can work with the early screening of pneumonia and opportune clinical intercessions. In any case, there actually exist numerous nodule candidates delivered by starting harsh discovery in this framework, and how to decide credibility is a key issue. We set forward multi-resolution convolutional neural networks (CNN) to separate highlights of different levels and resolutions from various depth layers in the organization for order of pneumonia competitors. Lung nodule identification and

---

L. Swetha Rani · J. Jenitta (✉) · S. Manasa  
Department of Electronics and Communication Engineering, AMC Engineering College,  
Bengaluru, India  
e-mail: [jenitta.jebaraj@amceducation.in](mailto:jenitta.jebaraj@amceducation.in); [manasa.srinivas@amceducation.in](mailto:manasa.srinivas@amceducation.in)

division assumes a significant part in pneumonia finding. It is a difficult errand inferable from the shape and force varieties of a lung nodule. Accordingly, the specialists, radiologists might have issue to appropriately recognize the little lung knobs and their sort. To resolve this issue, in this paper, we propose a deep learning method to detect and classify pneumonia and COVID-19 using chest X-ray. We use tensor stream programming and labelling programming to recognize the area of the stone; CNN calculations like back-engendering for arrangement are proposed and examined.

## 2 Literature Survey

Marios Anthimopoulos et al. proposed a CNN [3] that is intended for the characterization of ILD. In this article, they propose and assess Convolutional Neural Networks (CNNs). The proposed network comprises of five convolutional layers with  $2 \times 2$  bits and Leaky ReLU enactments, trailed by normal pooling which has size equivalent to the size of the last component guides and three thick layers. They have used the dataset of 14,696 picture patches and inferred by 120 CT examines from various scanners and emergency clinics. The large number of parameters, the relatively slow training and fluctuation of the results, for the same input could be considered as a drawback of this method.

Ali Narin proposed Feature Extraction using ResNet-50 Convolutional Neural Network which could detect COVID-19. In this work, chest X-ray images, which can be obtained effectively and rapidly, were utilized [4]. This method needs the help of radiology subject matter experts and decreases the rate of false discovery. But in this, work the strength of Dataset is limited.

Junfeng Li proposed COVID GATNet Architecture to detect COVID-19 from X-ray images [5]. COVID-19 CXR Dataset is used in this work. The review coordinates three CXR informational indexes distributed on the web and Kaggle rivalry, including CXR pictures of solid, different sorts of pneumonia, and COVID-19 positive patients. Since there is less information for COVID-19 positive CXR pictures than the other two kinds of information, this exploration widened COVID-19 positive CXR images by scaling, pivoting, changing brilliance, and other increment strategies for picture information. The method produced the average accuracy of the model up to 94.3%.

Zhaohui Liang et al. proposed Deep Convolutional Generative Adversarial Networks cGAN Architecture and Optimization to detect COVID-19 from X-ray images [6]. The aim of this work is to gain a planning from the typical chest X-ray visual examples to the COVID-19 pneumonia chest X-ray designs. This study used a sequential CNN architecture which produced an accuracy of 93%.

Mohit proposed 2D CNN architecture. This paper intends to incorporate Artificial Intelligence with clinical science to foster a grouping instrument to perceive COVID-19 contamination and other lung illnesses [7]. Four circumstances considered were COVID-19 pneumonia, non-COVID-19 pneumonia, pneumonia, and



typical lungs. The proposed AI framework is separated into two phases. Stage one group chest X-ray into pneumonia and non-pneumonia. Stage two gets input from stage one if the X-ray has a place with the pneumonic class and further characterizes it into COVID-19 positive and COVID-19 negative.

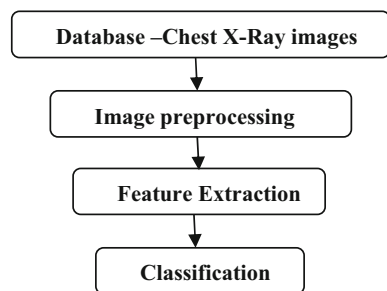
Harsh Sharma proposed two CNN architectures: one with a dropout layer and another without a dropout layer to separate elements from pictures of chest X-ray and group the pictures to recognize in the event that an individual has pneumonia. To assess the impact of dataset size on the presentation of CNN, the proposed CNN's utilizing both the first as well as increased dataset. This work used early stopping and batch normalization [8].

### 3 Proposed Method

Figure 1 shows the flow diagram of the proposed model. The chest X-rays of total of 2000 images are taken as input. Out of that 80% and 20% of the database images are used to train and test the model, respectively. The input images have undergone preprocessing. Image enhancement is done to change the characteristics of an image to make it suitable to a task. Image segmentation is also done to partition the digital image into many segments to extract the region of interest from the whole image. The input X-ray image is preprocessed with the utilization of Discrete Wavelet Transform (DWT). Picture improvement or pre-handling of picture is done to discard of noise and light up the photo simplifying it to become mindful of the key abilities. The explanation of utilizing wavelets is to change over an information picture into a progression of wavelets, which can be put away more prominent productively contrasted with pixel blocks. The picture is broken down into high pass and low pass channels by applying discrete wavelet.

Deterioration of picture results into four sub-groups given as, HL, HH, LL, and LH. The LL sub-band contains the majority of the insights while the other better request groups contain the edges inside the upward, slanting, and even way. By utilizing Tensorflow engineering, the information goes toward one side, courses through the arrangement of numerous tasks, and comes out the opposite end as a result.

**Fig. 1** Flow diagram of the proposed method



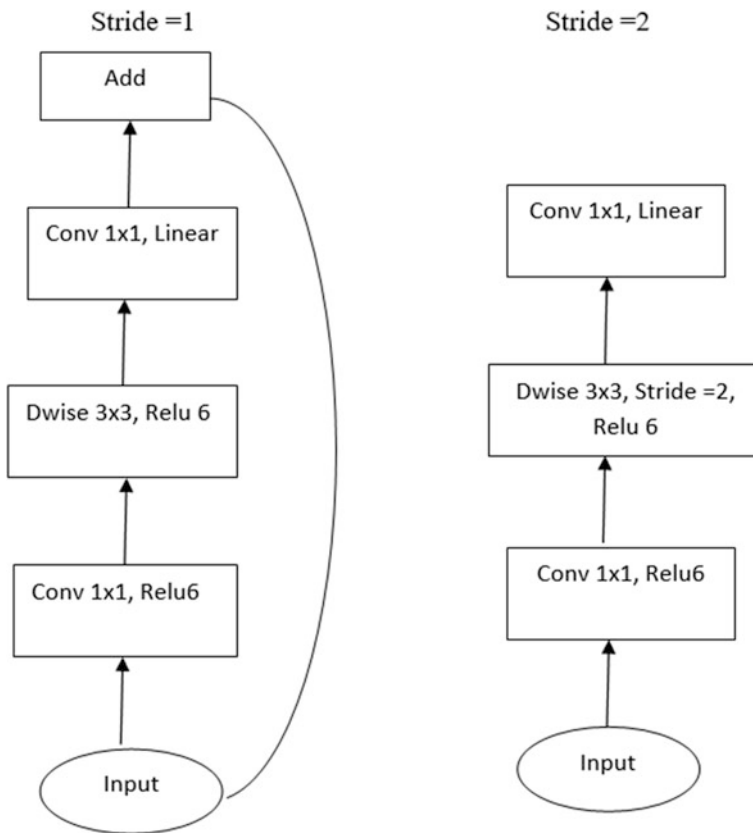


Fig. 2 Architecture of MobileNet V<sub>2</sub>

In this proposed work, MobileNet V2 architecture is used. This preprocessed image is given as an input to the input layer of the MobileNet. In the convolution layer, the features of the images are extracted. Based on the extracted features, the disease is detected and it is classified. The MobileNet V2 architecture has 53 convolution layers and 1 Average Pool with nearly 350 GFLOP. Inverted Residual Block and Bottleneck Residual Block are the two main components. There are two types of Convolution layers in MobileNet V2 architecture: (1)  $1 \times 1$  Convolution. (2)  $3 \times 3$  Depth wise Convolution. Each block has three different layers: (1)  $1 \times 1$  Convolution with Relu6. (2) Depth wise Convolution. (3)  $1 \times 1$  Convolution without any linearity. These are the two different components in MobileNet V2 model which is shown in Fig. 2.

### 4 Experimental Results

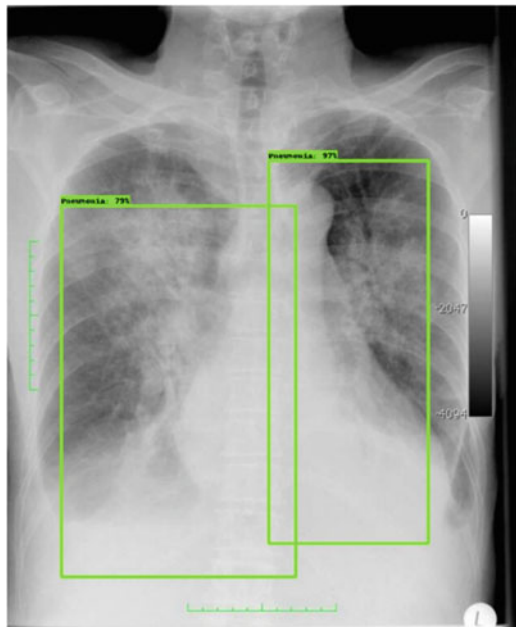
The input data set for the proposed work is taken from Kaggle. Total of 2000 images are taken. Eighty percent is used for training the model and 20% for testing the model. The images are labeled using LabelImg. The model is trained and tested. The obtained results are shown in Figs. 3, 4, and 5. Figure 3 shows the disease is classified as pneumonia and both lungs are affected by it. Figure 4 shows that disease is classified as both pneumonia and COVID-19 that has affected only one lung. Figure 5 shows that the disease is classified as only pneumonia that has affected only one lung.

The number of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) are calculated from the confusion matrix. Using Eqs. (1)–(4) the Accuracy, Precision, Recall, and F1 score are calculated.

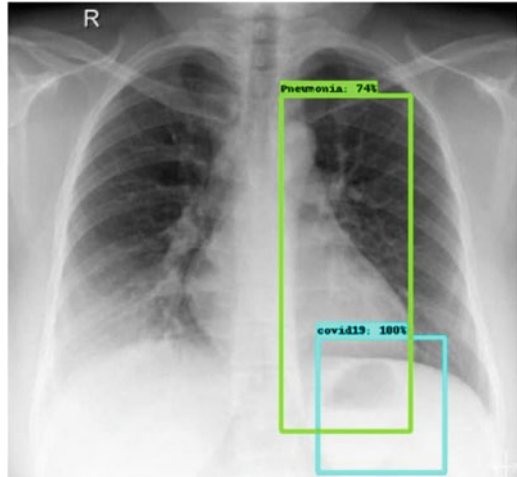
$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{Total inputs}} \tag{1}$$

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

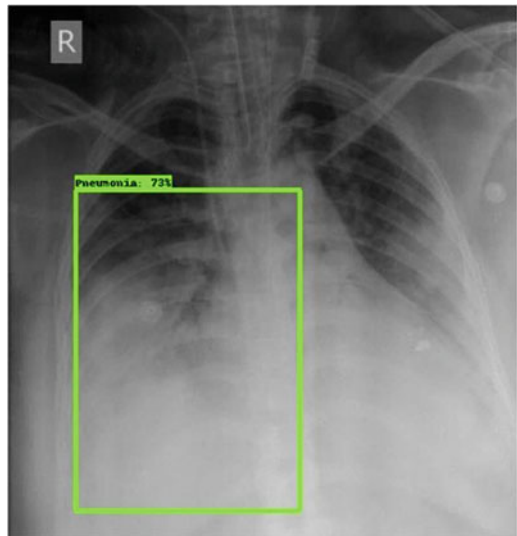
**Fig. 3** Infection in both the lungs identified and classified as pneumonia



**Fig. 4** Infection in both the lunges identified and classified as pneumonia with 74% confidence and COVID in one lunge as 100% confidence



**Fig. 5** Infection in both the lunges identified and classified as pneumonia with 94% confidence



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

$$\text{F1 score} = 2 \times \frac{\text{Precision} \times \text{recall}}{(\text{Precision} + \text{recall})} \quad (4)$$

For the proposed network, the obtained Accuracy is 98.3%, Precision is 0.97, Recall is 0.98, and F1 score is 0.98. The above experimental results confirm that the

proposed method performs very well in the classification of the disease pneumonia and COVID-19.

## 5 Conclusion and Future Work

COVID-19 has become a pandemic disease, and the mortality rate is very high. Early detection of infection of COVID can stop spreading this deadly disease. To classify whether the patient is affected by COVID-19 or pneumonia, this proposed method is used. This method uses CNN with MobileNet V2 architecture. This method gives an accuracy of 98.3%.

## References

1. WHO Homepage, <https://covid19.who.int/>. Last accessed 2 Dec 2022
2. Pneumonia Homepage, <https://www.who.int/news-room/fact-sheets/detail/pneumonia>. Last accessed 2 Dec 2022
3. F. Marios Anthimopoulos, S. Stergios Christodoulidis, T. Lukas Ebner, A. Christe, F. Stavroula Mouggiakakou, Lung pattern classification for interstitial lung diseases using a deep convolutional neural network. *IEEE Trans. Med. Imaging* **35**, 1207–1216 (2016)
4. A.F. Narin, C.S. Kaya, Z.T. Pamuk, Automatic detection of coronavirus disease (COVID-19) using X-ray images and deep convolutional neural networks. *Pattern Anal. Appl.* **24**, 1207–1220 (2021)
5. J. Li, D.F. Zhang, Q.S. Liu, R.T. Bu, Q.F. Wei, COVID-GATNet: A deep learning framework for screening of COVID-19 from chest X-ray images, in *2020 IEEE 6th International Conference on Computer and Communications (ICCC)*, (IEEE, 2020), pp. 1897–1902
6. Z. Liang, J.X. Huang, J. Li, S. Chan, Enhancing automated COVID-19 chest X-ray diagnosis by image-to-image GAN translation, in *IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, (IEEE, 2020)
7. F. Mohit Mishra, S. Varun Parashar, T. Rushikesh Shimpi, Development and evaluation of an AI System for early detection of Covid-19 pneumonia using X-ray, in *IEEE Sixth International Conference on Multimedia Big Data (BigMM)*, (IEEE, 2020)
8. R. Karthik, R. Menaka, M. Hariharan, Learning distinctive filters for COVID-19 detection from chest X-ray using shuffled residual CNN. *Appl. Soft Comput.* **99**, 106744 (2021)

**Part V**  
**Bigdata in Future Advancements**

# Stopwords Aware Emotion-Based Sentiment Analysis of News Articles



Chhaya Yadav and Tirthankar Gayen

## 1 Introduction

Sentiment analysis is procedurally distinguishing, extricating, evaluating, and contemplating abstract data by devices of computational phonetics, computational linguistics, biometrics, and text analysis. The sentiment analyzer assesses what constitutes a positive, neutral, or negative piece of writing. Sentiment analysis is helpful for data analysts within large enterprises for conducting research on market, public opinion, brand reputation, and understanding customer experiences [1]. The already existing product or a service is the object considered for sentiment analysis. Sentiment analysis can be broadly classified into four types, namely, fine-grained, emotion-based, aspect-based, and intent-based sentiment analysis. In fine-grained sentiment analysis, the polarity of the opinion is determined by simple binary classification of positive and negative sentiment. Depending on the use case, this type may also fit within the higher specification [2]. To find indications of particular emotional states mentioned in the text, emotion detection is used. Advanced sentiment analysis is done using aspect-based sentiment analysis, and the objective is to convey opinion with respect to the certain part of the input. The intent-based sentiment analysis ascertains the type of intention that the message is expressing. Opinion mining is synonymous to sentiment analysis despite the general understanding that sentiments are emotionally loaded opinions [3].

Analyzing enormous amount of unstructured text into a more specific news articles, devising suitable algorithms to understand the opinion from text and finding positive and negative score out of it is a challenging task. Thus, there is a need to incorporate suitable techniques to improve the accuracy of the results obtained from

---

C. Yadav (✉) · T. Gayen  
Jawaharlal Nehru University, New Delhi, India

sentiment analysis of the news articles [4]. Although there are several approaches concerned with sentiment analysis of news articles, the outputs provided by these approaches lack accuracy to a considerable extent [5].

Negative stopwords include important information about the sentiment of the sentence, yet it has been discovered that most sentiment analysis pre-processing techniques discard these stopwords [6]. As a result, several times semantic information gets lost, resulting in inaccurate sentiment analysis. The contributions of importance of this article are as follows:

1. This article presents an approach where every sentence is processed and examined at the sentence level to establish its polarity. In order to minimize the loss of significant information for labelling news articles, the proposed approach does the pre-processing considering the negative stopwords and labels the sentiments of the article using Support Vector Machine (SVM).
2. The results obtained after applying the proposed approach on the dataset of different categories of news articles obtained from BBC are usually found to yield a comparatively higher accuracy for providing the sentiment polarity of various news items.
3. A regression analysis is presented to confirm that the proposed approach provides comparatively better accuracy for sentiment analysis of news articles.

A survey of relevant research on sentiment analysis of text data is described in Sect. 2 of this article. The proposed approach for sentiment analysis of news items is described in Sect. 3, which is followed by an implementation utilizing news articles from the BBC dataset in Sect. 4. This section also provides an evaluation of the proposed approach. Section 5 concludes with a discussion on the benefits and limitations of the proposed approach as well as outlines the goals and enhancements for future work.

## 2 Related Work

In Natural Language Processing (NLP), the field of sentiment analysis has been explored with a variety of approaches. There are plenty of researchers who are working on Sentiment Analysis of Texts. The researchers have extensively contributed to its development and enhancing its applications in various fields. There are various methods, ranging from dictionary-based methods to machine learning methods. In 2018, Urologin [7] proposed techniques for extracting and displaying text data. It performs combined text summarization and sentiment analysis. A text summarization technique based on pronoun replacement is created, and sentiment data is gathered using the VADER sentiment analyzer. However, their summarization approach may cause loss in semantic information, which can lead to a wrong sentiment analysis of the document. In their work, they used a standard sentiment analysis repository (Github repo VADER). Taj et al. [8] characterized news articles as positive, negative, and unbiased classes by gathering the all-out



opinion scores of the sentences in the article using lexical-based methodology. It implements Lexicon-based sentiment assessment of news stories, but it has insufficient or restricted word inclusion. As a result, numerous new lexical items with distinct semantics should be refreshed in lexical data set. It solely employs news articles in English from one hotspot for sentiment analysis. Vilasrao et al. [9] intended to develop a system with emotion dataset and training dataset to obtain valency (in the form of emotional and neutral). They have used Lexicon-based approach and Deep Learning Technology. They presented their estimation and investigation utilizing dictionary-based methodology and deep learning techniques to deal with emotion classes. The output of their proposed framework (which perceives the presence of assumptions extremity) can be used to enhance the sentiment analysis framework but their approach is only lexical based and uses very less amount of data for training using traditional machine learning (ML) algorithm. As a result, the accuracy of their output is not very high. Souma et al.'s [10] work was to forecast the financial news sentiments. They used Simple Sequential LSTM network architecture for the analysis, but in their work, an assumption (which may be error prone) is made that if the stock log return value is negative then the sentiment is negative and vice versa. No standard dataset was used to perform the experiment and obtain the results. No normalization of the statements in the news articles was done since same weightage was given to all the statements. Shirsat et al.'s [11] work was concerned with sentence level negation identification from news articles. They used a dictionary-based approach with ML techniques but no standard dataset was used (scraped dataset was used) and the used dataset was quite small. Their approach was dictionary based so no semantic information was used. For obtaining word-level emotion distribution Li et al. [12] considered the use of dictionary with word-level emotion distribution (known as NRC-Valence arousal dominance) for assigning emotions along with intensities to the sentiment words as efficient. Two models were proposed by Basiri et al. [13] in their study that employed a three-way decision theory and proposed two models. The three-way fusion of one deep learning model and the conventional learning method was used in the first model (3W1DT), whereas, three-way fusion of three deep learning models was used in the second model (3W3DT). The results obtained using [Drugs.com](#) dataset showed that both frameworks outperformed the traditional deep learning methods. In addition, it was noted that the first fusion model performed significantly better than the second model in terms of accuracy and F1-metric. Using the Rotten Tomato movie review dataset, Tiwari et al. [14] have implemented 3 ML algorithms (Maximum Entropy, Naive Bayes, and SVM) with the  $n$ -gram feature extraction technique. They noted a drop in accuracy for  $n$ -grams with larger values of  $n$ , such as  $n = 4, 5$ , and  $6$ . Using various feature vectors like Bag of Words (BOW), Unigram with Sentiwordnet Soumya et al.'s [15] work divided 3184 Malayalam tweets into negative and positive opinions. They used the ML algorithms Naive Bayes, Random Forest and found that Random Forest performed better (with an accuracy of 95.6%) with Unigram Sentiwordnet when negation words were taken into account. Rao et al. [16] used Long Short-Term Memory (LSTM) to improve sentiment analysis by first cleaning the datasets and removing the sentences with weaker emotional

polarity. On three publicly accessible document-level review datasets, their model outperforms the state-of-the-art models.

From the survey, it is found that although there are several approaches concerned with sentiment analysis, but the output provided by these approaches lacks accuracy to a considerable extent. In many approaches, it is found that no standard dataset was used to perform the experiment and obtain the results. In some approaches, no normalization of the statements in the news articles were done since same weightage was given to all the statements [17]. The dataset used in some of the approaches were quite small. In some approaches, no semantic information was used. In few approaches, there was loss in semantic information which lead to wrong sentiment analysis of the document. Analyzing enormous amount of unstructured text into a more specific news articles, devising suitable algorithms to understand the opinion from text and finding positive and negative score out of it is a challenging task. In essence, it is the process of determining the emotional tone behind a series of words, used to gain an understanding of the attitudes, opinions, and emotions expressed within an online mention. Negative stopwords carry significant information about the sentiment of the sentence, but it is found that most of the approaches concerned with sentiment analysis removed these stopwords during preprocessing stage [18]. As a result, there can be loss in semantic information leading to incorrect sentiment analysis [19]. Thus, there is a need to incorporate suitable techniques to improve the accuracy of the results obtained from the sentiment analysis [20]. Hence, the intention is to develop a suitable approach to improve the accuracy of sentiment analysis by considering the negative stopwords [21].

### 3 Proposed Approach

The polarity of the text data is determined or expressed by the sentiment analysis approach. Essentially, there are three layers of sentiment analysis: A sentiment analysis at the document level will be performed first to ascertain the polarity of the document. In the case of a text file containing only product reviews, the algorithm decides the polarity of all the content in the document. It is because of this that the document only conveys opinions about one specific subject and cannot be used to evaluate other products. Every sentence is processed and examined at the sentence level to establish its polarity. Finding emotions about things and their characteristics are made possible by aspect-level sentiment analysis. The proposed algorithm *Negative Stopwords Aware Sentiment Analysis* (NSASA) does the preprocessing considering the negative stopwords and labels the sentiments of the article using SVM. The steps of the proposed algorithm NSASA are as follows:

**Algorithm** Negative Stopwords Aware Sentiment Analysis (NSASA)

---

```

Step 01 : Initialisation of negation_words ;
Step 02: pos_list= set(opinion_lexicon.positive()); /* generating a set of positive
words */
Step 03: neg_list=set(opinion_lexicon.negative()); /* generating a set of negative
words */
Step 04: stop_words = stopwords.words('english'); /* storing english stop words */
Step 05: lemma = nltk.wordnet.WordNetLemmatizer(); /* lemmatizer object */
Step 06: opinionDict={}; /* dictionary which has key 'pos', 'neg' which contains
corresponding words */
Step 07: opinionDict['pos']= pos_list ;
Step 08: neg_list.update(negation_words) ; /* updating negative list with inclusion of
negation_words */

Step 09: opinionDict['neg'] = neg_list ;
Step 10: for word in negation_words:
    stop_words.remove(word) ; /* removes words in negation_words diction-
    ary from stop_words dictionary */
    end for
Step 11: sentiment_vectorizer = CountVectorizer(input = "content", encoding =
"utf-8", decode_error = "replace", ngram_range = (1, 1), preprocessor =
preprocessText, analyzer = "word", vocabulary = sentiment_words,
tokenizer = None); /* creating the vector */
Step 12: x_train = sentiment_vectorizer.transform(xtrain); /* dividing the dataset into
train dataset */
Step 13: x_test = sentiment_vectorizer.transform(xtest); /* dividing the dataset into
test dataset */

Step 14: x_train.shape
(360, 6823); /* fitting the model over the dataset*/
Step 15: model = LinearSVC(penalty = "l2", loss = "squared_hinge", dual = True, tol
= 0.0001, C = 1.0, multi_class = "ovr", fit_intercept = True, inter-
cept_scaling = 1, max_iter = 20000); /* fitting the model
over the dataset*/
Step 16: model.fit(x_train, ytrain); /* fitting the model over the dataset */
Step 17: predictions = model.predict(x_test); /* evaluative tool */
Step 18: print(classification_report(ytest, predictions)); /* outputting the evaluative
scores */

```

---

Algorithm NSASA begins with the initialization and storing of negative words in the `negation_words` variable depicted in step 1. Generation of a set of positive and negative words takes place in steps 2 and 3. English stopwords are stored in step 3 and lemmatizer object is created in step 4. A dictionary is created for storing separately the positive and negative set of words as depicted in steps 6–9. The dictionary containing stopwords is updated by removing negative words list from it as depicted in step 10. Training and testing of dataset takes place from steps 11 to 13. Step 14 fits the model over the dataset. SVM is applied on the model and the

predictions are made and final label is printed from steps 14 to 18. The model used in this proposed algorithm corresponds to LinearSVC.

Various user-defined functions which are used in the algorithm NSASA are defined in Table 1. Table 2 provides the descriptions of various predefined functions and parameters which are used in the algorithm NSASA.

## 4 Implementation Details and Evaluation

The proposed approach has been implemented using Python 3.7 with Google Colaboratory and the NLTK package. From the NLTK package `nltk.download('punkt')`, `nltk.download('stopwords')`, `nltk.download('opinion_lexicon')` were imported. Table 3 below displays the article's categorization and the proportion of neutral, favorable, and unfavorable terms in it. This study makes use of the Bing Liu dictionary, which has 4783 negative words and 2006 positive terms [11].

Table 4 shows the article's categorization and the proportion of neutral, favorable, and unfavorable terms in it after using the proposed algorithm NSASA. Additionally, it makes use of the Bing Liu dictionary, which has 4783 negative words and 2006 positive terms.

Figure 1 shows the percentage accuracy values obtained from the approach proposed by Shirsat et al. and the proposed NSASA for four different types of datasets.

For the purpose of evaluation, a comparison of the proposed approach using NSASA has been made with the approach provided by Shirsat et al. [11], and the results obtained are summarized in Table 5. Based on the results obtained after the experimentation it is found that the highest accuracy achieved by the approach proposed by Shirsat et al. in the Tech. data is 86%, and the lowest accuracy achieved by the approach proposed by Shirsat et al. in the business data is 75%. Whereas the highest accuracy achieved by the proposed NSASA in Tech. data is 98%, and the lowest accuracy achieved by the proposed NSASA in business data is 73%.

Thus, the approach of Shirsat et al. is found to provide results with an average accuracy of 80.25 whereas the proposed NSASA is found to provide results with an average accuracy of 85.75. The accuracy values obtained from the technique of Shirsat et al. and the proposed NSASA were also employed in a regression analysis, with the outputs reported in Table 6. Upon data analysis, it is discovered that the proposed NSASA algorithm's  $p$ -value is 0.009156.

Regression coefficient  $r$  (Multiple R) = 0.9908 and  $p < 0.05$  are obtained in Table 6. This suggests that the accuracy values acquired from Shirsat et al. [11], and the accuracy values obtained from the proposed NSASA algorithm have a positive relationship. Thus, the proposed NSASA seems to provide better accuracy in sentiment analysis of news articles as compared to the approach specified by Shirsat et al.

**Table 1** User-defined functions and their definitions

| Function                       | Definition   |
|--------------------------------|--|
| removeStopWords(text)          | <pre>removeStopWords(text) { word_tokens = word_tokenize(text.lower());   filtered_sentence = " ".join([w for w in word_tokens if     not w.lower() in stop_words]);    return filtered_sentence; }</pre>  |
| removeDigits(text)             | <pre>removeDigits(text) { res = " ".join([i for i in text if not i.isdigit()]);   return res; }</pre>  |
| remove_punctuation(text)       | <pre>remove_punctuation(text) { punctuationfree = " ".join([i for i in text if i not in   string.punctuation]);   return punctuationfree; }</pre>  |
| removeExtraSpaces(text)        | <pre>removeExtraSpaces(text) {return re.sub(' +', ' ', text);}</pre>   |
| textStemmer(text)              | <pre>textStemmer(text) {ps = PorterStemmer();   res = " ".join([ps.stem(i) for i in text.split() ]);   return res; }</pre>   |
| OpinionOfWord(word)            | <pre>OpinionOfWord(word) {if word in opinionDict['pos'];   return 1;   elif word in opinionDict['neg'];   return -1;   else;   return 0; }</pre>   |
| getOpinionListofSentence(text) | <pre>getOpinionListofSentence(text) {res = [OpinionOfWord(word) for word in text.split()];   return res; }</pre>   |
| getPosScore(opinionList)       | <pre>getPosScore(opinionList) { return opinionList.count(1)/len(opinionList); }</pre>  |
| getNegScore(opinionList)       | <pre>getNegScore(opinionList) {return opinionList.count(-1)/len(opinionList); }</pre>  |
| getNeuScore(opinionList)       | <pre>getNeuScore(opinionList) {return opinionList.count(0)/len(opinionList); }</pre>   |
| getGT(list)                    | <pre>getGT(list) {if getPosScore(list)&gt;getNegScore(list);   return 1;   elif getPosScore(list)&lt;getNegScore(list);   return -1;   else;   return 0; }</pre>   |
| getDiff(list)                  | <pre>getDiff(list) {return getPosScore(list) - getNegScore(list); }</pre>  |
| preprocessText(text)           | <pre>preprocessText(text) {filtered_sentence = removeStopWords(text);   removedPunctuationVar =   remove_punctuation(filtered_sentence);   removedDigitsVar   =removeDigits(removedPunctuationVar);   removedEx-   trSpacesVar=removeExtraSpaces(removedDigitsVar);   stemmedSentencetextStem-   mer(removedExtraSpacesVar);   return stemmedSentence; }</pre> |

**Table 2** Predefined functions, parameters, and their description

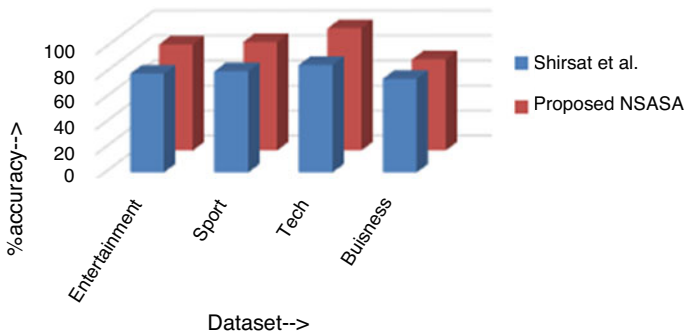
| Func-tions/parameters           | Descriptions  |
|---------------------------------|---|
| Set()                           | Function can be used to create sets   |
| Opinion_lexicon.positive()      | Return all positive words in alphabetical order   |
| Opinion_lexicon.negative()      | Return all negative words in alphabetical order   |
| Stopwords.words('english')      | Return list of stopwords stored in English  |
| nlk.wordnet.WordNetLemmatizer() | Returns the input word unchanged if it cannot be found in WordNet   |
| Update()                        | Inserts the specified items to the dictionary   |
| Stopwords.remove()              | Remove stop words from string   |
| penalty{'11', '12'}             | It specifies the standard that was applied in the penalty. The benchmark used in SVC, also known as Ridge Regression, is the "12" penalty   |
| loss = "squared_hinge"          | For "maximum margin" binary classification issues, the squared hinge loss is a loss function that is employed   |
| dual = True                     | This chooses the algorithm to solve the dual optimization problem or the primary optimization problem   |
| tol = 0.0001                    | It is tolerance of limiting standards   |
| C = 1.0                         | The Regulator C is parameter. The strength of the regularization is inversely connected to C. It can only be positive   |
| multi_class = "ovr"             | "ovr" trains n_classes one-vs-rest classifiers  |
| fit_intercept = True            | Calculations will use the intercept if it is set to True  |
| fit_intercept = True            | If it is set to True, calculations will use the intercept   |
| intercept_scaling = 1           | Intercept scaling lessens regularization's effects on feature weight and hence on the intercept   |
| max_iter = 20000                | The most iterations that can be performed   |
| negation_words                  | ['no','against','nor','not',"aren't",'couldn't','didn't','didn't','doesn','doesn't','hadn','hadn't','hasn','hasn't','haven','haven't','isn','isn't','ma','mightn','mightn't','mustn','mustn't','needn','needn't','shan','shan't','shouldn','shouldn't','wasn','wasn't','weren','weren't','won','won't','wouldn','wouldn't'] |

**Table 3** Category wise document polarity using Shirsat et al.

| Sr no | Name of category | Neutral | Negative | Positive | Total |
|-------|------------------|---------|----------|----------|-------|
| 1     | Business         | 34      | 214      | 262      | 510   |
| 2     | Entertainment    | 21      | 244      | 136      | 401   |
| 3     | Politics         | 17      | 190      | 210      | 417   |
| 4     | Sport            | 33      | 327      | 151      | 511   |
| 5     | Tech             | 21      | 244      | 136      | 401   |

**Table 4** Category wise document polarity using NSASA

| Sr no | Name of category | Neutral | Negative | Positive | Total |
|-------|------------------|---------|----------|----------|-------|
| 1     | Business         | 35      | 219      | 256      | 510   |
| 2     | Entertainment    | 15      | 87       | 284      | 401   |
| 3     | Politics         | 34      | 180      | 203      | 417   |
| 4     | Sport            | 35      | 117      | 359      | 511   |
| 5     | Tech             | 16      | 128      | 257      | 401   |



**Fig. 1** Obtained accuracy values for four different datasets

**Table 5** Evaluation results

| Dataset       | Shirsat et al. |           |                 | Proposed NSASA |           |                 |
|---------------|----------------|-----------|-----------------|----------------|-----------|-----------------|
|               | Accuracy       | Precision | <i>F</i> -score | Accuracy       | Precision | <i>F</i> -score |
| Entertainment | 79             | 81        | 79              | 85             | 88        | 84              |
| Sport         | 81             | 78        | 79              | 87             | 88        | 87              |
| Tech          | 86             | 89        | 87              | 98             | 98        | 98              |
| Business      | 75             | 69        | 72              | 73             | 69        | 77              |

## 5 Conclusion

The amount of text continually expanding is an invaluable source of knowledge and information that must be effectively retrieved to reap its benefits. It may be quite challenging to analyze the vast amount of unstructured content into more specialized news items to create the proper algorithms to extract opinions from text

**Table 6** Regression analysis results

| Regression statistics |              |                |               |                 |                       |
|-----------------------|--------------|----------------|---------------|-----------------|-----------------------|
| Multiple R            | 0.990844423  |                |               |                 |                       |
| R square              | 0.981772671  |                |               |                 |                       |
| Adjusted R square     | 0.972659007  |                |               |                 |                       |
| Standard error        | 1.693672311  |                |               |                 |                       |
| Observations          | 4            |                |               |                 |                       |
|                       | <i>df</i>    | SS             | MS            | <i>F</i>        | Significance <i>F</i> |
| Regression            | 1            | 309.0129482    | 309.0129482   | 107.7253        | 0.0091555             |
| Residual              | 2            | 5.737051793    | 2.868525896   |                 |                       |
| Total                 | 3            | 314.75         |               |                 |                       |
|                       | Coefficient  | Standard error | <i>t</i> Stat | <i>p</i> -Value |                       |
| Intercept             | -92.33466135 | 17.17892068    | -5.374881408  | 0.032915        |                       |
| SVM                   | 2.219123506  | 0.213807297    | 10.3790822    | 0.009156        |                       |

and assign positive and negative scores to it. Problems are encountered with the existing techniques for sentiment analysis in the presence of punctuations, ironical sentences, etc., which results in incoherent sentiment. In the proposed approach, every sentence is processed and examined at the sentence level to establish its polarity. Aspect-level sentiment analysis is used for finding emotions about things and their characteristics. The proposed algorithm NSASA does the preprocessing considering the negative stopwords and labels the sentiments of the article using SVM. The inclusion of negative stopwords in the proposed approach ensures that there will be minimum loss of significant information for labeling news articles. The proposed approach using SVM can be considered to be an improvement over Shirsat et al.'s approach (which has not considered the negative stopwords) to provide more accurate results. The proposed approach can be considered as an approach for sentiment analysis using negative stopwords to provide more accurate results for obtaining labeled positive, negative, and neutral sentiments from news articles. When the target classes are overlapping and the data set includes more noise, SVM does not perform very well. In the future, the proposed approach of using negative stopwords can be used to enhance the accuracy of the Naïve Bayes approach and other Machine Learning algorithms. Presently, the proposed approach has been tested on the news article dataset from BBC. This approach can be further applied to other datasets traditionally used in sentiment analysis, such as DUC-2002 and DUC-2004. The proposed approach is anticipated to be very helpful in determining the sentiment polarity of various news items and in retaining the information with the least amount of exclusion of important terms in the article. Apart from creating a sentiment analysis of news articles, the proposed approach can also benefit the governments in determining public opinion related to policies and program implementation expressed on various social networking platforms with better accuracy.



## References

1. MonkeyLearn. Sentiment analysis: A definitive guide, <https://monkeylearn.com/sentiment-analysis/>. Last accessed 5 Dec 2022
2. A. Zhao, Y. Yu, Knowledge-enabled BERT for aspect-based sentiment analysis. *Knowledge-Based Syst.* **227**, 107220 (2021) ISSN 0950-7051
3. K. Roebuck, *Sentiment Analysis: High-Impact Strategies What You Need to Now: Definitions, Techniques and Applications for Sentiment Analysis, Adoptions, Impact, Benefits, Maturity* (Emergo Publishing, 2012)
4. P. Pooja, G. Sharvari, A survey of sentiment classification techniques used for Indian regional languages. *Int. J. Comput. Sci. Appl.* **5**(2), 13–26 (2015)
5. A. Shoukry, Sentence-level Arabic sentiment analysis, in *2012 International Conference on Collaboration Technologies and Systems (CTS)*, (IEEE, 2012), pp. 546–550
6. P. Melville, W. Gryc, R.D. Lawrence, Sentiment analysis of blogs by combining lexical knowledge with text classification, in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (ACM, 2009), pp. 1275–1284
7. S. Urologin, Sentiment analysis, visualization and classification of summarized news articles: A novel approach. *Int. J. Adv. Comput. Sci. Appl.* **9**(8), 616–625 (2018)
8. S. Taj, B.B. Shaikh, A.F. Meghji, Sentiment analysis of news articles: A lexicon based approach, in *2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, (IEEE, 2019), pp. 1–5
9. G.S. Vilasrao, P.D. Sathya, Lexical approach for sentiment analysis on news articles with deep learning method. *Int. J. Sci. Res.* **8**(12), 725–730 (2019)
10. W. Souma, I. Vodenska, H. Aoyama, Enhanced news sentiment analysis using deep learning methods. *J. Comput. Soc. Sci.* **2**, 33–46 (2019)
11. V.S. Shirsat, R.S. Jagdale, S.N. Deshmukh, Sentence level sentiment identification and calculation from news articles using machine learning techniques, in *Computing, Communication and Signal Processing*, (Springer, 2019), pp. 371–376
12. Z. Li, H. Xie, G. Cheng, Q. Li, Word-level emotion distribution with two schemas for short text emotion classification. *Knowledge-Based Syst.*, Elsevier **227**, 107163 (2021)
13. M.E. Basiri, M. Abdar, M.A. Cifci, S. Nemati, U.R. Acharya, A novel method for sentiment classification of drug reviews using fusion of deep and machine learning techniques. *Knowledge-Based Syst.*, Elsevier **198**, 105949 (2020)
14. P. Tiwari, B.K. Mishra, S. Kumar, V. Kumar, Implementation of n-gram methodology for rotten tomatoes review dataset sentiment analysis. *Int. J. Knowl. Discovery Bioinf.*, IGI Global **7**(1), 30–41 (2017)
15. S. Soumya, K.V. Pramod, Sentiment analysis of Malayalam tweets using machine learning techniques. *ICT Express*, Elsevier **6**(4), 300–305 (2020)
16. G. Rao, W. Huang, Z. Feng, Q. Cong, LSTM with sentence representations for document-level sentiment classification. *Neurocomputing*, Elsevier **308**, 49–57 (2018)
17. R. Mishra, T. Gayen, Automatic lossless-summarization of news articles with abstract meaning representation. *Procedia Comput. Sci.*, Elsevier **135**, 178–185 (2018)
18. M. Birjali, M. Kasri, A. Beni-Hssane, A comprehensive survey on sentiment analysis: Approaches, challenges and trends. *Knowledge-Based Syst.* **226**, 107134 (2021) ISSN 0950-7051
19. P. Wang, J. Li, J. Hou, S2SAN: A sentence-to-sentence attention network for sentiment analysis of online reviews. *Decis. Support Syst.* **149**, 113603 (2021) ISSN 0167-9236
20. W. Liao, B. Zeng, J. Liu, P. Wei, X. Cheng, W. Zhang, Multi-level graph neural network for text sentiment analysis. *Comput. Electr. Eng.* **92**, 107096 (2021) ISSN 0045-7906
21. S. Zitnik, S. Blagus, M. Bajec, Target-level sentiment analysis for news articles. *Knowledge-Based Syst.* **249**, 108939 (2022) ISSN 0950-7051

# An Empirical Study to Assess the Factors Influencing Banking Customers Toward FINTECH Adoption in Tamil Nadu



R. Mary Metilda  and S. D. Shamini

## 1 Introduction

The financial industry is changing rapidly by offering different services with latest technologies to meet the expectations of the end users. The leading technology and the growth of financial services industry helps to build the strong economy with more of digitalization. Even, the rapid adoption helps to serve their customers in a competitive way. Fintech is denoted as a novelty enabled with innovative technologies to offer novel services to their customers by companies in the financial services sector [1].

The advent of Fintech in India is aimed to reduce the floating of liquid cash and to improve the digital transaction. Though Fintech established in India during 1990s, post 2000 becomes the market for the digital economy, especially the growth was higher during COVID pandemic [2]. The Fintech adoption in the world market is expected to rise up to 52% [3], and the industry is estimated to reach \$9.82 trillion in 2023 at 15.64% Compound Annual Growth Rate (CAGR) [4].

The flow of the research work is divided into seven modules, namely, Introduction to the study, Research Objectives, Literature Review, Research Methodology, Analysis & Interpretation, Conclusion, Limitation, and Future Research. A detailed description about the study and its proposed objectives are discussed in the first two modules. Previously published research works and its outcomes are presented in the third module of this report. The proposed statistical tools, applications, and its inferences are discussed in fourth and fifth modules of this study. The concluding

---

R. Mary Metilda (✉)

Department of Management Studies, Sri Ramakrishna Engineering College, Coimbatore, Tamil Nadu, India

S. D. Shamini

SNS College of Technology, Coimbatore, Tamil Nadu, India

remarks of the research work and its future scope are discussed and presented in the last two modules.

Robust financial services are the challenged outcomes of the financial institutions with backbone support from latest technologies [5]. The recent interaction between customer and a bank is the outcome of revolution took place in the Fintech industry and it is new for the current generation people. The older methodology of visiting a bank to do financial transaction is gone, and everything is done via online with the help of mobile applications, internet banking, etc. The customers' behavior toward Fintech usage is highly influenced by various factors such as ease of use, perceived risk, and convenience [6]. On the other hand, perceived usefulness is found as the highest stimulus variable with respect to Fintech adoption in Taiwan [7]. Many may be new to technologies, and if the customers perceive technologies as new path way, they may be influenced toward Fintech services [8–10]. The entries of more start-ups in the Fintech sector are witnessed after 2015 and are the main reason for the growth of Fintech in India. The Indian market witnessed the growth of Fintech services from \$247 million to \$1.5 billion during 2015. Further, the growth of mobile phone usage with high speed Internet connections has helped the Fintech market to grow with more phase [11]. Despite of latest technologies in the Fintech industry, it still serve the existing customers to reach traditional financial products/services [12].

### ***1.1 Objectives of the Study***

The scope to do the research on implications of Fintech in Indian market still has its own space. The quantum of research on Fintech usage among banking customers and factors influencing Fintech adoption in India are limited. Considering this gap and need, the current research is designed to assess the variables stimulus Fintech adoption by the customers of banks in Tamil Nadu. Further, this study extended its scope to assess the significant impact of identified variables on the Fintech adoption by the banking customers.

### ***1.2 Research Methodology***

The current research is designed to assess the variables stimulus Fintech adoption by the customers of banks in Tamil Nadu. A structured questionnaire was constructed to collect the opinion of banking customers from Tamil Nadu. The designed questionnaire was circulated to 200 respondents who were selected randomly and collected their opinion. Google form was used as a tool to collect the responses and the collected responses were analyzed with the help of statistical tools such as Descriptive Statistics, Exploratory Factor Analysis, Pearson Co-efficient of Correlation, and Simple Linear Regression. Further the research assumption was tested with the help of hypothesis framed and as given below

H1: The identified factors are significantly influencing the banking customers toward Fintech adoption.

## 2 Analysis and Interpretation

The demographic profiles of the respondents are analyzed using simple percentage analysis and are presented in Table 1 as given below.

A look at the demographic data presented in Table 1 shows that out of 200 respondents, 48.5% are male while the remaining 51.5% are female. In terms of age, the distribution of customers falls in the younger part with 55% in the 18–25 years age group. Only two customers are aged 55 and over. The survey of respondents' occupation shows that 42% of the respondents work in the private sectors, the second highest 30% followed as students. The annual income of the respondents showed that most of the respondents' income was between 6 and 10 lakh per year; 88% of the respondents, followed by 7% fall under the 5 lakh category. Data quality of customers shows that most of the respondents, that is, 54% are postgraduates, followed by 37% as graduates. Further, the analysis confirmed that 56% of the respondents do use Fintech services on daily basis and 23% of respondents use Fintech services once in a week.

## 3 Fintech and Their Attributes

This study identified 17 variables as the attributes of Fintech, and the variables are identified as the outcomes of the literature review [10]. The respondents were requested to share their opinion about the 17 variables or attributes on the Likert scale of 1–5; 1 be the lower response and 5 be the higher response. The data collected were analyzed using simple mean and standard deviation and the outcome of the analysis are shown in Table 2 as given below.

The data from Table 2 confirms that the mean value between 1.28 and 2.81, showing that bank customers have all these qualities in them to varying degrees in the lower part. The difference in their response is in the range of 0.50–1.24, which shows the consistency of the answers.

In order to understand the significant relationship between the variables chosen for the study, a null hypothesis is framed and is tested using chi-square test/Bartlett's test of sphericity. The result shown in Table 3 confirms that the Chi-square value is statistically significant at 1% level of significance and proved that the null hypothesis is rejected and alternate hypothesis is accepted. This signifies that the variables selected for the current research are correlated each other with statistical acceptance level. In addition, the correlation matrix showed in Table 4 indicates that

**Table 1** Demographic profile of the respondents

| S.No | Variable                      | Category           | No. of respondents | Percentage   |
|------|-------------------------------|--------------------|--------------------|--------------|
| 1    | Gender                        | Male               | 97                 | 48.5         |
|      |                               | Female             | 103                | 51.5         |
|      |                               | <i>Total</i>       | <i>200</i>         | <i>100.0</i> |
| 2    | Age                           | 18–25 years        | 110                | 55.0         |
|      |                               | 26–35 years        | 34                 | 17.0         |
|      |                               | 36–45 years        | 32                 | 16.0         |
|      |                               | 46–55 years        | 22                 | 11.0         |
|      |                               | More than 55 years | 2                  | 1.0          |
|      |                               | <i>Total</i>       | <i>200</i>         | <i>100.0</i> |
| 3    | Occupation                    | Student            | 60                 | 30.0         |
|      |                               | Govt. employee     | 6                  | 3.0          |
|      |                               | Private employee   | 84                 | 42.0         |
|      |                               | Self employed      | 32                 | 16.0         |
|      |                               | Home maker         | 18                 | 9.0          |
|      |                               | <i>Total</i>       | <i>200</i>         | <i>100.0</i> |
| 4    | Annual income                 | 1–5 lakh           | 14                 | 7.0          |
|      |                               | 6–10 lakh          | 176                | 88.0         |
|      |                               | More than 10 lakh  | 10                 | 5.0          |
|      |                               | <i>Total</i>       | <i>200</i>         | <i>100.0</i> |
| 5    | Marital status                | Married            | 86                 | 43.0         |
|      |                               | Unmarried          | 112                | 56.0         |
|      |                               | Divorced           | 2                  | 1.0          |
|      |                               | <i>Total</i>       | <i>200</i>         | <i>100.0</i> |
| 6    | Educational qualification     | Not a graduate     | 18                 | 9.0          |
|      |                               | Graduate           | 74                 | 37.0         |
|      |                               | Post graduate      | 108                | 54.0         |
|      |                               | <i>Total</i>       | <i>200</i>         | <i>100.0</i> |
| 7    | Frequency of using technology | Daily              | 112                | 56.0         |
|      |                               | Weekly             | 46                 | 23.0         |
|      |                               | Monthly            | 34                 | 17.0         |
|      |                               | Yearly             | 6                  | 3.0          |
|      |                               | Occasionally       | 2                  | 1.0          |
|      |                               | <i>Total</i>       | <i>200</i>         | <i>100.0</i> |

there is no higher correlation;  $R$  value is not greater than the standard/acceptable value of 0.8. Hence, it is proved the absence of multicollinearity in the structure.

Table 5 indicates the amount of variance (Communality) extracted by the Fintech attributes identified for the study. It has already been defined that the communalities should be greater than 0.5 to define a structure as valid one [13]. From the data, it is very clear all the extraction values are higher than 0.6 and confirmed the valid structure.

**Table 2** Fintech and their attributes

| S.No | Attribute   | Mean | SD    |
|------|---|------|-------|
| 1    | Convenient to work (F1)                                 | 1.45 | 0.681 |
| 2    | Convenient to work with latest electronic gadgets (F2)  | 1.35 | 0.512 |
| 3    | Paperless operation (F3)                                | 1.36 | 0.573 |
| 4    | Less working duration (F4)                              | 1.28 | 0.504 |
| 5    | Meet my requirements (F5)                               | 1.66 | 0.787 |
| 6    | Useful (F6)   | 1.37 | 0.545 |
| 7    | 24 * 7 Service (F7)                                     | 1.52 | 0.702 |
| 8    | Confidentiality in the personal information stored (F8) | 2.05 | 0.974 |
| 9    | Satisfied service mechanism (F9)                        | 1.89 | 0.803 |
| 10   | Maintains good will/reputation (F10)                    | 2.01 | 0.831 |
| 11   | Referred by neighbors (F11)                             | 2.33 | 1.024 |
| 12   | To get latest discounts/offers (F12)                    | 2.27 | 1.044 |
| 13   | Latest products and services (F13)                      | 2.13 | 0.943 |
| 14   | Threat of loss of money (F14)                           | 2.81 | 1.242 |
| 15   | Threat of system hacking (F15)                          | 2.66 | 1.011 |
| 16   | Affordable cost to access the service (F16)             | 1.99 | 0.863 |
| 17   | Less or no human interface (F17)                        | 2.16 | 0.973 |

( ) – Inside parenthesis are the variable labels

**Table 3** Chi-square test and measure of sampling adequacy

|  |                         |                    |
|--|-------------------------|--------------------|
| Kaiser-Meyer-Olkin (KMO) measure (sampling adequacy) |                         | 0.813              |
| Bartlett’s test of sphericity                        | Chi-square (approx.)    | 851.986            |
|  | Degrees of freedom (df) | 136                |
|  | Significance            | 0.000 <sup>a</sup> |

<sup>a</sup>Significant at 1% LoS

The Principal Component Analysis (PCA) was used to extract the identified 17 variables and found that the Eigen values of three variables are higher than the standard level of one. It is sufficient to have 50–60% of total variance explained by all the variables extracted by above said methods [14]. The outcome of the analysis is given in Table 6, and from the table, it is understood that 61.078% of cumulative variance are extracted with the help of three factors identified.

### 3.1 Reliability Level and Grouping of Variables

Reliability of the collected respondents’ opinion (data) is to be checked before proceeding further analysis. Hence, Cronbach’s alpha was applied on the 17 variables constituted under three factors, and the results are 0.882, 0.835, and 0.755. The values of Cronbach’s alpha are higher than the standard value of 0.6 and hence proved the internal consistency of the questionnaire/collected data [15]. The factor

**Table 4** Degree of relationship between the variables

| Var | F1   | F2   | F3   | F4   | F5   | F6   | F7   | F8   | F9   | F10  | F11  | F12   | F13  | F14  | F15  | F16  | F17 |
|-----|------|------|------|------|------|------|------|------|------|------|------|-------|------|------|------|------|-----|
| F1  | 1.0  |      |      |      |      |      |      |      |      |      |      |       |      |      |      |      |     |
| F2  | 0.61 | 1.0  |      |      |      |      |      |      |      |      |      |       |      |      |      |      |     |
| F3  | 0.52 | 0.63 | 1.0  |      |      |      |      |      |      |      |      |       |      |      |      |      |     |
| F4  | 0.64 | 0.64 | 0.63 | 1.0  |      |      |      |      |      |      |      |       |      |      |      |      |     |
| F5  | 0.42 | 0.41 | 0.34 | 0.44 | 1.0  |      |      |      |      |      |      |       |      |      |      |      |     |
| F6  | 0.60 | 0.52 | 0.42 | 0.73 | 0.50 | 1.0  |      |      |      |      |      |       |      |      |      |      |     |
| F7  | 0.51 | 0.53 | 0.34 | 0.53 | 0.52 | 0.59 | 1.0  |      |      |      |      |       |      |      |      |      |     |
| F8  | 0.22 | 0.33 | 0.11 | 0.33 | 0.33 | 0.31 | 0.33 | 1.0  |      |      |      |       |      |      |      |      |     |
| F9  | 0.22 | 0.32 | 0.14 | 0.32 | 0.31 | 0.34 | 0.39 | 0.66 | 1.0  |      |      |       |      |      |      |      |     |
| F10 | 0.33 | 0.44 | 0.23 | 0.31 | 0.34 | 0.32 | 0.37 | 0.52 | 0.64 | 1.0  |      |       |      |      |      |      |     |
| F11 | 0.13 | 0.13 | 0.11 | 0.10 | 0.12 | 0.13 | 0.21 | 0.24 | 0.33 | 0.22 | 1.0  |       |      |      |      |      |     |
| F12 | 0.22 | 0.20 | 0.11 | 0.24 | 0.33 | 0.33 | 0.34 | 0.30 | 0.25 | 0.32 | 0.38 | 1.0   |      |      |      |      |     |
| F13 | 0.33 | 0.33 | 0.22 | 0.22 | 0.44 | 0.32 | 0.38 | 0.31 | 0.44 | 0.43 | 0.41 | 0.482 | 1.0  |      |      |      |     |
| F14 | 0.11 | 0.13 | 0.10 | 0.14 | 0.21 | 0.11 | 0.28 | 0.41 | 0.11 | 0.14 | 0.24 | 0.43  | 0.33 | 1.0  |      |      |     |
| F15 | 0.22 | 0.21 | 0.10 | 0.31 | 0.41 | 0.09 | 0.23 | 0.13 | 0.32 | 0.18 | 0.33 | 0.44  | 0.53 | 0.12 | 1.0  |      |     |
| F16 | 0.21 | 0.14 | 0.32 | 0.42 | 0.10 | 0.17 | 0.31 | 0.22 | 0.23 | 0.10 | 0.42 | 0.33  | 0.24 | 0.53 | 0.13 | 1.0  |     |
| F17 | 0.11 | 0.23 | 0.33 | 0.40 | 0.21 | 0.14 | 0.21 | 0.33 | 0.23 | 0.21 | 0.22 | 0.33  | 0.44 | 0.44 | 0.23 | 0.42 | 1.0 |

**Table 5** Communalities

| S.No | Attributes  | Initial | Extraction |
|------|---|---------|------------|
| 1    | Convenient to work (F1)                                 | 1.00    | 0.669      |
| 2    | Convenient to work with latest electronic gadgets (F2)  | 1.00    | 0.670      |
| 3    | Paperless operation (F3)                                | 1.00    | 0.661      |
| 4    | Less working duration (F4)                              | 1.00    | 0.757      |
| 5    | Meet my requirements (F5)                               | 1.00    | 0.586      |
| 6    | Useful (F6)   | 1.00    | 0.678      |
| 7    | 24 * 7 Service (F7)                                     | 1.00    | 0.549      |
| 8    | Confidentiality in the personal information stored (F8) | 1.00    | 0.636      |
| 9    | Satisfied service mechanism (F9)                        | 1.00    | 0.741      |
| 10   | Maintains good will/reputation (F10)                    | 1.00    | 0.535      |
| 11   | Referred by neighbors (F11)                             | 1.00    | 0.502      |
| 12   | To get latest discounts/offers (F12)                    | 1.00    | 0.517      |
| 13   | Latest products and services (F13)                      | 1.00    | 0.547      |
| 14   | Threat of loss of money (F14)                           | 1.00    | 0.537      |
| 15   | Threat of system hacking (F15)                          | 1.00    | 0.646      |
| 16   | Affordable cost to access the service (F16)             | 1.00    | 0.738      |
| 17   | Less or no human interface (F17)                        | 1.00    | 0.521      |

( ) – Inside parenthesis are the variable labels

**Table 6** Outcomes of PCA

| Component | Eigen values/percentage variance explained |                         |                                    |
|-----------|--|-------------------------|------------------------------------|
|           | Total                                      | Variance explained in % | Cumulative variance explained in % |
| 1         | 4.315                                      | 25.313                  | 25.313                             |
| 2         | 3.668                                      | 21.627                  | 46.940                             |
| 3         | 2.415                                      | 14.138                  | 61.078                             |

loadings of 17 variables are normalized in order to determine the influence of the variables in determining the factor structure. The variance is squared and the squared loadings are taken in to consideration, as factor loadings is the correlation between factors and the variables.

The significant variation for the factors was considered and named after the deviation noted for the variables. The values of the rotated component matrix for the 17 variables are presented in Table 7, and the values are the correlation between the first factor and the variable. All the factor loadings are higher than the standard value of 0.5, with the maximum value as 0.865 and minimum factor loading as 0.560.

The factor loadings for the 17 variables are identified and presented in Table 8 with the factors marked. It is understood from the factor loadings that 24.32% of the variation is explained by the factor Conducive, 23.64% of the variation is explained by the factor Adaptability, and 13.18% of the variation is explained by the factor Security, cumulatively the variation explained by all three factors reaching 61.14%.



**Table 7** Rotated component matrix

| S.No | Attributes  | Component |       |       |
|------|---|-----------|-------|-------|
|      |   | 1         | 2     | 3     |
| 1    | Less working duration (F4)                              | 0.825     |       |       |
| 2    | Paperless operation (F3)                                | 0.817     |       |       |
| 3    | Convenient to work (F1)                                 | 0.783     |       |       |
| 4    | Convenient to work with latest electronic gadgets (F2)  | 0.766     |       |       |
| 5    | Useful (F6)   | 0.765     |       |       |
| 6    | 24 * 7 Service (F7)                                     | 0.607     |       |       |
| 7    | Meet my requirements (F5)                               | 0.595     |       |       |
| 8    | Satisfied service mechanism (F9)                        |           | 0.865 |       |
| 9    | Confidentiality in the personal information stored (F8) |           | 0.798 |       |
| 10   | Maintains good will/reputation (F10)                    |           | 0.712 |       |
| 11   | Affordable cost to access the service (F16)             |           | 0.714 |       |
| 12   | Latest products and services (F13)                      |           | 0.598 |       |
| 13   | To get latest discounts/offers (F12)                    |           | 0.560 |       |
| 14   | Referred by neighbors (F11)                             |           | 0.564 |       |
| 15   | Threat of system hacking (F15)                          |           |       | 0.824 |
| 16   | Threat of loss of money (F14)                           |           |       | 0.843 |
| 17   | Less or no human interface (F17)                        |           |       | 0.684 |

\*PCA method of extraction

**Table 8** Identification of factors

| S.No | Factors      | Attributes  | Factor loadings |
|------|--------------|---|-----------------|
| 1    | Conducive    | Less working duration (F4)                              | 0.825           |
| 2    |              | Paperless operation (F3)                                | 0.817           |
| 3    |              | Convenient to work (F1)                                 | 0.783           |
| 4    |              | Convenient to work with latest electronic gadgets (F2)  | 0.766           |
| 5    |              | Useful (F6)   | 0.765           |
| 6    |              | 24 * 7 Service (F7)                                     | 0.607           |
| 7    |              | Meet my requirements (F5)                               | 0.595           |
| 8    | Adaptability | Satisfied service mechanism (F9)                        | 0.865           |
| 9    |              | Confidentiality in the personal information stored (F8) | 0.798           |
| 10   |              | Maintains good will/reputation (F10)                    | 0.712           |
| 11   |              | Affordable cost to access the service (F16)             | 0.714           |
| 12   |              | Latest products and services (F13)                      | 0.598           |
| 13   |              | To get latest discounts/offers (F12)                    | 0.560           |
| 14   |              | Referred by neighbors (F11)                             | 0.564           |
| 15   | Security     | Threat of system hacking (F15)                          | 0.824           |
| 16   |              | Threat of loss of money (F14)                           | 0.843           |
| 17   |              | Less or no human interface (F17)                        | 0.684           |

### 3.2 Influence of Factors on Fintech Usage

The significant influence of all the three factors, namely, Conducive, Adaptability, and Security on the customers’ attitude toward Fintech is tested with the help of simple regression. The factors such as Conducive, Adaptability, and Security are considered as independent variables while the customers’ Fintech adoption is considered as the dependent variable. The regression summary, ANOVA, and the coefficient tables are presented in Tables 9, 10, and 11, respectively, as given below. The regression summary table confirms that the values of  $R$  and  $R^2$  are higher than the standard value of 0.6, and further the Durbin-Watson test reveals the model fit as the data is less than 2. Hence, the regression model considered as fit and well defined. From the ANOVA table (Table 10), it is understood that the value of  $F$  statistics is 4.594, and it is statistically significant at 1% level of significance as indicated by the significance value. Hence, the validity of the model is proved and proceeds further to assess the coefficient values.

The values of coefficients are presented in Table 11 for all three factors along with the constant value identified from the regression analysis. The  $t$  test value and its significance confirmed that significant influence on Fintech adoption was

**Table 9** Regression summary for the usage of Fintech services

| $R$   | $R^2$ | Adjusted $R^2$ | S.E.  | Durbin-Watson |
|-------|-------|----------------|-------|---------------|
| 0.870 | 0.759 | 0.759          | 0.462 | 0.794         |

**Table 10** ANOVA

| Model      | Sum of squares (SoS) | Degrees of freedom (df) | Mean square | $F$   | Sig.               |
|------------|----------------------|-------------------------|-------------|-------|--------------------|
| Regression | 6.839                | 6                       | 2.343       | 4.594 | 0.003 <sup>a</sup> |
| Residual   | 43.524               | 193                     | 0.545       |       |                    |
| Total      | 50.363               | 199                     |             |       |                    |

<sup>a</sup>Significant at 1% LoS

**Table 11** Regression summary – Fintech usage & Fintech factors

| Model                   | Unstandardized coefficients |            | Standardized coefficients | $T$    | Sign               | 95% confidence level |             |
|-------------------------|-----------------------------|------------|---------------------------|--------|--------------------|----------------------|-------------|
|                         | $B$                         | Std. error | Beta                      |        |                    | Lower bound          | Upper bound |
| Constant                | 0.867                       | 0.343      |                           | 2.642  | 0.005 <sup>a</sup> | 0.2284               | 1.524       |
| Factor 1 (Conducive)    | 0.485                       | 0.245      | 0.364                     | 2.624  | 0.005              | 0.167                | 0.884       |
| Factor 2 (Adaptability) | 0.094                       | 0.182      | 0.084                     | 0.585  | 0.652              | -0.184               | 0.351       |
| Factor 3 (Security)     | -0.024                      | 0.121      | -0.021                    | -0.124 | 0.814              | -0.169               | 0.184       |

<sup>a</sup>Significant at 1% LoS

observed in a variable, namely, Conducive, and other two factors failed to influence significantly as the  $t$  test values are not significant either at 1% or 5% level of significance.

The regression equation can be formatted as:

$$Y = 0.867 + 0.485 \times X$$

where  $Y$  is the Fintech usage,  $X$  is the Conducive factor.

Thus by concluding that the identified factor (Conducive) is significantly influencing the Fintech usage of the banking customers from Tamil Nadu, and the other two factors are not influencing significantly.

## 4 Conclusion

This paper examines the factors influencing the banking customers' in adopting Fintech services. The Fintech attributes and the customers opinion about Fintech adoption were collected using the framed variables (17 No's) constructed under three factors, namely, Conducive, Adaptability, and Security. The reliability of the framed questionnaire was tested using Cronbach Alpha and found satisfactory. Further, the Exploratory Factor Analysis confirmed the total variance explained about 61.14% constituting, 24.32% of the variation from Conducive factor, 23.64% of the variation from Adaptability factor, and 13.18% of the variation from Security factor. Further, it is found that the customer wants to do banking transactions in a convenient way in a short time without going to the bank and is confident about the services offered by the bank. The regression analysis proved that out of three factors identified for this study, Conducive has significantly influencing the banking customers toward Fintech adoption, whereas, the other factors, namely, Adaptability and Security has not influencing the customers significantly toward Fintech adoption. Overall, these results can be seen as supporting additions to existing research pertaining to Fintech adoption by the banking customers.

### 4.1 Limitations of the Study

The data collection of this study is restricted and limited to Tamil Nadu only. An exclusive study can be done on various factors affecting the use of Fintech among banking customers. This study can be extended further by analyzing the influence of customers' demography, intention to adopt, knowledge level of digital transaction, and their attitude toward using Fintech.

## References

1. Financial Stability Board: Financial stability implications from FinTech: Supervisory and regulatory issues that merit authorities' attention, Retrieved from <http://www.fsb.org/wp-content/uploads/R270617.pdf> (2017)
2. C.T. Huei, L.S. Cheng, L.C. Seong, A.A. Khin, R.L.L. Bin, Preliminary study on consumer attitude towards FinTech products and services in Malaysia. *Int. J. Eng. Technol.* **7**, 166–169 (2018)
3. EY: EY FinTech adoption index (EY), pp. 1–44, Retrieved from <http://www.ey.com/GL/en/Industries/Financial-Services/ey-fintechadoption-20index4> (2016)
4. PWC: PWC report on “Emerging technology disrupting the financial sector”, pp. 1–56, Retrieved from <https://www.pwc.in/fintech> (2019)
5. R. Bates, Banking on future an exploration of Fintech and the consumer interest. Investopedia, Retrieved from <http://www.investopedia.com/uk/> (2017)
6. F.D. Davis, R.P. Bagozzi, P.R. Warshaw, User acceptance of computer technology: A comparison of two theoretical models. *Manag. Sci.* **35**(8), 982–1003 (1989)
7. L.M. Chuang, C.C. Liu, H.K. Kao, The adoption of Fintech service: TAM perspective. *Int. J. Manag. Adm. Sci.* **3**(7), 1–15 (2016)
8. C.L. Hsu, J.C.C. Lin, Effect of perceived value and social influences on mobile app stickiness and in-app purchase intention. *Technol. Forecast. Soc. Chang.* **108**, 42–53 (2016)
9. V. Venkatesh, F.D. Davis, A theoretical extension of the technology acceptance model: Four longitudinal field studies. *Manag. Sci.* **46**(2), 186–204 (2000)
10. T.H. Cham, L.C. Seong, S.C. Low, A.A. Khin, Preliminary study on consumer attitude towards Fintech products and services in Malaysia, Retrieved from <https://www.researchgate.net/publication/325779653> (2018)
11. KPMG: The pulse of Fintech, in Fintech in India – A global growth story, [KPMG.com.in](http://KPMG.com.in) (2016)
12. Z. Hu, S. Ding, S. Li, L.C.S. Yang, Adoption intention of Fintech services for Bank users: An empirical examination with an extended technology acceptance model. *Symmetry*, [www.mdpi.com/journal/symmetry](http://www.mdpi.com/journal/symmetry) (2019)
13. A. Field, *Discovering Statistics Using SPSS for Windows* (Sage Publications, London/Thousand Oaks/New Delhi, 2000)
14. J. Hair, R.E. Anderson, R.L. Tatham, W.C. Black, *Multivariate Data Analysis*, 4th edn. (Prentice-Hall Inc., New Jersey, 1995)
15. J.C. Nunnally, *Psychometric Theory*, 2nd edn. (McGraw-Hill, New York, 1978)

# Driver's Drowsiness Detection Using SpO<sub>2</sub>



P. Sugantha Priyadharshini , N. Jayakiruba , A. D. Janani ,  
and A. R. Harini 

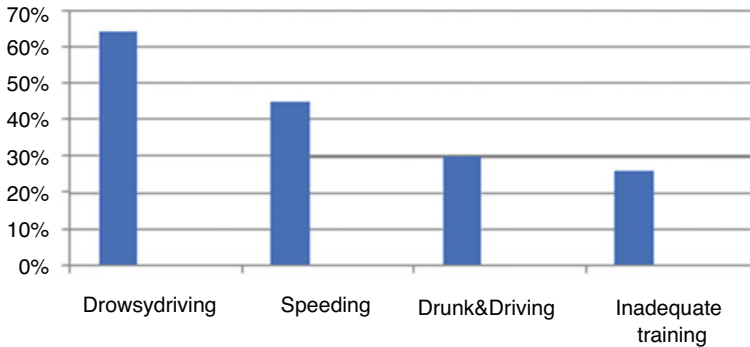
## 1 Introduction

Drowsiness is tiredness that expresses as heavy eyelids, yawning, daydreaming, rubbing of the eyes, and a lack of focus. We primarily focused on oxygen saturation levels in our study. Long-distance drivers who do not even take regular rests may become tired and make mistakes like vehicle accidents and other unfortunate incidents. Accidents and deaths are happening more frequently, which poses a serious threat to everyone in the world. Therefore, it is crucial to develop a driver's drowsiness detection system that really can alert the driver of their drowsiness and inattentiveness.

Driver drowsiness detection is a car safety technology that guards against accidents brought on by drowsy driving. According to numerous studies, drowsiness may be a factor in up to 50% of accidents on such roads, or about 20% of all accidents. Numerous investigations have discovered that driving can be affected by lack of sleep just as much as alcoholism. According to research, driving after 17–18 h of non-stop driving while awake is just as dangerous as doing so when impaired by alcohol legal limit in many European nations is 0.05%. Driving while impaired by blood alcohol content (BAC) is extremely risky than sleep deprivation. It has been established that sleep deprivation harms driving performance, particularly in four areas: coordination, longer reaction times, impaired judgment, memory, and ability to retain information. Human SpO<sub>2</sub> levels typically range from 95% to 100%. An individual's oxygen level will be low whenever they feel sleepy. As a result, it is specified in our article that the alarm will sound when a person's oxygen

---

P. Sugantha Priyadharshini (✉) · N. Jayakiruba · A. D. Janani · A. R. Harini  
Sri Ramakrishna Engineering College, Coimbatore, Tamil Nadu, India  
e-mail: [sugantha.ravee@srec.ac.in](mailto:sugantha.ravee@srec.ac.in); [jayakiruba.2001058@srec.ac.in](mailto:jayakiruba.2001058@srec.ac.in); [janani.2001057@srec.ac.in](mailto:janani.2001057@srec.ac.in);  
[harini.2001049@srec.ac.in](mailto:harini.2001049@srec.ac.in)



**Fig. 1** Major reasons for road accidents

level is less than 96%. In order for the vehicle to quickly regain awareness and drive safely, we developed the concept of a driver drowsiness detection system that alerts the driver when feeling tired to solve this issue. People can fall asleep at any time, so it is imperative to have a real-time drowsiness detection gadget. The integration of the Arduino board in this device is mainly to transmit the input data obtained and process the data to obtain the outcome. It binds the connectivity between input, output, and the sensor SpO<sub>2</sub>. Technology or system such as detection and monitoring systems for driver fatigue is essential for reducing road accidents. As the drivers face problems and difficulty in detecting fatigue, the vehicle must be installed with a fatigue detection and monitoring system.

Figure 1 depicts the different reasons of occurring road accidents. It is shown in graphical representation with percentile values.

## 2 Literature Review

A driver who is drowsy and experiencing micro sleep is probably considerably more dangerous on the road than a driver who is driving too fast. Automakers and researchers are working to find technical solutions to this issue to prevent a crisis of this magnitude. In this study, they emphasize neural network-based techniques for the identification of such micro sleep and sleepiness [1]. To detect the same, their earlier research in this area used machine learning and multi-layer perceptrons. In this study, the convolutional neural network (CNN) was used to classify tiredness, and accuracy was improved by using facial landmarks that the camera detects and sends to the network.

Road accidents, which are now nearly a daily issue in the modern world, result in significant loss of life and property and harm the economy. Drunk driving is a major factor in these collisions. To prevent these unwanted events, it is crucial to identify drowsy drivers as soon as possible. In this work, two physiological parameters—

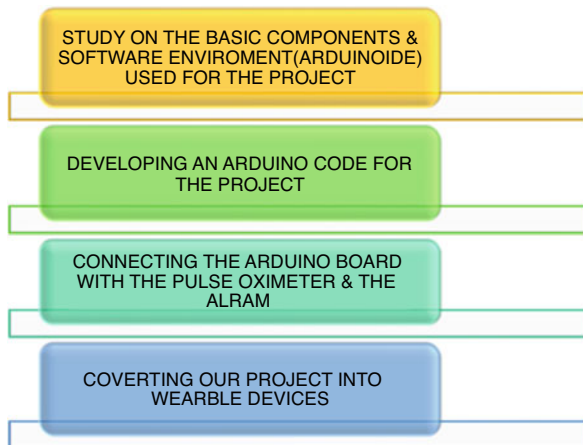
heart rate and eye-blink rate—are continuously analyzed to provide a very reliable and cost-effective method for determining if a driver is sleepy or weary [2]. This system also has a reliable alarm function [3]. The system identifies the driver’s tired or sleepy condition and immediately informs him to restore consciousness if the threshold values of the drowsiness checking parameters are surpassed [4].

The driver face monitoring system can identify driver distraction and fatigue in real time using machine vision techniques. This research presents the method for detecting driver hypervigilance (fatigue and distraction) based on symptoms in the face and eye areas [5]. This technique extracts hypervigilance symptoms from the face and eye, respectively, by horizontally projecting the top-half portion of the face image and matching face templates. A distraction that is removed from the facial region can be detected by the head-turning. Eyelid distance alterations relative to the usual eyelid distance, eye closure rate, and proportion of closed eyes are the retrieved symptoms from the eye area.

### 3 Objective of the Work

The objective of this research is to create a device that can detect drowsiness to lower accident rates in our nation [6]. The tool analyzes a driver’s blood saturation level, which assists in identifying the driver’s drowsiness and alerting him so that he is awake and aware while driving [7]. This tool will be incredibly useful because it biologically recognizes drowsiness. The goal of the research is to lower the country’s fatality rate and to notify the exhausted driver to take a break, which avoids road accidents [8]. It can be achieved by using a SpO2 sensor to detect tiredness because it gives an efficient and stable outcome, and it is also cost-efficient. Figure 2 depicts the workflow of the research [9].

Fig. 2 Workflow



## 4 System Analysis

### 4.1 Existing System

The existing system developed for driver drowsiness detection uses image processing for detecting whether the driver was feeling fatigued and sleepy. In this system, they analyze the drowsiness from the eyes of the person [10].

It detects how much time the eyes are closed on the driver. If the eyes are closed for longer than 20 s, the speaker included in the system will sound an alert, thus alerting the driver, waking him up, and preventing an accident.

### 4.2 Proposed System

The expected outcome of our research will be like the device will be wearable and continuously monitor the blood oxygen level once worn in the hand. When the oxygen level of the person goes below the threshold point, the device will give an alarm to the driver that he/she is drowsy and need to be active. Though there are several methods for measuring drowsiness, this approach is completely non-intrusive and does not affect the driver in anyway.

The existing systems are not accurate and reliable, but this research is very accurate and reliable because of using a biological method for the calculation and also cost affordable and easy to wear. Most of the previous research works, which are related to the driver's drowsiness, are all about sensing the eye blink, and a more complicated heart rate, but in our work, the oxygen level in the blood is sensed. If it is less than 96%, our work will produce an alert to the driver, which will awaken his consciousness.

Figure 3 shows the system architecture of the work. It explains the interconnection of the components and the steps involved in the detection of drowsiness.

## 5 Hardware and Software Requirements

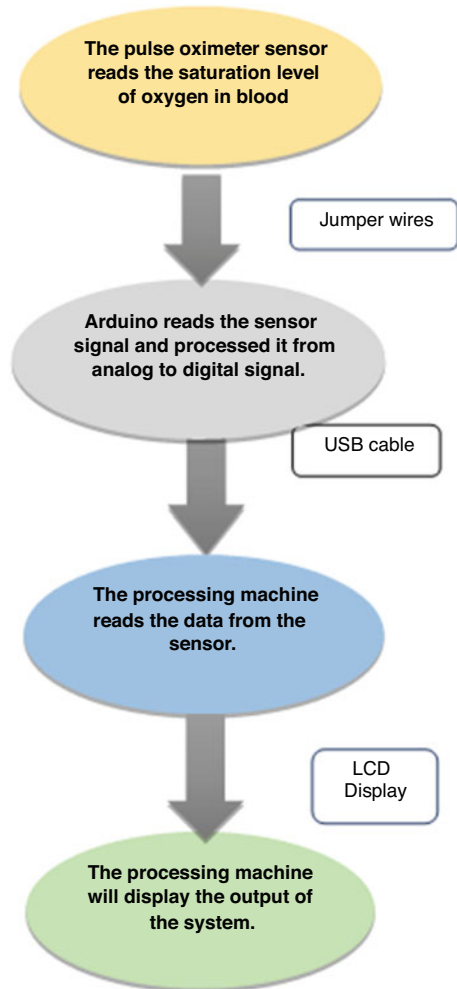
The hardware requirements include a SpO2 sensor, LCD Display, Alarm buzzer, Resistor, Transistor, Breadboard and also jumper wires, and software requirements like Arduino IDE and some of the libraries to be used for our work.

### 5.1 SpO2 Sensor

The amount of oxygen-carrying hemoglobin in the blood about the amount of hemoglobin that does not carry oxygen is measured by SpO2, also referred to as oxygen saturation. A specific amount of oxygen must be present in the blood for the



Fig. 3 System architecture



body to function properly. Figure 4 shows the SpO2 sensor, which senses the blood oxygen level.

### 5.2 LCD Display

LCD (Liquid Crystal Display) is a type of flat panel display which uses liquid crystals in its primary form of operation. LEDs have a large and varying set of use cases for consumers and businesses. Figure 4 depicts the LCD display where the level of the blood oxygen will be displayed.

**Fig. 4** SpO2 sensor



### 5.3 Alarm

Distinctive sound is to help attract attention to the computer or hardware device. A buzzer depicted in Fig. 5 is an audio signaling device, which may be mechanical, electromechanical, or piezoelectric (piezo for short). Typical uses of buzzers and beepers include alarm devices, timers, training, and confirmation of user input such as a mouse click or keystroke (Fig. 6).

### 5.4 ArduinoIDE

The Arduino Integrated Development Environment—or Arduino Software (IDE)—contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions, and a series of menus (Fig. 7). It connects to the Arduino hardware to upload programs and communicate with them.

**Fig. 5** LCD display



**Fig. 6** Alarm buzzer



**Fig. 7** Arduino IDE logo

## 6 Implementation and Results

The driver's wrist will host the wearable device. To prevent accidents, the driver must wear this drowsiness detection gadget after the vehicle is started in the driveway. The gadget includes an anti-sleep alarm and pulse oximetry, which are connected to an Arduino Uno R3 attached to the device to rouse the sleepy motorist. The sensor for pulse oximetry detects the driver's heart rate and SpO2 level (blood oxygen saturation). If the heart rate falls below a certain level and when SpO2 falls to 96% of oxygen at 60 bpm, the active alarm buzzer emits a few passive internal shocks and sound vibrations, which are used to shake the motorist out of sleep. First, the system's code implementation is completed, and the circuit connection with the necessary parts is created. The code is uploaded to the Arduino board after proper connection, the operation of the sensor has been verified, and the system is now implemented. This research work is certain to provide the best result in comparison to the existing research. Utilizing this device, we can lessen accidents in daily life and preserve valuable life.

## 7 Experimental Results

The experiment's objectives are to determine the SpO2 level in the body's blood and to determine whether the apparatus is operating correctly. The outcomes will show the driver's condition, and whether or not he or she is active while driving. The states of the driver are identified from the experiment, and if the driver is not active, the alert sounds. These findings are crucial to fulfilling the project's requirements and enhancing the caliber and effectiveness of the research. We propose a novel system for evaluating the driver's level of consciousness based on SpO2 level (blood saturation level) detection in the human body. We use the experimental results to track the drowsiness level of the driver. Therefore, this research work is almost a real-time system as it has a high operation speed. From the experimental results, the driver's drowsiness detection system using SpO2 applies to different circumstances and can offer stable performance.

Figure 8 illustrates the working principle of pulse oximeter sensor, detection of blood level and triggering of an alarm that meets the desired level of SpO2.

The experimental detection of blood oxygen level is depicted in Fig. 9.

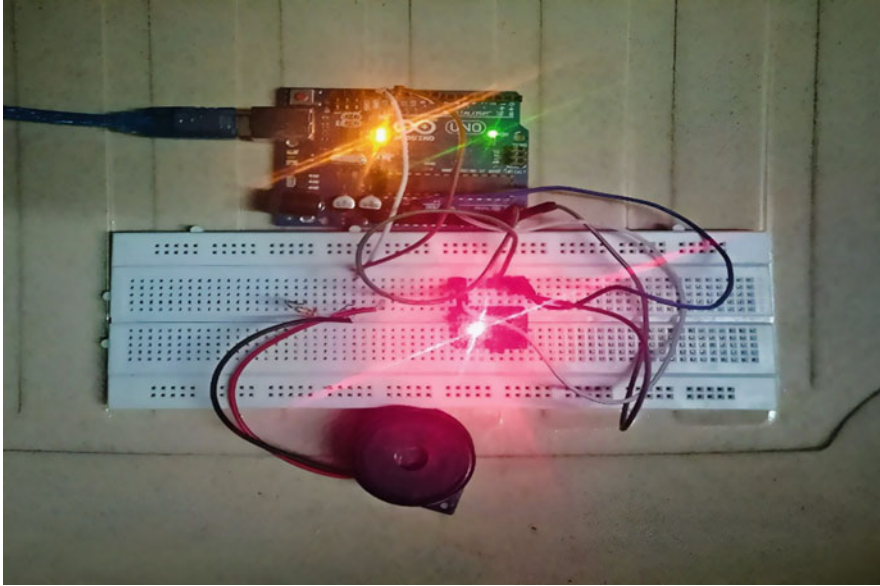


Fig. 8 Working of the pulse oximeter sensor

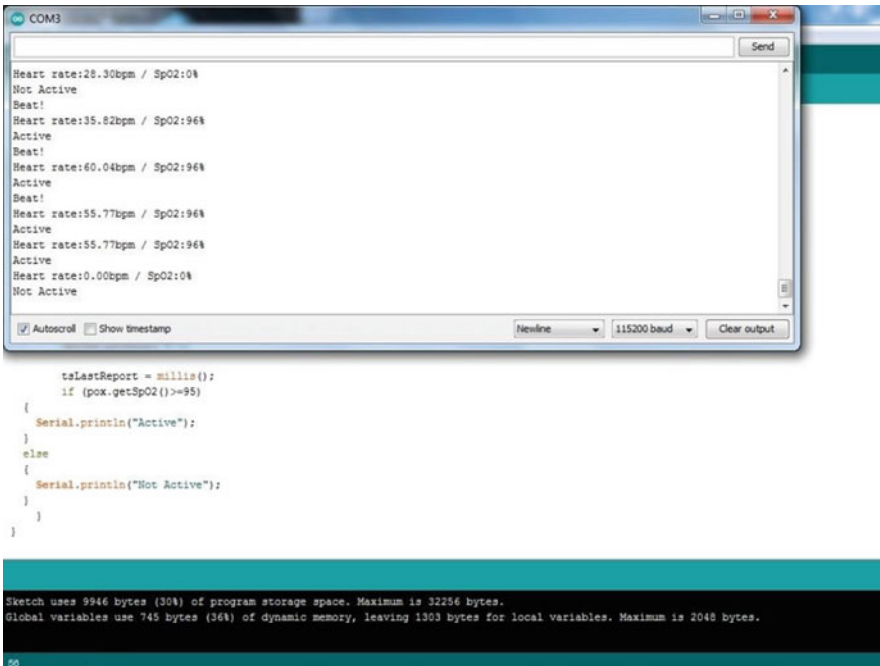


Fig. 9 Console window

## 8 Conclusion and Future Work

The purpose of this research is to find effective methods for handling situations where a driver's drowsiness is suspected. The analysis leads to the conclusion that in other existing research, they have incorporated the domain, of image processing which was a bit complicated process, and also there are fewer chances to convert their model into wearable devices. Our drowsiness detection system focused on detecting the oxygen saturation level in the blood, which helps to detect the drowsiness of the driver.

Our future work is to convert the prototype into a wearable device that will be useful for commercial purposes. From all the above experiments, it is proven that the developed system using a pulse oximeter sensor can detect the driver's drowsiness condition. The device successfully detects the drowsiness level of the driver and produces good results. However, the implementation of our work can be extended with some more physiological sensors like heart rate and monitors, but still, it is effective research with a success rate. Despite various results from various domains like image processing and hybrid-based systems, this biological method is reliable, affordable, and easy to use and helps us in detecting drowsiness level more accurately.

## References

1. R. Jabbar et al., Driver drowsiness detection model using convolutional neural networks techniques for android application. arXiv:2002.03728v1, 1–6 (2020)
2. S.H. Alam et al., A cost-effective driver drowsiness recognition systems, in *2019 IEEE International Conference on Biomedical Engineering, Computer and Information Technology for Health (BECITHCON)*, (IEEE, 2020)
3. A.Y. Avidan, Chapter 101: Sleep and its disorders, in *Bradley and Daroff's Neurology in Clinical Practice*, ed. by J. Jankovic, J.C. Mazziotta, S.L. Pomeroy, N.J. Newman, 8th edn., (Elsevier, Philadelphia, 2022)
4. M. Doudou, A. Bouabdallah, V. Berge-Cherfaoui, Driver drowsiness measurement technologies. *Int. J. Intell. Transp. Syst. Res.* **18**(2), 297–319 (2020)
5. J. Batista, A drowsiness and point of the attendance monitoring system for driver vigilance, in *Proceedings of Intelligent Transportation Systems Conference*, Seattle, WA, USA (October 2007), pp. 702–708
6. G. Li, B.-L. Lee, W.-Y. Chung, Smartwatch-based wearable EEG system for driver drowsiness detection. *IEEE Sens. J.* **15**(12), 7169–7180 (2015)
7. R. Sasikala, S. Suresh, J. Chandramohan, M. Valanraj Kumar, Driver drowsiness detection system using image processing technique by the human visual system. *Int. J. Emerg. Technol. Eng. Res.* **6**(6), 1–11 (2018)
8. J. He, W. Choi, Y. Yang, J. Lu, X. Wu, K. Peng, Detection of driver drowsiness using wearable devices: A feasibility study of the proximity sensor. *Appl. Ergon.* **65**, 473–480 (2017)
9. J.N. Mindoro, C.D. Casuat, A.S. Alon, M.A.F. Malbog, J.A.B. Susa, Drowsy or not? Early drowsiness detection utilizing Arduino based on electroencephalogram (EEG) neuro-signal. *Int. J. Adv. Trends Comput. Sci. Eng.* **9**(2), 2221 (2020)
10. B. Warwick, N. Symons, X. Chen, K. Xiong, Detecting driver drowsiness using wireless wearables, in *2015 IEEE 12th International Conference on Mobile Ad Hoc and Sensor Systems*, (IEEE, 2015), pp. 585–588

# A Blockchain Framework for Investment Authorities to Manage Assets and Funds



P. C. Sherimon , Vinu Sherimon , Jeff Thomas , and Kevin Jaimon 

## 1 Introduction

State General Reserve Fund (SGRF) of Ministry of Finance (MoF) was established in 1980 under the Royal Decree 1/1980 [1] to manage the investment and financial matters on behalf of Government of Sultanate of Oman. It is the wealth deposit of the Sultanate of Oman. It is the body who manages and invests in public and private markets from the extra capital attained from oil and gas profits. Unlike other funds in Oman, this body mainly concentrates on investments outside the Sultanate. The IT department of OIA manages the development of the Information Systems used in the various departments. The targeted sectors in private investments include health sector, mining industry, food industries, logistics, and other diverse sources. Later in 1998, Tanmia, the Oman National Investment company was established [1].

In 2004, the company started investing in real estate, the first being the procurement of Regent Wharf property in London, UK [1]. The first direct investment in Private equity took place in 2006, with the acquisition of 6.1% of shares of AWAS, an Irish aviation company [1]. The investment of Oman in Vietnam happened with the establishment of Vietnam Oman Investment (VOI) in 2008 [1]. Later, a joint venture was established in the year 2010 with Uzbekistan, by the founding of

---

P. C. Sherimon  
Faculty of Computer Studies, Arab Open University, Muscat, Sultanate of Oman  
e-mail: [sherimon@aou.edu.om](mailto:sherimon@aou.edu.om)

V. Sherimon (✉)  
Department of IT, University of Technology and Applied Sciences, Muscat, Sultanate of Oman

J. Thomas · K. Jaimon  
Department of Computer Science and Engineering, Saintgits College of Engineering,  
Pathamuttam, Kottayam, Kerala, India

Uzbek Oman Investment Joint Venture (UOI) [1]. Oman India Joint Investment Fund (OIJIF) was established in the year 2011 [1].

SGRF invests in more than 35 countries worldwide and allocates funds between 65–85% in public market shares and between 35–15% in private market shares [1]. The working of SGRF involves large number of transactions, which requires high level of transparency and governance between the involved stakeholders. In 2020, through a Royal Decree, SGRF have been transferred to Oman Investment Authority (OIA).

Generally transacted funds are recorded and tracked in the ledgers of the respective parties involved. To manage and allocate funds, organizations operate through various third parties to fulfil their objectives. This certainly comes with a cost of fee and more importantly, time. Like this, the OIA's work involves a plethora of transactions, which requires transparency and governance among their stakeholders, and all these are overcome with a general financial software. As of now, in the case of transparency, they operate through various manuals such as business conduct which sets a clear guide for all OIA staff concerning business ethics and behaviors and another whistle blowing system to report any sort of violations to preserve the rights of all parties. As mentioned above, all these are time-consuming, and the records may vary among the parties and may not create any consensus. Therefore, it is observed that it is mandatory to implement a better system or framework in OIA to manage the allocation of funds to public and private markets.

This research proposes a blockchain-based framework for OIA to achieve a better operational and investment performance, to track the funds allocated to different companies, and to ensure an effective system that enforces latest and best international practices. Blockchain technology will drastically improve the management of the distribution process of fund allocation. As per the survey of World Economic forum, September 2015, at least 10% of the global GDP will be stored on blockchain platforms by 2025 [2]. The fund sector involves lot of intermediate parties such as transfer agents, distributors, cash managers, and fund administrators [2]. With the use of blockchain technology, these intermediaries can be reduced, and the transactions can be secured at every stage, while maintaining the transparency. The data is encrypted using hash algorithms which is preserved and verified by every entity involved in the transaction.

Any framework based on blockchain eliminates third parties involved in the operations, reduces the transaction time and costs, and increases the transparency. When we consider a fund allocation body, its operations need to be done in a faster, efficient, and a transparent way. Because many organizations are looking forward to meeting the costs of their projects from OIA, it must be allocated at the earliest and in a fair manner [3]. Therefore, the proposed research is very significant in terms of usefulness. The proposed research uses IR4.0 technology, which is blockchain, a technological revolution of this era.

To achieve the main objective of the research, the following sub-objectives are formulated:

- Analyze similar use cases in blockchain technology.
- Propose an architecture for the prototype.

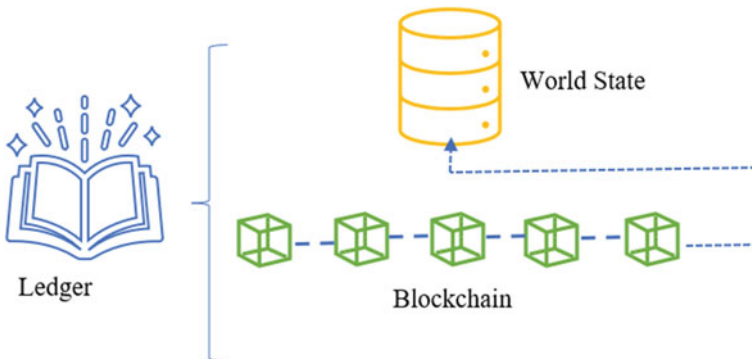
## 2 Background

This section includes information about the details of blockchain ledger, Hyperledger Fabric, the permission based private blockchain framework, and Hyperledger Minifabric, the web based tool used to set up a fabric network easily.

### 2.1 Basic Ledger in a Blockchain

A ledger is a centralized repository that records information of a business object's current state and provides its attribute value and transaction history. These entries could include investments, accounts receivable/payable, and customer deposits.

Figure 1 depicts a blockchain ledger. It is divided into two separate yet related segments: a "blockchain" and "world state (state database)." World state stores the current state of data in the network, whereas the blockchain keeps the entirety of the transaction log (the endorsed transactions) in a blockchain data structure. Unlike other ledgers, these are, however, immutable. This means that in blockchain, once a block is created, i.e., added to the chain, it is almost impossible to modify it. In comparison, the global state might be a database that contains the set of key-value pairs that are added, changed, or removed by the set of blockchain transactions that have been verified and committed [5]. For every channel created within the network, there is a separate ledger belonging to the respective channel.



**Fig. 1** Basic ledger in a blockchain [4]



## 2.2 Hyperledger Fabric

It contains a ledger, implements smart contracts (chaincode), and functions as a system by which members control and store their transactions. However, it differs from other blockchain technologies in that its transactions are private and permissioned. It uses plug-and-play components that are aimed for use within private enterprises.

The most basic Hyperledger Fabric network is shown in Fig. 2 with two organizations (Org1 and Org2) sharing the same channel [6, 7]. A channel could be thought of as a pipeline through which one organization can communicate privately with other involved stakeholders that have joined the same channel [5]. The concept of channels brings a new dimension within the network as it allows the participating organizations to form and join their own network for various reasons and each channel would have its own ledger. Organizations outside the channel will not be involved in any sort of transactions and do not query about any information related with that channel. A single organization can participate in several channels at once.

A peer is a node on the blockchain that records every transaction on a participating channel. Each peer has the option to join one or more channels as needed. The task of organizing transactions, establishing a new block containing ordered transactions, and distributing the newly generated block to all peers on a pertinent channel fall under the purview of the Orderer service. The management of user credentials, including user registration, user enrolment, user revocation, and other actions, is the responsibility of the certificate authority, or CA. To describe rights, responsibilities, and properties to each user, Hyperledger Fabric employs an X.509 standard certificate [5]. An application that communicates with the Fabric

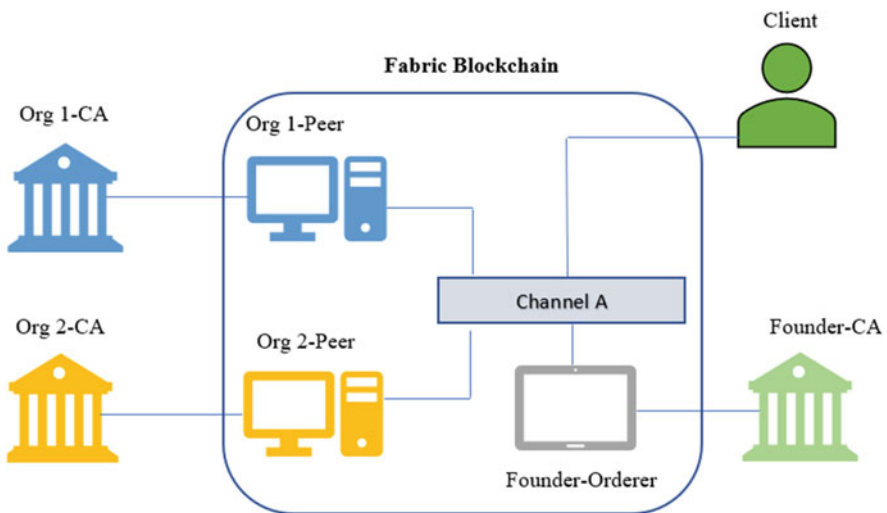


Fig. 2 Simple Fabric Network with two organizations joining same channel A [5]

blockchain network is referred to as a client. That is, the Client can communicate with the Fabric network in accordance with the permissions, roles, and attributes listed in the certificate it obtained from the CA server of the associated company. The entities given inside the blue box are part of the blockchain network, and the outside entities are off chain.

### 2.3 *Hyperledger Minifabric*

With the help of this web-based tool, developers can easily become familiar with Hyperledger Fabric, model their business network, test it, and then deploy it to a live instance of a blockchain network.

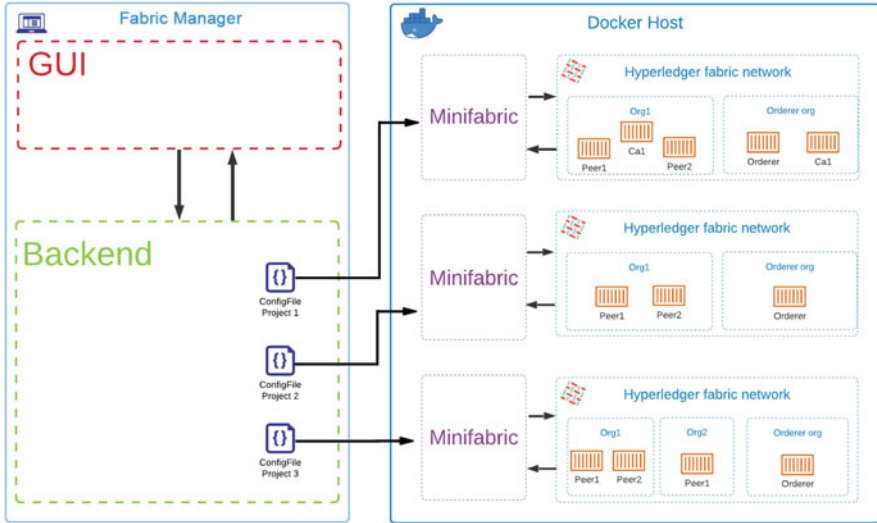
The easiest method to utilize Fabric technology is via Mini Fabric. Unlike Fabric, with the help of just a docker environment, we can smoothly run our first minifab command, regardless of if it is a production server or is a VirtualBox virtual machine. The network can be set up in a few minutes. Minifabric is small but it allows to experience the full skills of Hyperledger Fabric.

The highlighted features are as follows:

- Establishing a fabric network and growing it by incorporating additional organizations.
- Establish, join, query, and update channels.
- Installation, approval, instantiation, invocation, inquiry, and gathering of private data for chaincode.
- Support for block queries, ledger heights, and Hyperledger Explorer.
- Monitoring, evaluation, and identification of nodes.
- Minifabric is only a single little script and a docker container image, thus the ecosystem is not polluted (Fig. 3).

A simple “minifab” command would be enough to start up a fabric network in the provided working directory. Upon its completion, a Fabric network will be running normally using the latest Fabric release on your machine. An application channel named “mychannel” will be created, all peers defined in the network specifications file will be joined to that channel, and a chaincode will be installed and instantiated. The above command will execute most of the Fabric network operations, and the process will take approximately a couple of minutes to complete with respect to the network connection provided to the system in which it is running as the process downloads the Hyperledger Fabric official images from Docker Hub [9].

Business network fundamental data, such as the business model, transaction logic, and access restrictions, are captured by the Business Network Archive (BNA), which then bundles and distributes these parts to a runtime. Fabric also includes a Loopback connector for business networks that provides an active network as a REST API that client apps can use to easily interact with non-blockchain applications. Basic smart contracts are simply stored programs that execute when certain conditions are satisfied. They are frequently used to streamline



**Fig. 3** Basic Mini Fabric architecture [8]

the implementation of an agreement so that each person can instantly be certain of the outcome, without the involvement of a go-between or a waste of time. They can also automate a procedure at work by triggering the subsequent action when certain criteria are fulfilled [10].

### 3 Related Works

In 2018, Indian researchers proposed a blockchain-based architecture for allocating government funds and increasing transparency. The system's prototype was built using Hyperledger composer [11]. Calastone, an investment technology firm, has unveiled a blockchain-based infrastructure for the trading and settlement of investment funds. In this platform, 1800 service providers from 41 countries are connected through a live distributed ledger. These participants have real-time access to trade, service, and monitor their investments [12]. The Internal Revenue Service of Kogi is responsible for collecting and managing domestic revenues that form the state's financial backbone [13]. There are several inherent problems in the present system. The challenge in recognizing tax dodgers, computational errors, a high level of duplication and discrepancies in records, a low level of data security, and the failure to extract and gather pertinent data for timely decision-making rapidly and correctly were just a few of the issues highlighted.

To bring trust to the donors in an NPO in Russia, a system was developed to track the flow of funds and increase transparency. This system was made using Ethereum. The Smart contract was developed using solidity programming language. The server

side is developed using Node.js platform and express framework. MySQL is used for centralized off chain data storage. This project is being carried out because of funding provided by the government of the Russian Federation for applied research [14]. After closely analyzing the above-mentioned research, we were not able to identify many projects that use blockchain technology to achieve our use case.

### 4 Proposed Framework

Figure 4 illustrates the proposed framework.

There are various entities present in this system like the Oman Investment Authority and various other organizations that are receiving investments from OIA.

- *Oman Investment Authority:* With the help of the proposed blockchain network, the OIA can easily allocate and track funds. This system will also increase the transparency and Immutability which thereby will increase the efficiency of the Oman Investment Authority.
- *Organizations:* Every organization falls under the receiver side of the application. They receive investments from the OIA through the application. Since the middlemen are cut off from the system, the system can be trusted. These organizations can also track the transaction made to them.
- *Auditor:* They are responsible for keeping account of the transactions that happened in the network. Each organization in the system has an auditor. An

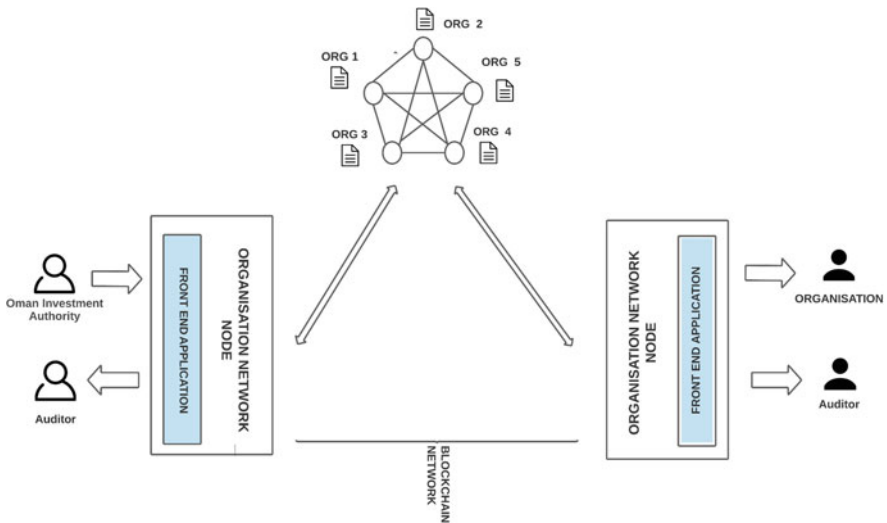


Fig. 4 Proposed framework

auditor cannot look to get more details about the agreement. But he can have access to information like the name of sender and receiver, the amount that is being transferred and the transaction date and time.

The actors are the frontend application, the backend, and the decentralized network. Each user will have two keys, one to login to the application and the other key to submit a transaction. The details regarding the user’s profile are stored in the off-chain database whereas the information regarding the transaction is stored as blocks in the blockchain. Whenever the user wants to view the ledger in which the transactions are stored, a fetch request is given, and the details are retrieved using REST API.

The processes involved in completing a transaction are as follows:

- **Log-in to the Decentralized Application:** In order to conduct a transaction, the user needs to login to the Dapp using his private key. Every user has a different interface depending on the roles given to them by the organization.
- **Create Smart Contract:** The Oman Investment Authority (OIA) needs to create a Smart Contract such that the contract may include the details about the sender and the receiver, fund that is being transferred, and set of conditions that needs to be met. Every network has a maintainer/validator who is responsible for reviewing, verifying, and uploading the smart contract into the network.
- **Notify the Receiver:** The organization that is receiving investments is notified about the Smart Contract and the organization can review the agreement (Fig. 5).
- **Agree to the Contract:** The OIA and the organization must mutually agree on the agreement by signing the smart contract with a digital signature (Fig. 6).
- **Publishing on the blockchain:** When the OIA and the organization sign the Smart Contract, it becomes an official document that is shared on the blockchain.
- **Viewing the smart contract:** Once the transaction is made, anybody with access can view the ledger and evaluate the transaction. Decisions like who can see and what can be seen are made by the organization.

The following explains how smart contracts are used to manage the fund:

A smart contract takes up the role of a lot of intermediaries. Simple “if/else...then...” statements are written on a blockchain to make smart contracts work. If the conditions mentioned by the organization are met, the



**Fig. 5** Creating smart contracts and notifying the organizations



**Fig. 6** Reaching consensus and signing of agreement

smart contract is executed, and the funds are released from the sender's account to the receiver's account. If the receiver does not follow the conditions mentioned in the contract, then the transaction can be reversed or kept in hold. Only parties who have been granted permission can view the results. The entities will be provided a set of rules/guidelines that will ensure the network's integrity. The entities can only change these rules or guidelines through voting, and a vote of 75% of the total network is required to change or add a new rule or guideline.

## 5 Conclusion

Oman Investment Authority fund allocation and tracking currently uses primitive methods to transfer funds. They now use publications like the Authorities and Responsibilities manual, the Investment Manual, the Code of Business Conduct, and others to ensure transaction transparency. However, these methods are not efficient to ensure the transparency of the transaction. This research paper discusses the architecture of a blockchain network to transfer funds based on the Industrial Revolution 4.0 standards. The transaction is transparent, traceable, and immutable due to this technology. The user interface provided by the front-end application will allow users to effortlessly transfer funds and view transactions. The collection of rules supplied will ensure that the network runs smoothly. The development of the network and the user interface, as well as the addition of entities and system testing, will be the emphasis of future work on this project.

**Acknowledgment** The research leading to these results has received funding from the Research Council (TRC) of the Sultanate of Oman under the Block Funding Program BFP/RGP/ICT/20/438.

## References

1. Oman State General Reserve Fund (Oman SGRF) – Sovereign Wealth Fund, Oman – SWFI [Online]. Available: <https://www.swfinstitute.org/profile/598cdaa60124e9fd2d05ba27>. Accessed 19 Jun 2022
2. Deloitte: Impacts of the Blockchain on Fund Distribution, p. 24

3. Oman Investment Authority [Online]. Available: <https://oia.gov.om/Index.php?r=en%2Fsite%2Fpages&slug=governance&csrt=10462007450093286215# GOVERNANCE>. Accessed 19 Jun 2022
4. Hyperledger Fabric: Ledger—Hyperledger-fabricdocs main documentation, <https://hyperledger-fabric.readthedocs.io/en/release-2.2/ledger/ledger.html>. Accessed 20 Jun 2022
5. P. Thummavet, *Demystifying Hyperledger Fabric (1/3): Fabric Architecture* (Coinmonks, 28 Aug 2020), <https://medium.com/coinmonks/demystifying-hyperledger-fabric-1-3-fabric-architecture-a2fdb587f6cb>. Accessed 20 Jun 2022
6. Hyperledger Foundation: Hyperledger fabric, <https://www.hyperledger.org/use/fabric>. Accessed 20 Jun 2022
7. Hyperledger Fabric: Using the Fabric test network—Hyperledger-fabricdocs main documentation, [https://hyperledger-fabric.readthedocs.io/en/latest/test\\_network.html](https://hyperledger-fabric.readthedocs.io/en/latest/test_network.html). Accessed 20 Jun 2022
8. K. Siripatkerdpong, Fabric-Manager (2021) [Online]. Available: <https://github.com/new4761/Fabric-Manager>. Accessed 20 Jun 2022
9. T. Li, IBM Open Technology: *Minifabric: A Hyperledger Fabric Quick Start Tool (with Video Guides)* (Hyperledger Foundation, 29 Apr 2020), <https://www.hyperledger.org/blog/2020/04/29/minifabric-a-hyperledger-fabric-quick-start-tool-with-video-guides>. Accessed 20 Jun 2022
10. IBM: What are smart contracts on blockchain?, <https://www.ibm.com/in-en/topics/smart-contracts>. Accessed 20 Jun 2022
11. A. Mohite, A. Acharya, Blockchain for government fund tracking using Hyperledger, in *2018 International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS)*, (IEEE, 2018), pp. 231–234. <https://doi.org/10.1109/CTEMS.2018.8769200>
12. N. Reeve, *Technology Roundup: Blockchain-Based Fund Trading System Launches* (IPE, 2019), <https://www.ipe.com/technology-roundup-blockchain-based-fund-trading-system-launches/10031307.article>. Accessed 20 Jun 2022
13. S.F. Ayegba, Automated internal revenue processing system: A panacea for financial problems in Kogi state. *West Afr. J. Ind. Acad. Res.* 7(1), 56–69 (2013)
14. H. Saleh, S. Avdoshin, A. Dzhonov, Platform for tracking donations of charitable foundations based on blockchain technology, in *2019 Actual Problems of Systems and Software Engineering (APSSE)*, (IEEE, 2019), pp. 182–187

# Index

## A

Agile methodology, 55, 57  
Anomaly detection, 76–78, 80, 82, 86  
Artificial intelligence (AI), 39, 145, 151, 153, 175  
Assignment problem, 125–140

## B

Balanced assignment problem, 126, 128, 131–134, 136, 140  
Big data, v, 39–52, 89–97  
Big data analytics, 39, 42, 43, 52  
Binary label, 111, 112, 114, 122  
Blockchain networks, 221, 223, 225  
Blood oxygen level, 210, 211, 213  
Boron, 27, 28, 30, 31, 33–36

## C

Chatbot, 146–149, 151–153  
Chest X-ray, 173–179  
Classification algorithm, 28, 39–52, 78, 79, 115, 166  
Convolutional neural networks (CNN), 6, 149, 159, 160, 173–179, 208  
COVID-19, 173–179

## D

Data envelopment analysis (DEA), 89–97  
Data pre-processing, 48, 79, 83–84, 163–164, 170  
DEAP dataset, 160–163, 170

Decision making unit (DMU), 92, 93  
Deep learning, 3, 6, 10, 13, 14, 27, 87, 150, 153, 160, 174, 185  
Degraded document, 102  
Dense layer, 3–14  
Digital banking, 195, 204  
Document image binarization, 33, 101, 102  
Drowsiness, 207–215

## E

Electroencephalography signals, 157–170  
Emotions, 146–149, 157–170, 183–192  
ESP32, 66, 70  
Extreme learning machine (ELM), 17–24

## F

Factor analysis, 196, 204  
Feasible solutions, 140  
Feature extraction, 29–33, 36, 146, 159, 160, 164–166, 170, 174, 176, 185  
FESEM, 27–31, 33, 36  
Financial industry, 195  
FINTECH, 195–204  
Fraud prevention, 75–87  
Fund allocation, 218, 225  
Fund transfer, 225

## H

Human brain, 157  
Hungarian method, 126, 128, 129, 131, 133, 134, 140  
Hyperledger, 219–222



**I**

Image enhancement, 101–108, 175  
 Internet of Things (IoT), 39, 42, 43, 45, 65, 66,  
 70, 146, 149, 150, 153  
 Iron, 28, 30, 31, 33–36  
 Iterative Global Thresholding (IGT), 102–104,  
 106

**K**

K-nearest Neighbors (K-NN), 29, 31–36, 160

**L**

Label tuning (LT), 111–122  
 Lifelogging, 145, 146, 151  
 Linear function, 3, 5, 14  
 Linear programming, 92, 93, 127

**M**

Machine learning (ML), 17, 19, 20, 27–36,  
 39–52, 75–87, 114, 115, 120, 157–170,  
 184, 185, 192  
 Memory bank, 152  
 Metaheuristic algorithms, 17–24  
 Mosquito server, 66, 70  
 MQTT, 65–73  
 MSE, 102–104, 107, 108

**N**

Naive Bayes, 29, 40, 42, 46, 112, 111, 113,  
 115, 116, 149, 185, 192  
 Nanomaterials, 27  
 Nanoparticle images, 27–36  
 Natural language processing (NLP), 147, 153,  
 184  
 Neuron, 3–6, 10–14, 17, 28

**O**

Older adults, 145, 146  
 Oman Investment Authority (OIA), 217, 218,  
 223, 224  
 Optimal solution, 19, 128, 137  
 Optimization, 17–24, 40, 127, 128, 140, 174,  
 190

**P**

Palm leaf manuscripts, 101–108  
 Permalink, 55–60

Pneumonia, 173–179

Prediction, 23, 24, 40, 42, 46, 83, 111, 120,  
 121, 159, 160, 187, 188  
 Probabilistic Neural Network (PNN), 29,  
 31–36  
 PSNR, 102–104, 107, 108

**Q**

Quadratic neuron, 11, 12

**R**

Reduced data storage, 52  
 Reminiscence, 145, 146, 148, 150, 152, 153  
 Retail sector, 89–97

**S**

Scientific workflow, 111–122  
 Security, 51, 52, 56, 60, 65–67, 73, 90, 93–96,  
 146, 149, 201–204  
 Sensors, 39, 41, 43, 45, 65–71, 73, 146, 149,  
 208–211, 213–215  
 Sentiment analysis, 183–192  
 Silver, 28, 30, 31, 33–36  
 Similarity checking, 113, 115–117  
 Smart home management, 65–73  
 Standard deviation, 33, 102, 197  
 Stopwords, 183–192  
 Support vector machine (SVM), 22, 24, 29,  
 40, 42, 46, 78, 159, 160, 166, 170,  
 184–187, 192  
 Synthesis, 27, 30, 33

**T**

TEM, 27, 28, 30, 31, 33, 36  
 Thresholding, 31, 33, 101–104, 108

**U**

Unbalanced assignment problem, 126, 128,  
 133–140

**V**

Vehicle, 126, 207, 208, 213

**W**

Wearable device, 149, 213, 215  
 Wireless mesh network (WMN), 68