IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 14, No. 6, December 2025, pp. 5218~5230
ISSN: 2252-8938, DOI: 10.11591/ijai.v14.i6.pp5218-5230

Graph based semantic email classification: a novel approach for academic institutions

Aruna Kumara B.[1], Madan H. T.[2], Rashmi C.[3], Sarvamangala D. R.[3]
[1]Department of Computer Science and Engineering, Proudhadevaraya Institute of Technology, Hosapete, India
[2]Department of Electronics and Communication (Advanced Communication Technology), NMAM Institute of Technology, Nitte (Deemed to be University), Karkala, India
[3]School of Computing and Information Technology, REVA University, Bangalore, India

Article Info

Article history:

Received Nov 11, 2024
Revised Oct 3, 2025
Accepted Oct 18, 2025

Keywords:

Email classification
Graph based classification
Multiclass classification
Natural language processing
Semantic similarity

ABSTRACT

Electronic mail classification in educational institutes becomes the fundamental task to manage information efficiently. Due to the globalization and the technological advancement, volume of email users increasing consistently, which in turn increases the volume of digital data exponentially. This necessitates the developing automated email classification systems for the better and organized work. This paper develops a novel graph-based similarity (GBS) approach based on semantic similarity to address these challenges. The method initially selects the most relevant features based on feature weights, later it builds a graph by using Jaccard co-efficient method for each category with features as nodes and correlation between the nodes as edges. Later, these graphs are used as templates for each category and classifies each new incoming email into the specific class based on the similarity among the graph templates and a new email. The GBS method was compared with the well-known benchmarked email classifiers and the findings demonstrated that the GBS method outperformed with 98.91% accuracy after fine-tuning of graph parameters and the classifier hyper parameters. Additionally, receiver operating characteristic (ROC) curve analysis was conducted, achieving a highest area under curve (AUC) score 0.989, demonstrating robust classification proficiency across all categories.

Corresponding Author:

Madan H. T.
Department of Electronics and Communication (Advanced Communication Technology)
NMAM Institute of Technology (NMAMIT), Nitte (Deemed to be University)
Nitte, Karkala Taluk, Udupi-574110, Karnataka, India
Email: madan.ht@nitte.edu.in

1. INTRODUCTION

In the current digital age, email remains a basic communication medium at both personal and professional levels. The usage of emails for communication purpose is still growing exponentially in spite of the swift technological development because it offers exclusive composite of proficiency, protected, and trustworthiness. After the pandemic, email communication increased exponentially in the academic field, because the learning was shifted to online mode. According to the Radicati groups of email statistics report, 4.549 billion users using email globally as of 2025; by the end of 2027, that number is predicted to reach 4.849 billion [1].

A classic academician gets almost 60-70 emails per day, which leads to a flood of emails if he goes on vacation of 10-15 days. In addition, he needs to devote a substantial part of his working hours to process emails, which reduces the quality and productivity of the employee as well the organization. Accordingly, email management is an essential task that both individuals and organizations must deal with. In general, the primary tool for email management is categorizing emails into specified categories based on user requirements. For instance, in an academic institution/university, one can classify an incoming email into academics, research, placements, examinations, and others for easy access.

To categorize emails into predetermined groups, different categories of learning methods are available, including supervised learning, content-based learning, unsupervised learning, and semi-supervised learning [2], [3]. This work used the concept of supervised learning for email classification, because it is a robust choice for academic email classification as it offers vibrant performance metrics and interpretability. Support vector machines (SVM) [4], genetic algorithms (GA) [5], artificial neural networks (ANN) [6], [7], decision trees (DT) [8], naïve Bayes (NB) [9], random forest (RF) [10], convolution neural networks (CNN) [11]–[13], and k-nearest neighbor (KNN) [14], are some of the methods that apply the supervised learning principle. Some of these classifiers showed prominent performance on email, but still faced challenges due to the nature of email data. We know that, the data in real- time email is unstructured, noisy, and high dimensional, and it makes it difficult to understand the structure of the data by a classifier. Then, researchers developed different classifiers based on semantic nature [15], [16] and tree/graph-based nature [7], [17]–[19] to solve these problems and also performed well on the public datasets. However, these classifiers failed to focus on the structure and relationship between the emails when using real-time datasets. Hence, this work focused on developing a graph-based email classifier since tree-based and graph-based classifiers have drawn a lot of attention recently because of their non-linear nature, which allows them to adapt to virtually any classification task.

This research proposed a unique graph-based email classifier for effective email classification. Following are the key contributions of this work: i) a novel graph-based similarity (GBS) approach is proposed for a multi-class email classification system, where each email is represented as a graph built from top-k term frequency-inverse document frequency (TF-IDF) features as nodes. Edges between the features are computed using the Jaccard coefficient which allows the model to capture the semantic relationship between the nodes unlike the conventional methods; ii) a template-based classification approach is presented, where each category is associated with its class representative graph. Categorization of new unseen incoming email is performed by comparing its graph with the class specific template graphs; and iii) the proposed GBS approach is extensively evaluated on a real-time academic email dataset and it was compared with conventional classifiers such as multinomial naïve Bayes (MNB), linear support vector classifier (LSVC), long short-term memory (LSTM), and semantic based forensic analysis and classification of email data (seFACED). The model undergoes fine-tuning across different TF-IDF threshold values. The receiver operating characteristic (ROC) curve analysis also conducted to validate the discriminative capability of the method, and the findings demonstrates superior performance predominantly with imbalanced classes.

The remainder of this paper is structured as follows: section 2 gives the various works carried out on different email classification techniques used for email classification. Section 3 discusses the proposed methodology, which describes building graphs, finding the node similarity between graphs, and grouping emails into predefined categories. Section 4 discusses the results obtained after rigorous experiments on real-time and benchmark datasets. Finally, section 5 concluded the work.

2. RELATED WORK

The relevant work was summarized into three categories: supervised methods, tree-based methods, and sematic-based methods. These methods classify emails into predefined categories. Some of them are detailed as follows.

Angelova and Weikum [20] developed a novel method for text classification that utilized a graph in which text is hyperlinked. This method used neighbor-learning to ascertain the link between data items. The accuracy and robustness of the classifiers were enhanced by this technique. Furthermore, neighbor trimming and edge weighting based on similarity were employed to lessen the noise in the neighborhood; class label-based similarity was utilized to further alter the weight. The challenging task of multi-class automated hate speech categorization for text [21] was solved in this study with significantly improved outcomes when significant challenges were first identified. Ten distinct binary-categorized datasets with various kinds of hate speech were created.

Adnan *et al.* [22] addressed various challenges in spam email classification using ensemble learning techniques. Their method employed different classifiers including DT, LR, KNN, AdaBoost, and Gaussian NB as base classifiers. Their evaluation demonstrated that AdaBoost was the strongest individual classifier and also it showed that the proposed stacking method accomplished higher performance.

Hassanat [23] proposed a unique way for accelerating big data categorization. Two approaches were used, and the instances were sorted according to how closely they resembled two local locations.

The first strategy chooses local points based on how much they resemble the extreme global points, whereas the second technique selects local sites at random. The outcomes of numerous trials performed on multiple large datasets reveal respectable accuracy rates compared to cutting-edge techniques and the KNN classifier.

A DT method [24] based on the Rao-Stirling index was proposed. When quantifying data impurity, the Rao-Stirling index takes class distances into account and gives higher weight to reference pairs in classes that are farther apart. The outcomes demonstrated that the recommended method is more accurate, suggesting that accounting for class distances could improve DT accuracy.

Sonowal [25] used binary search feature selection (BSFS) with a rating system based on the Pearson correlation coefficient to categorize phishing emails. The proposed strategy makes advantage of four categories of features from the hyperlinks, email body, and content. Generally, the four dimensions mentioned earlier were used to choose 41 attributes. Thus, the BSFS method outperformed the sequential forward feature selection (SFFS) method (95.63%) and the without feature selection (WFS) method (95.56%) with a score of 97.41%. This study showed that while the SFFS takes the most time to find the optimal feature set, the WFS does so the quickest. But compared to other approaches, the WFS's accuracy is rather low. The main finding of the experiment was that the BFSF took the smallest amount of time to evaluate the best feature set more accurately, even after eliminating a few features from the feature corpus.

A novel method for resolving the proximity searching issue was presented [26]. The methodology relied on employing a directed graph known as the k-nearest neighbor graph (k-NNG) to index the database. This graph establishes a connection between every entry and its nearest neighbors. For range and nearest neighbor queries, the method offered two finding algorithms that used the metric and navigational properties of the k-NNG network. In addition, the authors demonstrated the competitiveness of the approach strategy versus existing ones. For instance, utilizing only 0.25% of active electronically scanned array (AESAs) required space, the nearest neighbor search methods used in this paper completed 30% more distance evaluations in the metric document space. Conversely, the pivot-based method was ineffective in the same field.

Shimomura and Kaster [27] presented two significant improvements to graph related algorithms for similarity findings. In the first, the primary graph categories that are currently used for similarity checks were reviewed, and the best illustrative graphs in an environment that is shared by precise and closest search strategies were experimentally evaluated. The later one was a novel connected-partition technique to proximity graph construction and similarity search responses called HGraph.

Malkov *et al.* [28] proposed a new method for resolving the metric space KNN search problem. An accessible world network with vertices for the components to be stored, edges for the connections between those elements, and a form of the greedy method for finding served as the foundation for the search structure. The initial Delaunay graph approximation linkages were simply maintained to construct the navigable tiny environment. The approach, which was presented in terms of arbitrary metric spaces, was both general and simple. The probabilistic KNN queries accuracy can be modified without rewriting the structure.

Numerous classifiers for machine learning have been developed, and some of them have demonstrated notable performance on a variety of datasets, according to a thorough review of the literature. Additionally, classifiers based on trees and graphs have gained popularity in recent years, and only a few tree-based and graph-based classifiers showed good performance on high-dimensionality email data. However, a large number of graph-based and tree-based classifiers have been developed but failed to understand the lexical and contextual relationship between features. Hence, this paper suggests a graph-based classifier based on semantic similarity for an effective email classification system using TF-IDF and the Jaccard coefficient.

3. METHODOLOGY

The proposed methodology employs a unique GBS method for email classification system. The method comprises five main phases: data acquisition, pre-processing the raw email data, finding and selecting relevant features, building a graph, and grouping email data according to the similarity between two graphs. The framework of the proposed email classification method based on GBS is illustrated in Figure 1.

3.1. Data acquisition

The REVA email dataset contains 3,400 email samples with 4 different categories such as: examination, research, academics, and placements. These categories are considered in this study [29]. The email samples in the dataset have been distributed unevenly and it is illustrated in the Figure 2.

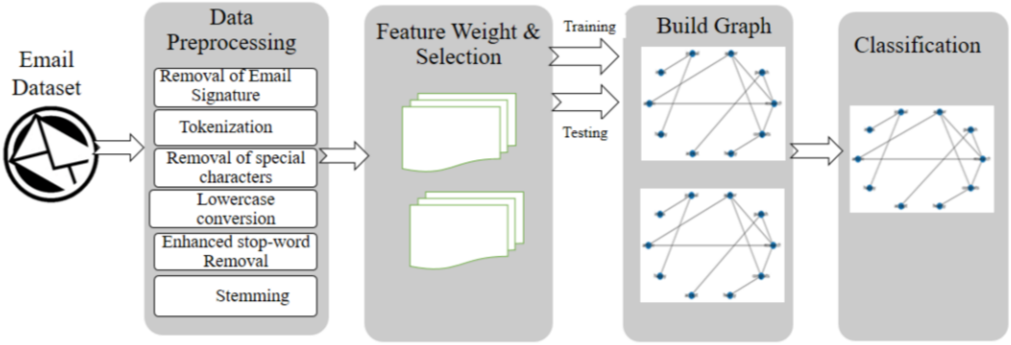


Figure 1. The proposed architecture for email classification using GBS

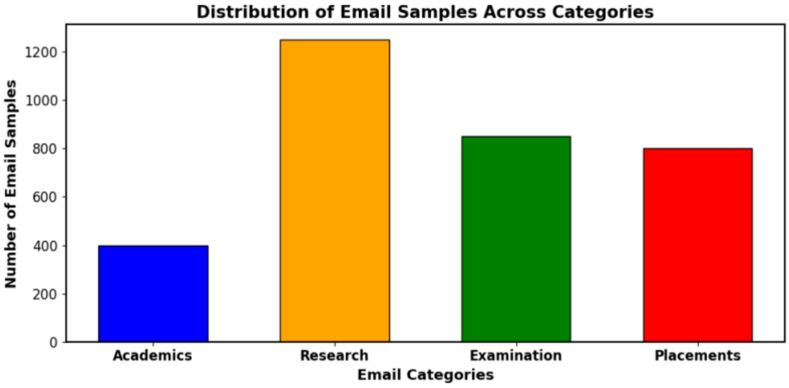


Figure 2. Distribution of email samples

3.2. Data preparation

To transform the raw email data into the format needed for additional processes like feature selection and classification, data pre-processing is necessary. Several data pre-processing methods are used on unprocessed email data, such as tokenization, lowercase conversion, email signature removal, stop-word removal, and punctuation removal [30], [31]. After pre-processing, there has been a clear decline in the volume of email data. This will improve the processing efficiency of the suggested email classification system. The detailed explanation of data preparation is provided as follows, accompanied by an example. Four email samples (D_1, D_2, D_3, and D_4) that fall under the dataset's research category were taken into consideration to demonstrate the suggested method's working technique.

- D_1: hearty congratulations to authors, published a research article in Scopus indexed journal.
- D_2: dear all, I'm pleased to announce that a research article I wrote has been published in a journal with an impact factor of 1.8 that is indexed by SCI.
- D_3: dear all, I am pleased to inform that the research paper co-authored by me and my RU Research Scholar Mr. Imran Khan is accepted in the International Journal of Electronics and Telecommunications, a Web of Science indexed journal.
- D_4: dear all, it is our great pleasure to share that one of our articles titled "A comprehensive research on its topologies, modelling, control and its applications" is accepted in one of the top-rated international peer reviewed Journal of "IET Power Electronics".

The reduced dataset after pre-processing techniques applied on email documents D1, D2, D3 and D4 are as follows:

- D_1’: {congrats, author, publish, research. article, Scopus, index, journal}.
- D_2’: {happy, announce, research, article, write, publish, journal, impact, factor, index, sci}.
- D_3’: {research, happy, article, scholar, author, publish, web, accept, science, journal, index,}.
- D_4’: {great, happy, article, research, accept, international, review, journal. IET, Power, Electronics, paper, write, scholar, author}.

Four email documents had a combined length of 145 words prior to the use of pre-processing techniques; then length of these four documents becomes 47 words after preprocessing phase, which is almost 69% reduction in the size of the original data. This helps the learning techniques work more efficiently. Further, from the reduced data, n unique words are considered occurs oftenly among the different email documents for the feature selection phase.

3.3. Feature engineering

This section describes the identification of weights for each feature/term and selecting them for further process. Initially, a binary model is created for n oftenly occurred words in the reduced dataset using bag of words (BoW) method. After that, a weight representing each word's relevance was determined using TF-IDF [32]. The BoW representation for the n unique words (i.e., n =10 for this study) of the reduced dataset is shown in Table 1.

Table 1. Binary BoW representation for the selected terms

Doc/Word	Research	Article	Journal	Publish	Happy	Author	Index	Scholar	Write	Accept
D_1	1	1	1	1	0	1	1	0	0	0
D_2	1	1	1	1	1	0	1	0	1	0
D_3	1	1	1	1	1	1	1	1	0	1
D_4	1	1	1	0	1	1	0	1	1	1

The BoW model, on the other hand, just indicates the presence of words; it makes no allowances for the implication of individual terms inside an email content. For instance, in the fourth document, the word "scholar" has greater significance than other nouns. However, when words exist in the email, they receive the value '1' in this model; otherwise, they receive the value '0'. This means that a bag of words model with TF-IDF score was used in place of "0s" and "1s" used in the original model. For a given word, TF-IDF multiplies TF and IDF to find the score for that word in the document. Consequently, the score for every word in the document computes as shown in (1) to (3).

$$TFIDF\ (word, doc) = TF\ (word, doc) \times IDF\ (word) \tag{1}$$

Thus, two matrices must be calculated for this method: first one (TF) counts the terms or words that occur in each document, and the other (IDF) determines the relevance of each word that occurs in each document. Both of them are calculated using (2) and (3).

$$TF(word, doc) = \frac{number\ of\ times\ word\ occur\ in\ document}{number\ of\ words\ in\ document} \tag{2}$$

$$IDF(word) = log\left(\frac{number\ of\ documents}{1{+}number\ of\ documents\ with\ word}\right) \tag{3}$$

To determine the meaning of each term, the TF dictionary for the "n" most often recurring words with their TF values was first produced. Afterwards, the same set of terms with their corresponding IDF values were included in the IDF lexicon. Ultimately, as Table 2 illustrates, the TF-IDF score is produced. Terms that have scores close to '0' imply less relevance, whereas those with high TF-IDF values indicate greater relevance. A feature set was generated using the top 10 words having the highest TF-IDF scores, and a graph was then created using this feature set. In the following step, a graph was created with these properties, and it was then utilized to categorize incoming emails in the future into the appropriate groups.

Table 2. Term frequency-inverse document frequency model

Doc/Word	Research	Article	Journal	Publish	Happy	Author	Index	Scholar	Write	Accept
$D_1^{'}$	0.989	0.042	0.989	0	0.987	0.034	0.581	0	0.081	0.085
$D_2^{'}$	0.989	0	0.989	0	0.0895	0.027	0.624	0	0.852	0.847
$D_3^{'}$	0.989	0	0.989	0.048	0.981	0.033	0.158	0.542	0.075	0.079
$D_4^{'}$	0.989	0	0.989	0.059	0.879	0	0.254	0.124	0.759	0.568

3.4. Graph building using Jaccard similarity

During this step, an email-document graph was constructed using the features with high TF-IDF scores for email classification. Here is how the graph is defined (4) to (6):

$$G = (N, E) \tag{4}$$

$$N = \{n_1, n_2, n_3, \dots, n_n\} \tag{5}$$

$$E = \{e_1, e_2, e_3, \dots, e_n\} \tag{6}$$

Where, G is a graph with set of nodes (N) and edges (E). Between TF-IDF scores of two emails M and N, the Jaccard similarity [33] is calculated as (7).

$$\mathrm{Sj}(M, N) = \frac{|M \cap N|}{|M \cup N|} = \frac{|M \cap N|}{|M|+|N|-|M \cap N|} \tag{7}$$

We define an edge between the vertices n_i and n_j if the Jaccard similarity exceeds τ as in (8).

$$d(i,j) = \begin{Bmatrix} S_j\left(D_i, D_j\right), & if\ S_j\left(D_i, D_j\right) > \tau \\ 0, & Otherwise \end{Bmatrix} \tag{8}$$

Thus, the edge $d_{i,\,j}$ represents the Jaccard similarity between two email documents d_i, and d_j.

After the research category graph was constructed as shown in Figure 3, the remaining categories graphs (academics, exams, and placements) were also constructed using the same procedure. Later, these graphs were preserved as templates. They are used during the categorization of incoming mail that has not read yet.

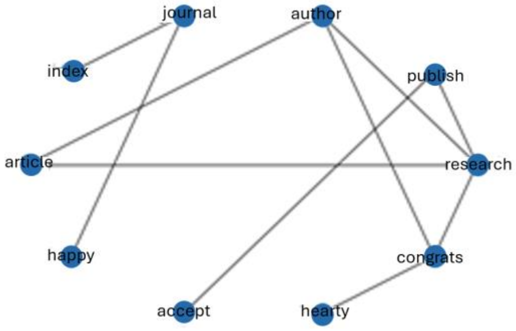


Figure 3. Graph with weighted edges

3.5. Graph based classification

In this phase, emails were classified into one of the predefined categories based on the similarity between graphs. The dataset has four categories: academics, examination, research, and placements. First, the input graph for the new unseen email was constructed using the methods described in sub-sections B, C, and D of section 3. Later, this graph is fed into the trained model to predict the category. Then, to predict the respective category, the method computes the graph similarity score between the input graph and the graph templates of various classes as (9).

$$S = \{C_1, C_2, C_3, \dots, C_{n1}\} \tag{9}$$

Where C_n represents the number of classes (In this study academics, research, examination, and placements).

For a new email document D_{new}, represent it as TF-IDF vector X_{new} and connect it to the graph G based on its Jaccard similarity to the existing nodes. A new email will be classified into one of the predefined categories using the existing labeled nodes in the graph. A simple KNN is used for classification. For the new email document, the classification task works as follows:

i) Compute the Jaccard similarity between X_{new} and the TF-IDF vectors of the existing emails.
ii) Find the k-closest neighbors based on Jaccard similarity.
iii) Assign the class Y_{new} to the fresh email document based on a majority vote from the classes of the nearest neighbors as (10).

$$Y_{new} = \arg\max C_k \sum_{i=1}^{k} 1(Y_i - C_k) \tag{10}$$

Where Y_i denotes the class label of the ith nearest neighbor, C_k is kth class and $\mathbb{I}$ $(Y_i = C_k)$ represents the indicator function that returns 1 when $Y_i = C_k$ and 0 otherwise.

Subsequently, the complete algorithm for classification of emails based on graph similarity approach is discussed in Algorithm 1. The Algorithm 1 explains the proposed graph-based email classification using TF-IDF and Jaccard co-efficient. In step 1, various algorithms are applied to clean the raw data and makes data to suitable for further machine learning operations, later relevant top k features are selected using TF-IDF score. Then, a graph was built using these features as vertices and the relationship between these features as edges, Jaccard co-efficient is used to find the similarity score between the vertices (threshold value considered in this study is 0.75). In step 2, a graph will be constructed for unseen email. Later, step 3 computes the similarity score between and unseen email graph and a graph of existing emails. Finally, in the step 5 the unseen email assigned with the class label with highest score obtained in the step 4.

Algorithm 1. Graph-based email classification using TF-IDF and Jaccard co-efficient

```
Input: email dataset $D = \{d_1, d_2, \dots, d_n\}$, unseen email $d_u$, feature number k, and threshold of
Jaccard
co-efficient ϴ
Output: predicted class label for unseen email
Steps:
  1.  for each email document di ∊ D:
      -   Preprocess $d_i$ → get terms $T = \{f_1, f_2, \dots, f_n\}$
      -   Compute TF-IDF(t, $d_i$) À t ∊ T
      -   Select the top features → $V_i$
      -   Initialize the set of edges $E_i$ ← Ф
      -   for every pair of edges (u, v) ∊ Vi * Vi,  u ≠ v:
          -   compute the Jaccard similarity:
              -   J(u, v) ← C (u) ∩ C(v) / C(u) Ù C(v)
          -   If J (u, v) >= ϴ:
              -   $E_i$ ← $E_i$ Ù {(u, v), weight = J (u, v)}
      -   Build graph $G_i = (V_i, E_i)$
  2.  Repeat the step1 to construct the graph for unseen email $d_u$
      -   $G_u$ ← ($V_u$, $E_u$)
  3.  Compute graph similarity score between $G_u$ and every $G_i$ using modified Jaccard co-
      efficient
      -   $Sim\,(Gu, Gi) = (Eu \cap Ei)/(Eu \cup Ei)$
  4.  Select the graph with highest similarity score
  5.  Assign the class label of $d_j$ to $d_u$
      return the class label for unseen email $d_u$
```

3.6. Model fine tuning and hyper parameter optimization

To enhance the performance of the GBS model, structured fine-tuning was performed on various parameters involved in the processing pipeline. The parameters considered for optimization include the volume of topmost features per email, ranging from 5 to 20. Additionally, the threshold value of Jaccard similarity for computing edge values was tested at 0.2, 0.3, 0.4, and 0.5.

4. RESULTS AND DISCUSSION

The proposed GBS method's performance was assessed using the academic email dataset. Additionally, the GBS method's performance was contrasted with that of other traditional email classification techniques. Including seFACED [34], LSTM [35], LSVC, RF, and MNB.

4.1. Performance analysis

The performance of various evaluation measures like accuracy [36], precision [37], recall [38], and F1-score [39] measures were tested on the different email classifiers and the proposed classifier as (11) to (14).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{11}$$

$$Precision = \frac{TP}{TP+FP} \tag{12}$$

$$Recall = \frac{TP}{TP+FN} \tag{13}$$

$$F1 - score = 2(\frac{precision*recall}{precision+recall}) \tag{14}$$

Where TP, FP, TN and FN specify true-positives, false-positives, true-negatives, and false-negatives correspondingly. The performance of different evaluation measures on different types of email classification is illustrated in Table 3. The average recall, F1-score, precision, and accuracy of the proposed GBS approach are 0.96, 0.97, 0.97, and 0.98 respectively.

Table 3. Performance of proposed model

Class	Accuracy	Precision	Recall	F1-score
Academic	0.989	0.978	0.958	0.973
Examination	0.989	0.985	0.964	0.976
Placements	0.974	0.967	0.955	0.978
Research	0.992	0.969	0.962	0.982

To evaluate the performance of GBS email classification method, a confusion matrix was constructed. Figure 4 gives the detailed insight of the proposed method's ability to correctly identify and differentiate between various classes. The matrix shows that the proposed method achieves better classification accuracy for each category, particularly for placement and research categories with minimal misclassifications. From the matrix it was observed that, the total correct predictions were 3,363 out of 3,400 email samples. The proposed GBS approach achieved with highest accuracy of 98.91% with low misclassification rate of 1.09%. Maximum misclassification occurred between the academics and examination category due to the mutual terms between these two classes.

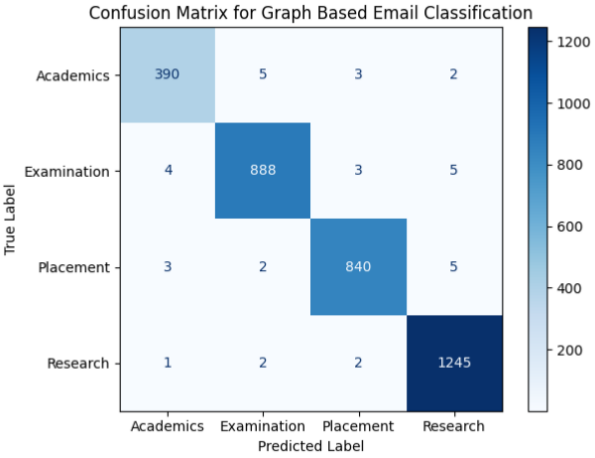
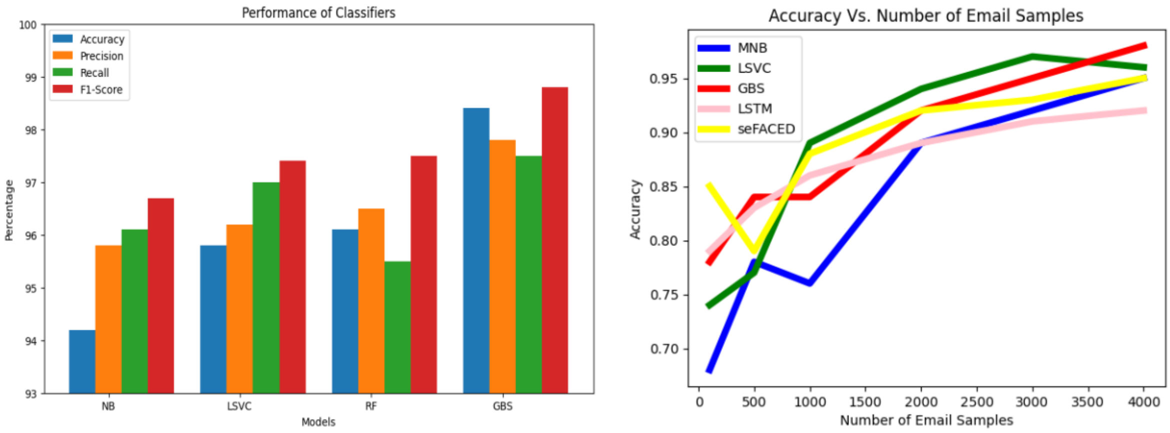


Figure 4. Confusion matrix of the proposed method

4.2. Comparative analysis

Figure 5 shows the performance analysis of proposed method. The performance of the GBS and conventional classifiers across a range of performance metrics is displayed in the Figure 5(a). NB obtained 94.1% accuracy, 95.8% precision, 96.1% recall, and 96.9% F1-score, as per the experimental findings. Consequently, with 97.8% accuracy, 96.2% precision, 97.0% recall, and 96.4% F1-score, LSVC outperformed the others. Furthermore, the RF technique produced good results, with a 96.3% F1-score, 97.1% recall, 97.0% precision, and 97.7% accuracy. With 98.2% accuracy, 97.8% precision, 98.8% recall, and 98.7% F1-score, the proposed GBS outperforms other classifiers.

Figure 5(b) recorded the assessment of GBS and other various email classifiers (NB, LSVC, RF, LSTM, and seFACED) on the academic email dataset. From the experimentation, it was perceived that, the proposed GBS scored 78.2% accuracy, whereas LSTM and seFACED scored 79.3% and 85.1% respectively when the method was tested on 500 email samples. Also, the results illustrate that, the GBS produced 92.1% accuracy, compared to LSTM and seFACED produced 89.02% and 91.6%. Furthermore, the results demonstrated that the recommended GBS performed more effectively than other classifiers with 95.2% and 98.91% accuracy when 3,000 and 4,000 email samples were considered. These results designate that the proposed approach's performance improves as the dataset size increase.



(a) (b)

Figure 5. Performance analysis of proposed method of (a) comparative analysis of GBS with traditional classifiers (b) accuracy comparison of GBS with conventional methods

The ROC curve was generated for all the four categories. With the area under curve (AUC) used to quantify the classification performance. The true positive rate (TPR) and false positive rate (FPR) were computed as (15) and (16).

$$TPR = \frac{TP}{TP+FN} \tag{15}$$

$$FPR = \frac{FP}{FP+TN} \tag{16}$$

Figure 6 illustrates the ROC curve of the proposed GBS classifier for all four categories, highlighting the classifier's ability to separate emails into their correct categories. The AUC values were 0.978 for academics, 0.981 for research, 0.989 for examination, and 0.982 for placements. These results were demonstrated that the classifier performs exceptionally well across all categories with the examination category exhibiting the highest classification accuracy. This validates the effectiveness of the graph-based classifier in handling multi-class email classification using the Jaccard similarity. Table 4 gives the comparison of proposed GBS method with the existing methods (i.e., semantic graph neural network (SGNN), stacking ensemble, seFACED, and MalFSCIL). The overall performance of the GBS method is 98.91% accuracy, which is 1.04, 0.09, 3.91, and 2.92% better than the existing methods. From this comparative analysis, we can conclude that the proposed GBS method outperformed other methods by achieving highest accuracy.

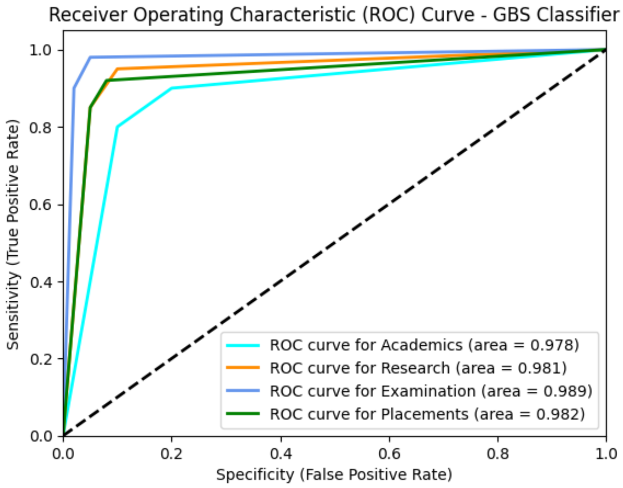


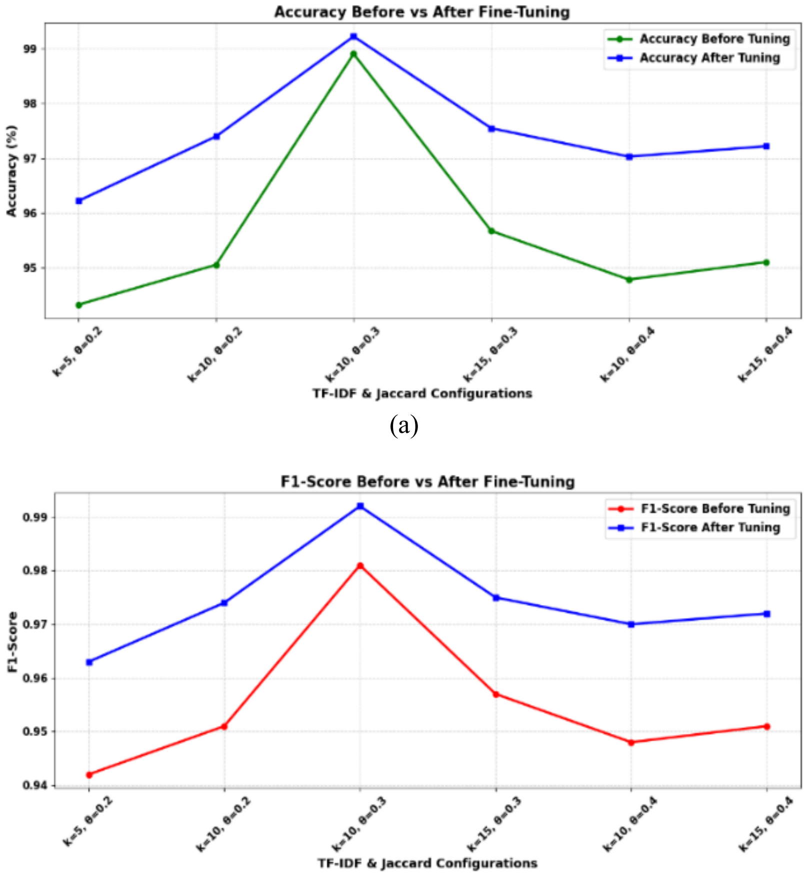
Figure 6. ROC curve for the proposed GBS email classifier

Table 4. A comparison of the proposed method with current methods

Author	Year	Methods	Accuracy (%)
Pan *et al.* [18]	2022	SGNN	97.872
Adnan *et al.* [22]	2024	Stacking Ensemble	98.80
Hina *et al.* [34]	2021	seFACED	95
Chai *et al.* [40]	2025	MalFSCIL	90.58
Proposed		GBS	98.91

4.3. Impact of fine tuning

To evaluate the impact of fine-tuning on the proposed model (Figure 7), different sets of experiments were conducted over TF-IDF feature sizes and threshold limits of Jaccard co-efficient. Figures 7(a) and 7(b) demonstrates the impact of fine-tuning approaches among accuracy and F1-score respectively. According to the findings, the proposed model outperformed consistently across all combinations after fine-tune the model. The proposed model enhanced its accuracy from 94.41% to 96.30% for the combination (k =5, theta =0.2) and obtained its highest 99.23% for the configuration (k =10, theta =0.3). Correspondingly, the F1-score enhanced among all combinations and achieved highest 0.992 for the similar combination. These findings validate that the proposed classifier boosts its performance after fine-tuning the top ranked features and the Jaccard's threshold value in graph construction.



(a)

(b)

Figure 7. Impact of fine tuning on GBS method of (a) accuracy of the GBS before and after fine tuning and (b) F1-value of the GBS before and after fine tuning

5. CONCLUSION

In this study, an innovative graph-based semantic email classification method was developed using TF-IDF and the Jaccard coefficient. Initially, the method used different data preprocessing methods to clean the raw data; later, TF-IDF was used to extract the most important and relevant features from the updated dataset. Next, a graph was constructed for a class using the Jaccard coefficient, where the features were used as nodes and the correlation between the nodes as edges. Then, graphs were built for other categories as well, and these graphs were used as templates during classification. Later, the approach finds the degree of similarity between both graphs by analyzing their similarities. Finally, the method classifies unseen email into the prescribed category based on the similarity between those graphs. The experimental demonstration showed that the GBS classifier performs better than the traditional classifiers with 98.91% accuracy before fine tuning the model, and achieved highest with 99.23% accuracy after fine tuning. These findings confirmed that fine-tuning of TF-IDF feature counts and edge thresholds significantly enhanced classification performance. While the findings are promising, future researchers can explore the incorporation of graph neural networks to help the model to learn richer feature interactions while constructing graphs. Consequently, it is planned to extend the work to support multi-modal data, and dynamic graph construction.

FUNDING INFORMATION

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Aruna Kumara B.	✓	✓	✓	✓	✓			✓	✓	✓			✓	
Madan H. T.	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	
Rashmi C.		✓	✓	✓	✓	✓	✓	✓		✓	✓		✓	
Sarvamangala D. R.		✓	✓	✓	✓	✓	✓		✓	✓	✓	✓		

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis

I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting

Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

Data availability is not applicable to this paper as no new data were created or analyzed in this study.

REFERENCES

[1] Radicati Group Inc., "Email statistics report, 2023-2027," *Radicati,* Palo Alto, CA, USA, 2023. [Online]. Available: https://www.radicati.com/wp/wp-content/uploads/2023/04/Email-Statistics-Report-2023-2027-Executive-Summary.pdf

[2] A. M. Idrees, N. S. Elhusseny, and S. Ouf, "Credit card fraud detection model-based machine learning algorithms," *IAENG International Journal of Computer Science*, vol. 51, no. 10, pp. 1649–1662, 2024.

[3] S. Wasi, S. I. Jami, and Z. A. Shaikh, "Context-based email classification model," *Expert Systems*, vol. 33, no. 2, pp. 129–144, Apr. 2016, doi: 10.1111/exsy.12136.

[4] Q. Zheng, X. Tian, M. Yang, and H. Su, "The email author identification system based on support vector machine (SVM) and analytic hierarchy process (AHP)," *IAENG International Journal of Computer Science*, vol. 46, no. 2, pp. 178–191, 2019.

[5] D. Liu and S. Fan, "A modified decision tree algorithm based on genetic algorithm for mobile user classification problem," *The Scientific World Journal*, vol. 2014, pp. 1–11, 2014, doi: 10.1155/2014/468324.

[6] K. Iqbal and M. S. Khan, "Email classification analysis using machine learning techniques," *Applied Computing and Informatics*, vol. 21, no. 3-4, pp. 390–402, May 2022, doi: 10.1108/ACI-01-2022-0012.

[7] W. Liu and Q. Guo, "Graph classification via hierarchical structure learning with multi-scale convolution and pooling," *Engineering Letters*, vol. 33, no. 5, pp. 1232–1242, 2025.

[8] H. Chen, Y. Zhan, and Y. Li, "The application of decision tree in Chinese email classification," in *2010 International Conference on Machine Learning and Cybernetics*, IEEE, Jul. 2010, pp. 305–308, doi: 10.1109/ICMLC.2010.5581046.

[9] C.-C. Le, P. W. C. Prasad, A. Alsadoon, L. Pham, and A. Elchouemi, "Text Classification: naïve Bayes classifier with sentiment Lexicon," *IAENG International Journal of Computer Science*, vol. 46, no. 2, pp. 141–148, 2019.

[10] A. A. Akinyelu and A. O. Adewumi, "Classification of phishing email using random forest machine learning technique," *Journal of Applied Mathematics*, vol. 2014, pp. 1–6, 2014, doi: 10.1155/2014/425731.

[11] A. Li and L. Zhang, "Multi-label text classification based on label-sentence bi-attention fusion network with multi-level feature extraction," *Electronics*, vol. 14, no. 1, Jan. 2025, doi: 10.3390/electronics14010185.

[12] M. Y. Arafat, S. Khatun, M. Halder, M. N. A. Sheikh, M. N. Adnan, and S. S. Kabir, "An efficient convolution neural network-based novel framework for potato leaf diseases classification and identification," *IAENG International Journal of Computer Science*, vol. 52, no. 7, pp. 2411–2428, 2025.

[13] X. Liao, Y. Wu, Z. Wang, D. Wang, and H. Zhang, "A convolutional spiking neural network with adaptive coding for motor imagery classification," *Neurocomputing*, vol. 549, Sep. 2023, doi: 10.1016/j.neucom.2023.126470.

[14] A. Harisinghaney, A. Dixit, S. Gupta, and A. Arora, "Text and image based spam email classification using KNN, naïve Bayes and reverse DBSCAN algorithm," in *2014 International Conference on Reliability Optimization and Information Technology*, IEEE, Feb. 2014, pp. 153–155, doi: 10.1109/ICROIT.2014.6798302.

[15] E. M. Bahgat, S. Rady, W. Gad, and I. F. Moawad, "Efficient email classification approach based on semantic methods," *Ain Shams Engineering Journal*, vol. 9, no. 4, pp. 3259–3269, Dec. 2018, doi: 10.1016/j.asej.2018.06.001.

[16] K. Wang, X. Fu, Y. Liu, W. Chen, and J. Chen, "Att-FMI: a fusing multi-information model with self-attentive strategy for relation extraction," *IAENG International Journal of Applied Mathematics*, vol. 53, no. 3, pp. 145–152, 2023.

[17] A. Sharaff and H. Gupta, "Extra-tree classifier with metaheuristics approach for email classification," in *Advances in Computer Communication and Computational Sciences*, Springer, Singapore, 2019, pp. 189–197, doi: 10.1007/978-981-13-6861-5_17.

[18] W. Pan *et al.*, "Semantic graph neural network: a conversion from spam email classification to graph classification," *Scientific Programming*, vol. 2022, pp. 1–8, Jan. 2022, doi: 10.1155/2022/6737080.

[19] N. Minakawa, K. Izumi, Y. Murayama, and H. Sakaji, "Firm default prediction by GNN with gravity-model informed neighbor node sampling," *The Review of Socionetwork Strategies*, vol. 18, no. 2, pp. 303–328, 2024, doi: 10.1007/s12626-024-00170-6.

[20] R. Angelova and G. Weikum, "Graph-based text classification," in *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, New York, USA: ACM, Aug. 2006, pp. 485–492, doi: 10.1145/1148170.1148254.

[21] K. A. Qureshi and M. Sabih, "Un-compromised credibility: social media based multi-class hate speech classification for text," *IEEE Access*, vol. 9, pp. 109465–109477, 2021, doi: 10.1109/ACCESS.2021.3101977.

[22] M. Adnan, M. O. Imam, M. F. Javed, and I. Murtza, "Improving spam email classification accuracy using ensemble techniques: a stacking approach," *International Journal of Information Security*, vol. 23, no. 1, pp. 505–517, 2024, doi: 10.1007/s10207-023-00756-1.

[23] A. B. A. Hassanat, "Two-point-based binary search trees for accelerating big data classification using KNN," *PLOS ONE*, vol. 13, no. 11, Nov. 2018, doi: 10.1371/journal.pone.0207772.

[24] S. Lee, C. Lee, K. G. Mun, and D. Kim, "Decision tree algorithm considering distances between classes," *IEEE Access*, vol. 10, pp. 69750–69756, 2022, doi: 10.1109/ACCESS.2022.3187172.

[25] G. Sonowal, "Phishing email detection based on binary search feature selection," *SN Computer Science*, vol. 1, no. 4, Jul. 2020, doi: 10.1007/s42979-020-00194-z.

[26] R. Paredes and E. Chávez, "Using the k-nearest neighbor graph for proximity searching in metric spaces," in *String Processing and Information Retrieval-(SPIRE 2005)*, Berlin, Heidelberg: Springer, 2005, pp. 127–138, doi: 10.1007/11575832_14.

[27] L. C. Shimomura and D. S. Kaster, "Proximity graphs for similarity searches: experimental survey and the new connected-partition approach HGraph," in *Anais Estendidos do XXXVI Simpósio Brasileiro de Banco de Dados (SBBD Estendido 2021)*, Sociedade Brasileira de Computação - SBC, Oct. 2021, pp. 171–176, doi: 10.5753/sbbd_estendido.2021.18181.

[28] Y. Malkov, A. Ponomarenko, A. Logvinov, and V. Krylov, "Approximate nearest neighbor algorithm based on navigable small world graphs," *Information Systems*, vol. 45, pp. 61–68, Sep. 2014, doi: 10.1016/j.is.2013.10.006.

[29] K. B. Aruna and M. M. Kodabagi, "A novel node similarity measure for efficient email classification," in *2023 2nd International Conference for Innovation in Technology*, IEEE, Mar. 2023, pp. 1–5, doi: 10.1109/INOCON57975.2023.10101323.

[30] M. AlShaikh, Y. Alrajeh, S. Alamri, S. Melhem, and A. A.-Khadrah, "Supervised methods of machine learning for email classification: a literature survey," *Systems Science & Control Engineering*, vol. 13, no. 1, Dec. 2025, doi: 10.1080/21642583.2025.2474450.

[31] K. Maharana, S. Mondal, and B. Nemade, "A review: data pre-processing and data augmentation techniques," *Global Transitions Proceedings*, vol. 3, no. 1, pp. 91–99, Jun. 2022, doi: 10.1016/j.gltp.2022.04.020.

[32] W. Zhang, T. Yoshida, and X. Tang, "A comparative study of TF*IDF, LSI and multi-words for text classification," *Expert Systems with Applications*, vol. 38, no. 3, pp. 2758–2765, Mar. 2011, doi: 10.1016/j.eswa.2010.08.066.

[33] A. G. K. Janecek, W. N. Gansterer, M. A. Demel, and G. F. Ecker, "On the relationship between feature selection and classification accuracy," in *JMLR: Workshop and Conference Proceedings*, vol. 4, 2008, pp. 90–105.

[34] M. Hina, M. Ali, A. R. Javed, F. Ghabban, L. A. Khan, and Z. Jalil, "SeFACED: semantic-based forensic analysis and classification of e-mail data using deep learning," *IEEE Access*, vol. 9, pp. 98398–98411, 2021, doi: 10.1109/ACCESS.2021.3095730.

[35] P. Pallavi and D. R. Sarvamangala, "Kannada text summarization using extractive technique," in *2022 IEEE International Conference on Electronics, Computing and Communication Technologies*, IEEE, Jul. 2022, pp. 1–4. doi: 10.1109/CONECCT55679.2022.9865107.

[36] A. K. B and M. M. Kodabagi, "Feature engineering with sentence similarity using the longest common subsequence for email classification," *Malaysian Journal of Computer Science*, pp. 65–78, Dec. 2022, doi: 10.22452/mjcs.sp2022no2.6.

[37] R.-C. Chen, C. Dewi, S.-W. Huang, and R. E. Caraka, "Selecting critical features for data classification based on machine learning methods," *Journal of Big Data*, vol. 7, no. 1, Dec. 2020, doi: 10.1186/s40537-020-00327-4.

[38] S. S. Patil and S. P. Sonavane, "Improved classification of large imbalanced data sets using rationalized technique: updated class purity maximization over_sampling technique (UCPMOT)," *Journal of Big Data*, vol. 4, no. 1, Dec. 2017, doi: 10.1186/s40537-017-0108-1.

[39] M. Hina, M. Ali, A. R. Javed, G. Srivastava, T. R. Gadekallu, and Z. Jalil, "Email classification and forensics analysis using machine learning," in *2021 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/IOP/SCI)*, IEEE, Oct. 2021, pp. 630–635, doi: 10.1109/SWC50871.2021.00093.

[40] Y. Chai *et al.*, "MalFSCIL: a few-shot class-incremental learning approach for malware detection," *IEEE Transactions on Information Forensics and Security*, vol. 20, pp. 2999–3014, 2025, doi: 10.1109/TIFS.2024.3516565.

BIOGRAPHIES OF AUTHORS

Aruna Kumara B. holds a Ph.D. in a Computer Science and Engineering with a research focus on natural language processing and graph theory for efficient email classification systems from REVA University, Bangalore, Karnataka, India in 2023. He earned a Bachelor's Degree and Master's Degree in Computer Science and Engineering from Visvesvaraya Technological University, Belgaum, India. He has 14+ years of academic and research experience, is currently working as an Associate Professor in the Department of Computer Science and Engineering, Proudhadevarya Institute of Technology, Hosapete, Karnataka, India. He has published 543 numerous research articles in reputed international journals and conferences. His research areas of expertise include artificial intelligence, natural language processing, machine learning, deep learning, and data analytics. He can be contacted at email: arunakumara.b11@gmail.com.

Dr. Madan H. T. holds a Ph.D. in Electronics and Communication Engineering from REVA University, Bangalore, Karnataka, India in 2022. He earned a Bachelor's Degree and Master's Degree in Computer Science and Engineering from Visveswaraya Technological University, Belgaum, India. He is currently working as an Associate Professor at NMAM Institute of Technology, Nitte affiliated with NITTE (Deemed to be University), Karnataka, India. With over 14 years of teaching experience and 8 years of research expertise, he has published several articles in the area of robotics, artificial intelligence, and wireless communication. He can be contacted at email: madan.ht@nitte.edu.in.

Dr. Rashmi C. holds a Ph.D. in a Computer Science and Engineering with a research focus on natural language processing from REVA University, Bangalore, Karnataka, India in 2023. She earned a Bachelor's Degree and Master's Degree in Computer Science and Engineering from Visvesvaraya Technological University, Belgaum, India. She is currently an Assistant Professor, in the School of Computing and Information Technology, REVA University, Bangalore. She has a total of 18 years of teaching experience. Her current research areas include social network analysis, machine learning, artificial intelligence. She has organized several international conferences as an organizing chair and session chair and reviewed many articles of IEEE journals. She is a member of professional bodies like IEEE, CSI, and ISTE. She can be contacted at email: rashmi.c31@gmail.com.

Dr. Sarvamangala D. R. holds a Ph.D. degree in Computer Science and Engineering from Visvesvaraya Technological University, Belgaum, Karnataka, India 2021. She earned a Bachelor's Degree and Master's Degree in Computer Science and Engineering from Visvesvaraya Technological University, Belgaum, India. She is currently working as Professor, in the School of Computing and Information Technology, REVA University, Bangalore. She has a total of 22 years of teaching experience. Her current research areas include digital image processing, computational intelligence, machine learning, and large language models. She is a member of professional bodies like IEEE, CSI, and ISTE. She can be contacted at email: sarvamangala.hita@gmail.com.