

Classification of Kannada documents using novel semantic symbolic representation and selection method

Ranganathbabu Kasturi Rangan¹, Bukahally Somashekar Harish²,
Chaluvegowda Kanakalakshmi Roopa²

¹Department of Information Science and Engineering, Vidyavardhaka College of Engineering, Mysore, India

²Department of Information Science and Engineering, JSS Science and Technology University, Mysore, India

Article Info

Article history:

Received Apr 21, 2024

Revised Feb 25, 2025

Accepted Mar 15, 2025

Keywords:

Classification

Feature selection

Kannada documents

Semantic analysis

Symbolic representation

ABSTRACT

Kannada is one of the 22 scheduled Indian regional languages. It is also a low-resource regional language. The Kannada document classification is arduous due to its vocabulary richness, agglutinative terms, and lack of resources. The good representation and the prominent feature selection aid in solving the challenges in document classification tasks. In this paper, we are proposing semantic symbolic representation and feature selection method, for better representation of Kannada terms in interval values embedded with positional information. Following, selection of prominent discriminative symbolic feature vectors is also proposed. Further the symbolic document classifier is used to classify the Kannada documents. The proposed cluster based symbolic representation preserves the intra class variance and reduces the ambiguity in classification of Kannada documents. The experiments are performed over two Kannada document datasets which are multilabel and unbalanced. The comparative analysis of proposed method with other standard methods is also presented.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Ranganathbabu Kasturi Rangan

Department of Information Science and Engineering, Vidyavardhaka College of Engineering

Gokulam, Mysore, Karnataka, India

Email: rkrangan@vvce.ac.in

1. INTRODUCTION

In the multilingual supportive digital world, natural language processing research is not confined to English. Many natural language applications are developed for various regional languages to avoid digital language divide between dominant languages and others. Kannada is one of the Indian regional languages and one of the 22 scheduled languages in Indian constitution. Kannada text is morphologically rich and agglutinative in nature. Therefore, proper representation of these texts makes a significant contribution in natural language understanding tasks.

In general, for the task of Kannada document classification, at first the raw Kannada documents are preprocessed. In preprocessing, the raw dataset is cleaned by removing punctuation, terms are tokenized, Stopwords are removed, transliteration, stemming and lemmatization (if required) are performed. Secondly, preprocessed data should be represented by using better representation methods. Further vital features are selected through feature selection methods [1] and classifiers are applied to learn the data. At the last stage, learning of the model is evaluated with test samples. This standard process is depicted in Figure 1.

In Kannada document classification better documents understanding leads to better results. The proposed semantic based symbolic representation preserves the contextual information and understands the intra class variations. The optimum unit for text representation and categorization in automatic Kannada

document classification is the term. Unfortunately, a text document lacks the rigid structure of a traditional database even if it can express a wide range of information. Unstructured data must be converted into structured data, especially free-running text data. Numerous preprocessing strategies are suggested in the literature to accomplish this. Once unstructured data is structured, we must create a powerful representation model to create a powerful classification system. The literature contains a wide variety of representational schemes.

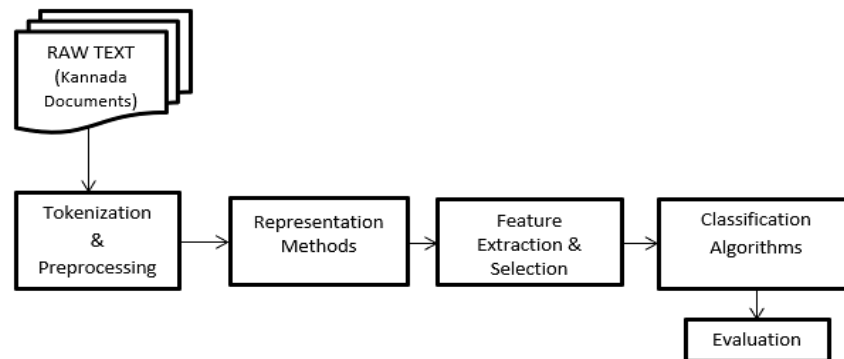


Figure 1. The standard process of Kannada documents classification

Although there are several models for representing text documents in the literature, the frequency-based vector space model produces good outcomes when used to classify texts. Unfortunately, this representational method has its own drawbacks. High dimension, loss of correlation, and loss of semantic link between terms in a document are few of them. Additionally, we must solve the term's complex morphology and agglutination problem in regional Indian languages like Kannada. All these above-mentioned challenges are addressed with various linguistic, statistical and machine learning methods in the proposed model. The main challenge is finding the ideal representation for the raw Kannada terms and its documents. The complex composition of Kannada term letters is represented numerically by universal coded character set (unicode) term encoding [2]. Further to address the loss of semantic information in the frequency-based vector space, the positional information of Kannada terms is embedded. This leads to the positionally encoded frequency-based representation of Kannada documents. Vaswani *et al.* [3] worked on attention-based transformers used this position encoding technique to get better outcomes.

Later, to address the challenge of preserving the intraclass variations, cluster based symbolic representation [4] is employed. In this representation the clustering techniques are used for finding the relations between documents and terms with respect to their classes. The intraclass variation of each feature is represented by the interval value rather than crisp values. In the proposed method this symbolic representation is used on the positionally encoded frequency-based representation [5], which leads to semantic based symbolic representation for Kannada documents [6]. Furthermore, due to the chances of features presence in multiple classes (interclass variations), ambiguity still prevails. Hence, it is important to choose features appropriately so that there is less overlap across different classes [7], [8]. Here, the correlation based symbolic feature selection is also applied for the dimensionality reduction. As formerly mentioned, in this article we have detailed descriptions of the following contributions:

- For the Kannada documents, semantic embedded symbolic representation is proposed.
- Proposed symbolic feature selection method on semantic-symbolic representation of Kannada documents.
- Classifier for interval valued representation of Kannada documents is proposed.
- Comparative analysis of stacked ensemble feature selection with symbolic feature selection for Kannada documents classification.

Further, section 2 presents the literature review with respect to representation and selection methods. The proposed methodology is presented in section 3 in detail. Later, the datasets and experimentations performed on those datasets are explained in section 4. Further, the comparative analysis is presented and concluded in section 5. The future scopes of these findings are also presented in section 5.

2. RELATED WORK

Language resources are crucial for tasks involving natural language processing. Low resource languages are those spoken in many regions of India that lack the resources needed for language processing activities. It is possible to do language processing tasks at the character, sentence, paragraph, or document levels. Researchers have focused more on character and sentence level work than document level work due to

the lack of regional language resources. In language identification task, to determine whether phrases in the tweeter dataset are in Hindi or English, Ansari *et al.* [1] used the chi-square feature selection method. To perform aspect-based sentiment analysis for Hindi reviews at the word level, Gandhi and Attar [9] presented category association word (CAW) features ensemble algorithm and achieved 76% of accuracy. Anand *et al.* [10] used a fuzzy-based convolutional network for feature selection together with ensemble learning methods to address the problem of multilingual offensive language detection and achieved 98% accuracy. The accuracy of authorship identification task for Kannada literature was 88%, and this was accomplished using stylometry features and a profile-based technique in [11].

To address the high dimensionality challenge at the document level language processing tasks, it is necessary to choose the most vital subset of features [12], [13]. There are basic approaches like filters and wrappers, for feature selection but ensemble technique performs better. Ensemble is the combination of basic feature selection methods in various ways it may be homogeneous or heterogeneous [14]. Homogeneous is the combination of the same feature selection method with different parameters but heterogeneous is the combination of the different feature selection techniques and yields better results [15]. Tian *et al.* [16] presented ensemble-based filter feature selection (heterogeneous strategy) using feature ranking methods like information gain, gain ratio, chi-squared, and ReliefF for proper feature selection. Wang *et al.* [17] used genetic algorithm to select the best ranking features. There are more heterogeneous feature selection ensembles, and its examples can be found in [18]–[21]. These homogeneous and heterogeneous techniques are analyzed in [21], [22] research articles. In the experiments section, the findings of one of the heterogeneous ensemble methods applied to Kannada documents are discussed in comparison with the outcomes of the proposed method.

Researchers have recently explored widely with character recognition for the Kannada language. As there are fewer corpora available, experiments at the document level are limited. On the Kannada-MNIST dataset, Gu [23] work with the Kannada character recognition problem. With 98.77% accuracy, the convolutional neural networks (CNN) model excels. Trishala and Mamatha [24] presented unsupervised Kannada terms stemmer and Kannada terms rule-based lemmatizer. They built a corpus of 17,825 Kannada root words for the experimentation. Additionally, Chandrakala and Thippeswamy [25] proposed historical handwritten Kannada stone inscription recognition and categorization of the 11th century. The characters were classified using two separate classification algorithms like stochastic gradient descent with momentum (SGDM) and support vector machine (SVM), using the features collected by the deep convolutional neural network (DCNN), and 70% accuracy was attained.

In the study of text classification, clustering is used as a different representation technique for text documents. There have been several clustering strategies put forth. These clusters take advantage of the relationship between documents and key terms. Sun *et al.* [26] addressed the imbalanced data classification by the adaptive weighted k-nearest neighbors (AWKNN) method which uses similarity-based feature clustering. The researchers [27]–[30] worked on the information bottleneck method and two-dimensional clustering algorithms, which help in the clustering of terms based on the distribution of each term's class labels. Further authors in [31], [32] worked on feature extraction using a clustering algorithm from the combination of labeled and unlabeled data. Authors in [33], [34] worked on a word embedding approach for dimensionality reduction leading to better feature selection. Towards semantic based representation, authors in [35], [36] presented term weighting technique. It is based on term's semantic similarity, which is computed using WordNet. Due to its complexity in computation, it shows lower performance than standard term weighting methods. Positional encoding is used by many attention-based models like BERT [37], RoBERTa [38], and GPT-2 [39]. Absolute or relative positional information of the term is extracted in positional encoding technique [40]. To extend transformers to tree domain activities (particularly binary trees), Sun *et al.* [41] provide a novel framework of customized positional encodings. Gehring *et al.* [42] proposed convolutional sequence to sequence learning model. They used positional encoding to extract sequence information of terms, for the translation task. In this way, representational approaches employ positional encoding. The literature indicates that semantic and symbolic representations are not explored for Kannada documents classification. Further, there is also need of discussion on Kannada documents classification experiments based on symbolic feature selection and classifications with other state-of-the-art learning algorithms.

3. PROPOSED METHOD

The raw Kannada text documents are classified into multiple categories based on the semanticity, symbolic representation and selection (SRS) method. This proposed process of representation (embedded with semantic information) and the symbolic feature selection method for Kannada document classification. It is depicted in Figure 2.

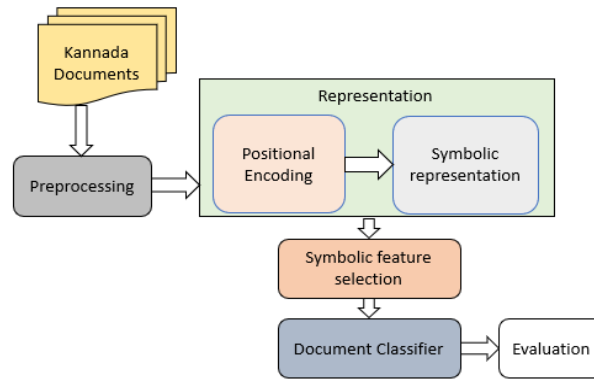


Figure 2. The proposed method of representation and selection of Kannada documents for classification task

In preprocessing tokenization, punctuation and stopwords removal tasks are performed. At first, the problem of term representation of regional language is addressed. Each Kannada term is represented by a unique decimal number by unicode encoding method. This is discussed in the section as follows.

3.1. Unicode encoding for Kannada term

To get around the discordance of ASCII values encoding for characters from languages other than English, there is a character set called unicode. Every character is encoded uniquely in unicode using a special number called a code-point. “\uXXXX,” is the representation of each code-point, here ‘u’ indicates that the value is a code-point, and ‘XXXX’ is the four-digit hexadecimal value. In the proposed experiments, we discovered that several agglutinative / morphologically rich term characters faded when text-processing activities are conducted directly on these Kannada terms. This results in feature information loss. In order to retain the meaning of each Kannada term intact and avoid the need for extra language corpora, we need a unicode-encoded decimal representation for each term [2]. An example is shown in Table 1. For example: A term in the Kannada language: “ಮನುಷ್ಯ” (English translation: human being).

Table 1. Unicode encoded Kannada term representation

Kannada characters of the term	ಮ (Ma)	ನ (Na)	ು	ಷ (sa)	್ಯ (Ya)	
Unicode code-points	\u0cae	\u0ca8	\u0cc1	\u0cb7	\u0ccd	\u0caf
Encoding of code-points (UTF-16)	b'\xff\xfe\xae\x0c\xa8\x0c\xc1\x0c\xb7\x0c\xcd\x0c\xaf\x0c'					
Decimal value	257257805393772252295682176515839					

3.2. Semantic representation

Following unicode encoding, document term matrix is used to represent Kannada documents in vector space model. The values of the term frequency (TF) or term frequency-inverse document frequency (TF-IDF) are included in the document term matrix [43]. Positional encoding is integrated with TF or TF-IDF to solve the problem of the absence of sequence information or semantic information.

Positional encoding maintains the sequence order of the terms. For an example, if input sentence is of length ‘T’, and to extract the ‘ k^{th} ’ term positional information in the input sequence, the positional encoding is calculated as shown in (1) and (2) using sine and cosine functions.

$$P.E(pos_k, 2i) = \sin\left(\frac{pos_k}{n^{2i/d}}\right) \quad (1)$$

$$P.E(pos_k, 2i + 1) = \cos\left(\frac{pos_k}{n^{2i/d}}\right) \quad (2)$$

Where “ pos_k ” is k^{th} object position, “ n ” is user defined scalar value set to 10,000 based on empirical results [3], dimension of output embedding space is represented by “ d ” and “ i ” is the index ranges between $0 \leq i < d/2$. The positional encoded (PE) values should also be convoluted, which means that the sine and cosine values of each term should be added as given in (3) and also presented in Algorithm 1.

$$V_k = \sin(x_k) + \cos(x_k) \quad (3)$$

Algorithm 1: Positional encoding

Input: Length of document, the output embedding value.

Data: PE = Positional encoding, $n=10000$ standard empirically determined value [3]

Output: Document's positional encoded matrix.

```

STEP 1: for  $k$  in range(length of document)
STEP 2:     for  $i$  in range(output embedding / 2)
STEP 3:          $PE_{(k,2i)} = \sin(\frac{k}{n^{2i/output\ embedding}})$ 
STEP 4:          $PE_{(k,2i+1)} = \cos(\frac{k}{n^{2i/output\ embedding}})$ 
STEP 5:          $PE_k = PE_{k,2i} + PE_{k,2i+1}$ 
STEP 6:     end
STEP 7: end

```

The vector space goes through shallow shifts because of the convolution [5]. In a document, the same phrase may appear in various positions. These positions of the same term are combined, and their means are calculated, as presented in (4) (m is Term's repetitive count in a document).

$$(\sum_{j=0}^m V_{jk})/m \quad (4)$$

The mean value obtained from (4) is embedded to the TF or TF-IDF values of k^{th} term in document term matrix as shown in (5) and (6).

$$TW_k = tf_k + ((\sum_{j=0}^m V_{jk})/m) \quad (5)$$

$$TW_k = tf_k \cdot \log \frac{N}{df_k} + ((\sum_{j=0}^m V_{jk})/m) \quad (6)$$

The obtained attention / semantic based term weights (TW_k) of k^{th} term from (5) and (6) is updated in document term matrix.

3.3. Cluster based symbolic representation

There will be significant intra-class variances in the semantic based TF vectors with regard to each class. As a result, an effective representation is created by using clustering to capture the variances and symbolizing each cluster with an interval-valued feature vector. Let there be L classes each with N documents, and each with a t dimensional TF vector to describe it. Let's say X is the proposed semantic based document term matrix (s-DTM) of size $(LN * t)$, where each row represents a document labelled with a class, and terms are represented in the matrix columns. The dimensionality reduction technique regularized locality preserving index (RLPI) [7] is applied on X resulting in reduced s-DTM Y represented as $(LN \times m)$, where m is selected vital features from total t features. Next, in the reduced s-DTM matrix, based on TF vectors training documents are clustered within each class. Let $[D_1, D_2, D_3, \dots, D_n]$ is a document cluster of n samples belonging to l^{th} class say C_j^l ; $j = 1, 2, 3, \dots, P$ (P denotes number of clusters) and $l = 1, 2, 3, \dots, L$. Further, $F_i = [f_{i1}, f_{i2}, \dots, f_{im}]$ be m features set, describing a sample D_i document, which belongs to C_j^l cluster. Further, for each k^{th} feature value belonging to the j^{th} cluster is represented by interval value $[f_{jk}^-, f_{jk}^+]$ to capture the intra class variations. The interval $[f_{jk}^-, f_{jk}^+]$ represents the ceiling and flooring values of a feature belong to a document cluster. Later, for a cluster C_j^l the reference document is represented by the interval feature values of the features $k = 1, 2, 3, \dots, m$ as shown in (7).

$$RF_j^l = \{[f_{j1}^-, f_{j1}^+], [f_{j2}^-, f_{j2}^+], \dots, [f_{jm}^-, f_{jm}^+]\} \quad (7)$$

In (7), the document clusters of class ' l ' are indexed by $j = 1, 2, 3, \dots, P$. It should be emphasized that, in contrast to traditional feature vectors, this one is an interval valued feature vector that is recorded in the knowledge base as a representation for the j^{th} cluster. This generates P number of symbolic vectors that reflect the class-specific clusters. As a result, we will be obtaining total $(L \times P)$ representative vectors for L classes in the dataset.

3.4. Symbolic feature selection

From the s-DTM, it is important to choose the best interval features that have the least amount of class overlap because these overlapping of interval features between classes reduce the classification accuracy. The aim of the symbolic feature selection is to select maximum variance features from each class. Hence, we create a proximity matrix of $(L \times P) \times (L \times P)$ size and each element are multivalued of m dimension features. In (8) results the similarity between the classes i and j with respect to k^{th} feature.

$$L_{i \rightarrow j}^k = \left(\frac{|I_{ik} \cap I_{jk}|}{|I_{jk}|} \right) \quad (8)$$

In (8), $I_{ik} = [f_{ik}^-, f_{ik}^+] \forall k = 1, 2, \dots, m$ are interval features of class i and similarly I_{jk} is for class j . Now, from the proximity matrix, the matrix M of size: $(L \times P)^2 \times m$ is built by listing multivalued type elements in rows. Further, the highest correlation features will be selected as the best features. The total correlations ($TCorr_k$) of k^{th} column with other y^{th} columns are calculated, and it is compared with average correlation value ($AvgTCorr_k$) as shown in (9) and (10). If $TCorr_k$ is higher than $AvgTCorr_k$ then those k^{th} column features are selected because we are interested in features with a high degree of discrimination.

$$TCorr_k = \sum_{y=0}^m \text{Corr}(k^{th} \text{Column}, y^{th} \text{Column}) \quad (9)$$

$$AvgTCorr_k = \sum_{k=0}^m TCorr_k / m \quad (10)$$

3.5. Symbolic classifier

The test document consists of m features with crisp values but we have representation with interval feature values of the respective cluster to compare and classify. Hence the classification will be performed based on degree of belongingness. For the test document, let $F_t = [f_{t1}, f_{t2}, \dots, f_{tm}]$ be a m dimensional feature vector. Let RF_j^l is the reference document of j^{th} cluster of l^{th} class with interval values. Each m^{th} feature value is compared with the corresponding intervals of RF_j^l . The degree of belongingness will be determined by the number of features whose values fall inside the corresponding interval. If the value falls inside the interval, then count is 1 else 0. Belongingness count B_c is used to determine the class label for the test document as shown in (11) and (12).

$$B_c = \sum_{k=1}^m C(f_{tk}, [f_{jk}^-, f_{jk}^+]) \quad (11)$$

$$C(f_{tk}, [f_{jk}^-, f_{jk}^+]) = \begin{cases} 1; & \text{if } (f_{tk} \geq f_{jk}^- \text{ and } f_{tk} \leq f_{jk}^+) \\ 0; & \text{Otherwise} \end{cases} \quad (12)$$

Belongingness count is computed for all clusters of all classes. Later the test document class label is predicted based on the class having the highest B_c . In this way the Kannada document classification task is performed. The experimental results with comparison of other representational methods and selection methods are discussed in next section.

4. EXPERIMENTATIONS WITH RESULTS

The subsequent section presents the information on the datasets used, experimentations carried out and comparison of the proposed feature selection methods.

4.1. The datasets

The Indian regional language, Kannada is less resourced specially at the document level. The proposed model is applied on the following two Kannada document datasets. The first version of dataset (small) has 300 Kannada documents of different sections. This small dataset contains 5 categories like space, politics, crime, sports, and economics. This dataset is a subset of the larger dataset presented further. The second dataset is a larger dataset [6] which contains 11,045 documents that are unevenly distributed among 10 categories. The details of these datasets are as shown in Figures 3 and 4.

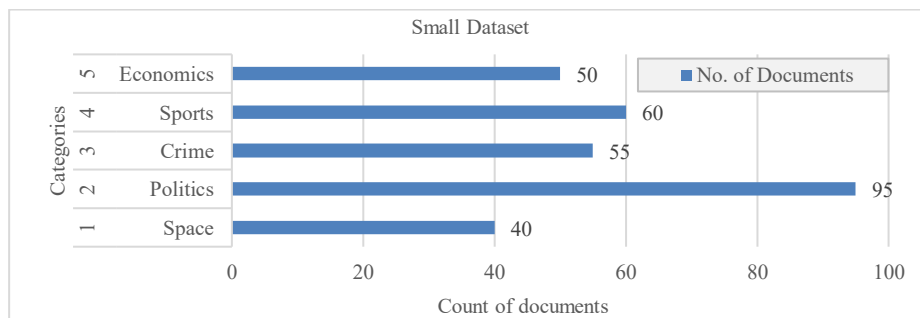


Figure 3. Details of Kannada documents dataset (smaller)

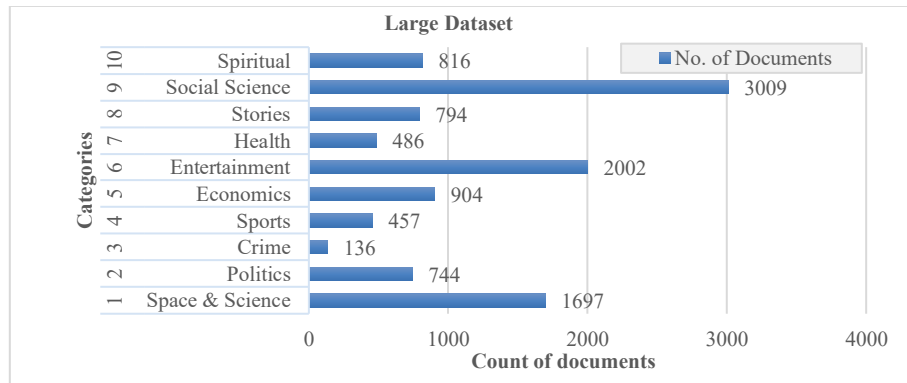


Figure 4. Details of Kannada documents dataset (larger)

4.2. Experimentations

The raw Kannada documents are preprocessed by removing punctuations, algebraic numbers, and stopwords (frequency based). Tokenized Kannada terms are unicode encoded as discussed in former section. Further, based on the positional encoding method the terms sequential information is extracted and embedded into the document term matrix. This leads to s-DTM X of size $(LN \times t)$. The RLPI is applied for dimensionality reduction and hence X transforms to Y of size $(LN \times m)$. RLPI [7] is chosen because it discovers the document space's discriminating structure.

Further, the experiments are conducted over the dataset split ratio of 50:50 and 60:40. As aforementioned, RLPI is applied to select m features ranging from 1 to 15 dimensions. Following, the proposed cluster based symbolic representation is applied for the training set, resulting in symbolic vectors for each class. Fuzzy C means (FCM) algorithm is used for clustering due to its strength of identifying the clusters and even the boundaries are overlapping in the data. The number of clusters in each experimentation is empirically decided. Later symbolic feature selection is applied to select optimal features subset, and symbolic document classifier is used for classification of test documents. The number of clusters in each experimentation is empirically decided. Each experiment is repeated 3 times, and minimum accuracy, maximum accuracy, and average accuracy is noted as shown in Tables 2 to 4. The experiment results are tabulated for various representations like SRS_TF, SRS_TF-IDF, SRS_PE-TF-IDF.

In Table 2, Kannada documents are classified by using proposed symbolic document classifier. In this work, the documents are represented using SRS_TF vectors. For the small dataset of 60:40 split ratio, with 3 clusters of documents in each class, resulted 85.57% of average accuracy. Similarly for large dataset, the 60:40 split ratio, with 3 clusters of documents in each class, resulted 84.69% of average accuracy. Further, with respect to SRS_TF-IDF document vectors' results are tabulated in Table 3. Here, for both small and large datasets, 60:40 split ratio with 3 document clusters at each class yielded 86.65% and 85.90% of average accuracy respectively. In Table 4, classification accuracy of symbolic document classifier is presented for the proposed method (semantic symbolic representation) SRS_PE_TF-IDF. Among 50:50 and 60:40 train-test splits, 60:40 split experiments with 3 clusters for both datasets yielded highest results with 89.10 and 87.65% average accuracy respectively.

Table 2. Classification accuracy of the symbolic document classifier on Kannada document datasets using SRS_TF

Dataset	Training vs Testing	Number of clusters	Minimum accuracy	Maximum accuracy	Average accuracy
Small dataset	50 vs 50	1	72.62	76.52	75.20
	60 vs 40	1	74.95	79.40	76.85
	50 vs 50	2	69.19	72.05	70.66
	60 vs 40	2	72.56	75.34	74.13
	50 vs 50	3	76.25	80.26	79.78
	60 vs 40	3	83.26	87.25	85.57
	50 vs 50	4	75.64	78.65	76.32
	60 vs 40	4	76.58	79.59	78.60
	50 vs 50	1	65.63	69.17	67.38
	60 vs 40	1	68.88	70.45	69.84
Large dataset	50 vs 50	2	71.38	75.39	74.99
	60 vs 40	2	72.56	76.45	74.97
	50 vs 50	3	76.52	79.45	78.61
	60 vs 40	3	81.26	85.57	84.69
	50 vs 50	4	68.26	71.52	68.95
	60 vs 40	4	70.56	74.52	72.26

Table 3. Classification accuracy of the symbolic document classifier on Kannada document datasets using SRS_TF-IDF

Dataset	Training vs Testing	Number of clusters	Minimum accuracy	Maximum accuracy	Average accuracy
Small dataset	50 vs 50	1	70.38	73.26	72.19
	60 vs 40	1	71.95	73.95	72.85
	50 vs 50	2	71.58	75.05	74.65
	60 vs 40	2	74.69	78.34	76.32
	50 vs 50	3	78.50	82.55	80.87
	60 vs 40	3	85.55	88.60	86.65
	50 vs 50	4	80.45	82.05	81.25
	60 vs 40	4	81.50	83.90	82.55
Large dataset	50 vs 50	1	68.85	70.25	69.50
	60 vs 40	1	70.65	73.95	72.40
	50 vs 50	2	72.40	76.39	74.50
	60 vs 40	2	74.60	78.85	77.65
	50 vs 50	3	79.68	83.58	82.50
	60 vs 40	3	83.50	86.90	85.90
	50 vs 50	4	70.30	72.10	71.55
	60 vs 40	4	74.60	77.25	76.55

Table 4. Classification accuracy of the symbolic document classifier on Kannada document datasets using SRS_PE-TF-IDF

Dataset	Training vs Testing	Number of clusters	Minimum accuracy	Maximum accuracy	Average accuracy
Small dataset	50 vs 50	1	74.95	77.65	76.50
	60 vs 40	1	76.10	78.58	77.10
	50 vs 50	2	73.26	75.05	74.12
	60 vs 40	2	74.65	77.45	75.50
	50 vs 50	3	81.90	84.60	83.65
	60 vs 40	3	87.10	90.25	89.10
	50 vs 50	4	82.65	83.15	82.95
	60 vs 40	4	84.60	86.85	85.95
Large dataset	50 vs 50	1	70.50	72.65	71.35
	60 vs 40	1	72.64	75.15	73.56
	50 vs 50	2	75.20	78.65	77.65
	60 vs 40	2	77.10	79.25	78.35
	50 vs 50	3	80.65	83.55	82.45
	60 vs 40	3	84.95	88.25	87.65
	50 vs 50	4	71.50	73.25	72.65
	60 vs 40	4	73.55	75.50	74.20

The various state-of-the-art machine learning classifiers results are tabulated in Tables 5 and 6. The classifiers like decision tree (DT), k-nearest neighbor (KNN), SVM with various kernels, rule-based classifier and the proposed symbolic classifier are compared. A special observation is noted for 50:50 train-test split ratio of large dataset. The proposed classifier yielded marginally high accuracy of 82.50% for SRS_TF-IDF than 82.45% of accuracy for SRS_PE-TF-IDF. But for 60:40 train-test split of large dataset, SRS_PE-TF-IDF yields 87.65% of accuracy, which is higher than SRS_TF-IDF. This observation reflects that more training samples contribute to the better semantic information analysis. For both small and large Kannada document datasets. The proposed classifier yields better results in all variants of symbolic representations. The semantic symbolic representation yields best average accuracy of 89.10% and 87.65% using symbolic classifier which is applied on small and large datasets respectively.

Table 5. Comparative analysis of the symbolic classifier with other classifiers using 50:50 ratio

Classifier	SRS_TF		SRS_TF-IDF		SRS_PE-TF-IDF	
	Small dataset	Large dataset	Small dataset	Large dataset	Small dataset	Large dataset
DT	64.50	62.35	69.55	67.45	71.55	70.20
KNN classifier	74.25	70.25	76.55	73.55	77.85	75.20
SVM - linear	76.55	74.60	78.95	75.15	79.65	76.50
SVM - RBF	71.40	70.55	73.65	71.05	76.55	72.20
SVM - sigmoid	78.90	76.30	80.15	76.95	80.94	78.60
SVM -polynomial	74.64	70.21	76.54	72.88	78.54	76.69
Rule based classifier	69.58	67.56	71.45	68.33	73.69	71.52
Symbolic classifier	79.78	78.61	80.87	82.50	83.65	82.45

Table 6. Comparative analysis of the symbolic classifier with other classifiers using 60:40 ratio

Classifier	SRS_TF		SRS_TF-IDF		SRS_PE-TF-IDF	
	Small dataset	Large dataset	Small dataset	Large dataset	Small dataset	Large dataset
DT	66.36	63.49	71.36	70.83	73.58	71.55
KNN classifier	77.84	73.55	78.36	76.59	79.64	78.65
SVM - linear	79.83	77.54	81.23	80.45	83.65	81.74
SVM - RBF	74.65	72.15	74.65	71.94	78.46	76.94
SVM - sigmoid	81.26	80.24	83.56	82.55	84.33	83.16
SVM -polynomial	76.92	74.56	79.55	76.54	81.76	80.38
Rule based classifier	70.12	69.15	74.36	71.55	76.58	72.66
Symbolic classifier	85.57	84.69	86.65	85.90	89.10	87.65

4.3. Comparison of proposed symbolic representation and selection with stacked ensemble feature selection

To overcome various shortcomings of conventional feature selection methods, ensemble feature selection methods are proposed. In ensemble we can find the right blending of various feature selection methods. As this is the extended work of stacked ensemble feature selection on Kannada documents [44], we discuss the comparison of results between the proposed semantic symbolic feature selection and the stacked ensemble feature selection methods. In stacked ensemble feature selection method, there are two layers. First layer consists of chi-square and mutual information gain statistical methods. Following in the second layer we have XGBoost method. Selected features of first layer are given as input for the second layer to further identify the most discriminative features. Through this ensemble of feature selection methods vital features are identified and used for the Kannada documents classification.

The proposed SRS (SRS_PE-TF-IDF) is compared with stacked ensemble feature selection by applying SVM, KNN, and DT classifiers for the large dataset split of 60:40 train-test split. The same is presented in Figure 5. Rather than the crisp feature values, interval value features yield better results. Here symbolic classifier is not used for comparison because stacked ensemble features are crisp valued. Among the aforementioned classifiers, SVM does better with both feature selection methods. Further, the proposed symbolic feature selection SRS_PE-TF-IDF yields significant increase in average accuracy of 83.16% when compared to stacked ensemble feature selection method for SVM classifier.

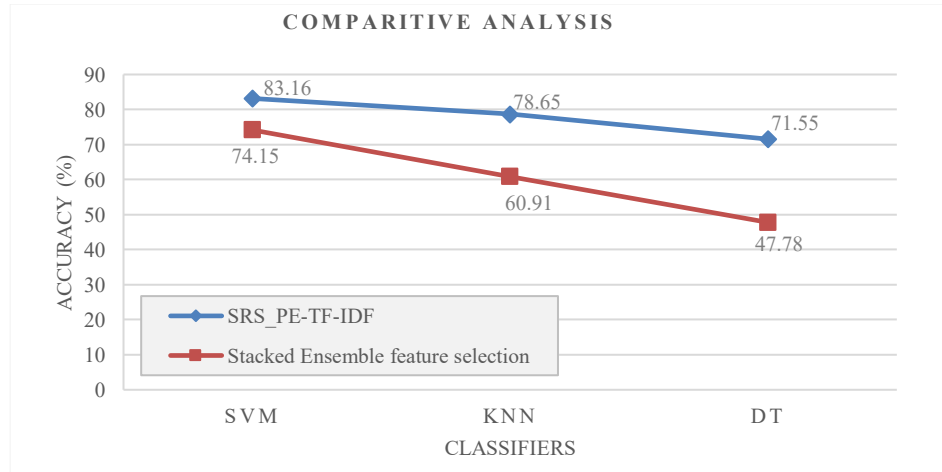


Figure 5. Comparison of proposed SRS with stacked ensemble feature selection

5. CONCLUSION WITH FUTURE SCOPE

It is evident from all the experimental results, that semantic symbolic representation with symbolic feature selection and symbolic document classifier results better in the Kannada document classification task. The proposed experiments reveal that the interval data representation aids in storing the intra class variance information. The positional encoding embeds the terms sequence positional information and helps in storing attention or semantic based information. At the document level of natural language processing tasks, dimensionality is one of the major challenges. To address the dimensionality reduction the symbolic feature selection is proposed, and it resulted in better accuracy for Kannada documents classification. From all experiments, the proposed representation and selection approach (SRS_PE-TF-IDF) resulted in the highest average accuracy of 87.65% for the 60:40 train-test split of a large dataset. The proposed methods also obtain

an average accuracy of 89.10% for the small dataset with a 60:40 train-test split ratio, which is higher than that of existing state-of-the-art methods. Since Kannada is a low resource language, the proposed methods could be used for newly constructed, larger Kannada document collections in future work. Further, as the dataset is unbalanced, the experimentations could be extended to K-Fold validations. After the creation of larger dataset, then the semantic symbolic feature vectors behaviors could be analyzed with various neural network classifiers in future. These proposed methods could also be applied for other natural language processing tasks like named entity recognition, sentiment analysis at paragraph level, and summarization.

FUNDING INFORMATION

Authors state there is no funding involved.

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
Ranganathbabu Kasturi Rangan	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓				
Bukahally Somashekar Harish		✓	✓	✓		✓	✓			✓	✓	✓	✓	
Chaluvegowda Kanakalakshmi Roopa	✓			✓	✓		✓			✓	✓	✓	✓	

C : Conceptualization

M : Methodology

So : Software

Va : Validation

Fo : Formal analysis

I : Investigation

R : Resources

D : Data Curation

O : Writing - Original Draft

E : Writing - Review & Editing

Vi : Visualization

Su : Supervision

P : Project administration

Fu : Funding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

The data that support the findings of this study are openly available in Kaggle repository at <https://doi.org/10.34740/kaggle/dsv/7376871>.

REFERENCES

- [1] M. Z. Ansari, T. Ahmad, and A. Fatima, "Feature selection on noisy Twitter short text messages for language identification," *arXiv-Computer Science*, pp. 1–19, 2020.
- [2] R. K. Rangan and B. S. Harish, "Kannada document classification using unicode term encoding over vector space," in *Recent Advances in Artificial Intelligence and Data Engineering*, 2022, pp. 387–400, doi: 10.1007/978-981-16-3342-3_31.
- [3] A. Vaswani *et al.*, "Attention is all you need," in *31st Conference on Neural Information Processing Systems (NIPS 2017)*, California, United States, 2017, pp. 1–11.
- [4] B. S. Harish, D. S. Guru, S. Manjunath, and R. Dinesh, "Cluster based symbolic representation and feature selection for text classification," in *Advanced Data Mining and Applications*, 2010, pp. 158–166, doi: 10.1007/978-3-642-17313-4_16.
- [5] R. K. Rangan, B. S. Harish, and C. K. Roopa, "Semantic term weighting representation for Kannada document classification," *Revue d'Intelligence Artificielle*, vol. 38, no. 4, pp. 1243–1253, 2024, doi: 10.18280/ria.380418.
- [6] R. K. Rangan, "Kannada documents for classification (KDC): Kannada documents dataset for NLP tasks," *Kaggle*. 2024. [Online]. Available: <https://www.kaggle.com/datasets/rkasturirangan/kannada-documents-for-classification-kdc>
- [7] D. Cai, X. He, W. V. Zhang, and J. Han, "Regularized locality preserving indexing via spectral regression," in *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, New York, United States: ACM, 2007, pp. 741–750, doi: 10.1145/1321440.1321544.
- [8] W. Li, H. Zhou, W. Xu, X.-Z. Wang, and W. Pedrycz, "Interval dominance-based feature selection for interval-valued ordered data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 10, pp. 6898–6912, Oct. 2023, doi: 10.1109/TNNLS.2022.3184120.
- [9] H. Gandhi and V. Attar, "Sentiment of primary features in aspect based sentiment analysis of hindi reviews," in *Applied Computational Technologies*, Singapore: Springer, 2022, pp. 567–578, doi: 10.1007/978-981-19-2719-5_54.
- [10] M. Anand, K. B. Sahay, M. A. Ahmed, D. Sultan, R. R. Chandan, and B. Singh, "Deep learning and natural language processing in computation for offensive language detection in online social networks by feature selection and ensemble classification techniques," *Theoretical Computer Science*, vol. 943, pp. 203–218, 2023, doi: 10.1016/j.tcs.2022.06.020.




Classification of Kannada documents using novel semantic symbolic ... (Ranganathbabu Kasturi Rangan)

- [11] C. P. Chandrika and J. S. Kallimani, "Authorship attribution for Kannada text using profile based approach," in *Proceedings of the 2nd International Conference on Recent Trends in Machine Learning, IoT, Smart Cities and Applications*, 2022, pp. 679–688, doi: 10.1007/978-981-16-6407-6_58.
- [12] A. Hashemi, M. B. Dowlatshahi, and H. Nezamabadi-Pour, "A bipartite matching-based feature selection for multi-label learning," *International Journal of Machine Learning and Cybernetics*, vol. 12, no. 2, pp. 459–475, 2021, doi: 10.1007/s13042-020-01180-w.
- [13] A. Hashemi, M. B. Dowlatshahi, and H. Nezamabadi-Pour, "An efficient Pareto-based feature selection algorithm for multi-label classification," *Information Sciences*, vol. 581, pp. 428–447, 2021, doi: 10.1016/j.ins.2021.09.052.
- [14] A. Hashemi and M. B. Dowlatshahi, "MLCR: a fast multi-label feature selection method based on K-means and L2-norm," in *2020 25th International Computer Conference, Computer Society of Iran (CSICC)*, 2020, pp. 1–7, doi: 10.1109/CSICC49403.2020.9050104.
- [15] V. Bolón-Canedo and A. Alonso-Betanzos, "Ensembles for feature selection: A review and future trends," *Information Fusion*, vol. 52, pp. 1–12, 2019, doi: 10.1016/j.inffus.2018.11.008.
- [16] Y. Tian, J. Zhang, J. Wang, Y. Geng, and X. Wang, "Robust human activity recognition using single accelerometer via wavelet energy spectrum features and ensemble feature selection," *Systems Science and Control Engineering*, vol. 8, no. 1, pp. 83–96, 2020, doi: 10.1080/21642583.2020.1723142.
- [17] H. Wang, C. He, and Z. Li, "A new ensemble feature selection approach based on genetic algorithm," *Soft Computing*, vol. 24, no. 20, pp. 15811–15820, 2020, doi: 10.1007/s00500-020-04911-x.
- [18] B. Seijo-Pardo, I. Porto-Díaz, V. Bolón-Canedo, and A. Alonso-Betanzos, "Ensemble feature selection: Homogeneous and heterogeneous approaches," *Knowledge-Based Systems*, vol. 118, pp. 124–139, 2017, doi: 10.1016/j.knosys.2016.11.017.
- [19] B. Seijo-Pardo, V. Bolón-Canedo, and A. Alonso-Betanzos, "On developing an automatic threshold applied to feature selection ensembles," *Information Fusion*, vol. 45, pp. 227–245, 2019, doi: 10.1016/j.inffus.2018.02.007.
- [20] V. Bolón-Canedo, N. Sánchez-Maróño, and A. Alonso-Betanzos, "An ensemble of filters and classifiers for microarray data classification," *Pattern Recognition*, vol. 45, no. 1, pp. 531–539, 2012, doi: 10.1016/j.patcog.2011.06.006.
- [21] D. S. Guru, M. Suhil, S. K. Pavithra, and G. R. Priya, "Ensemble of feature selection methods for text classification: An analytical study," *Advances in Intelligent Systems and Computing*, vol. 736, pp. 337–349, 2018, doi: 10.1007/978-3-319-76348-4_33.
- [22] A. B. Brahimi and M. Limam, "Ensemble feature selection for high dimensional data: a new method and a comparative study," *Advances in Data Analysis and Classification*, vol. 12, no. 4, pp. 937–952, 2018, doi: 10.1007/s11634-017-0285-y.
- [23] E. X. Gu, "Convolutional neural network based Kannada-MNIST classification," in *2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE)*, 2021, pp. 180–185, doi: 10.1109/ICCECE51280.2021.9342474.
- [24] G. Trishala and H. R. Mamatha, "Implementation of stemmer and lemmatizer for a low-resource language—Kannada," in *Proceedings of International Conference on Intelligent Computing, Information and Control Systems*, Singapore: Springer, 2021, pp. 345–358, doi: 10.1007/978-981-15-8443-5_28.
- [25] H. T. Chandrakala and G. Thippeswamy, "Deep convolutional neural networks for recognition of historical handwritten Kannada characters," *Advances in Intelligent Systems and Computing*, Singapore: Springer, pp. 69–77, Oct. 2020, doi: 10.1007/978-981-13-9920-6_7.
- [26] L. Sun, J. Zhang, W. Ding, and J. Xu, "Feature reduction for imbalanced data classification using similarity-based feature clustering with adaptive weighted K-nearest neighbors," *Information Sciences*, vol. 593, pp. 591–613, 2022, doi: 10.1016/j.ins.2022.02.004.
- [27] F. Pereira, N. Tishby, and L. Lee, "Distributional clustering of English words," in *Proceedings of the 31st annual meeting on Association for Computational Linguistics*, Morristown, United States: Association for Computational Linguistics, 1993, pp. 183–190, doi: 10.3115/981574.981598.
- [28] N. Slonim and N. Tishby, "The power of word clusters for text classification," *23rd European Colloquium on Information Retrieval Research*, vol. 1, pp. 1–12, 2001.
- [29] I. S. Dhillon, S. Mallela, and R. Kumar, "Enhanced word clustering for hierarchical text classification," in *Proceedings of the eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2002, pp. 191–200, doi: 10.1145/775047.775076.
- [30] H. Takamura and Y. Matsumoto, "Two-dimensional clustering for text categorization," in *COLING-02: proceedings of the 6th Conference on Natural Language Learning*, 2002, pp. 1–7, doi: 10.3115/1118853.1118881.
- [31] B. Raskutti, H. L. Ferrá, and A. Kowalczyk, "Using unlabelled data for text classification through addition of cluster parameters," in *ICML '02: Proceedings of the Nineteenth International Conference on Machine Learning*, 2002, pp. 514–521.
- [32] H.-J. Zeng, X.-H. Wang, Z. Chen, H. Lu, and W.-Y. Ma, "CBC: clustering based text classification requiring minimal labeled data," in *Third IEEE International Conference on Data Mining*, Melbourne, United States, 2003, pp. 443–450, doi: 10.1109/ICDM.2003.1250951.
- [33] K. N. Singh, S. D. Devi, H. M. Devi, and A. K. Mahanta, "A novel approach for dimension reduction using word embedding: An enhanced text classification approach," *International Journal of Information Management Data Insights*, vol. 2, no. 1, 2022, doi: 10.1016/j.jiime.2022.100061.
- [34] T. Sabri, S. Bahassine, O. El Beggar, and M. Kissi, "An improved Arabic text classification method using word embedding," *International Journal of Electrical and Computer Engineering*, vol. 14, no. 1, pp. 721–731, 2024, doi: 10.11591/ijece.v14i1.pp721-731.
- [35] Q. Luo, E. Chen, and H. Xiong, "A semantic term weighting scheme for text categorization," *Expert Systems with Applications*, vol. 38, no. 10, pp. 12708–12716, 2011, doi: 10.1016/j.eswa.2011.04.058.
- [36] B. Wei, B. Feng, F. He, and X. Fu, "An extended supervised term weighting method for text categorization," in *Proceedings of the International Conference on Human-centric Computing 2011 and Embedded and Multimedia Computing 2011*, Dordrecht, Netherlands: Springer, 2011, pp. 87–99, doi: 10.1007/978-94-007-2105-0_11.
- [37] J. Devlin, M.-W. Chang, K. Lee, K. T. Google, and A. I. Language, "BERT: Pre-training of deep bidirectional transformers for language understanding," *arXiv-Computer Science*, pp. 1–16, 2018.
- [38] Y. Liu *et al.*, "RoBERTa: A robustly optimized BERT pretraining approach," *arXiv-Computer Science*, pp. 1–13, 2019.
- [39] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI*, pp. 1–24, 2018.
- [40] P. Shaw, J. Uszkoreit, and A. Vaswani, "Self-attention with relative position representations," *arXiv-Computer Science*, pp. 1–5, 2018.
- [41] Z. Sun, Q. Zhu, Y. Xiong, Y. Sun, L. Mou, and L. Zhang, "TreeGen: A tree-based transformer architecture for code generation," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 5, pp. 8984–8991, 2020, doi: 10.1609/aaai.v34i05.6430.




- [42] J. Gehring, M. Auli, D. Grangier, D. Yarats, and Y. N. Dauphin, "Convolutional sequence to sequence learning," in *ICML '17: Proceedings of the 34th International Conference on Machine Learning*, 2017, pp. 1243–1252.
- [43] Z. Labd, S. Bahassine, K. Housni, F. Z. A. H. Aadi, and K. Benabbes, "Text classification supervised algorithms with term frequency inverse document frequency and global vectors for word representation: A comparative study," *International Journal of Electrical and Computer Engineering*, vol. 14, no. 1, pp. 589–599, 2024, doi: 10.11591/ijece.v14i1.pp589-599.
- [44] R. K. Rangan, B. S. Harish, and C. K. Roopa, "Stacked ensemble feature selection method for Kannada documents categorization," in *Proceedings of Data Analytics and Management*, 2024, pp. 431–442, doi: 10.1007/978-981-99-6547-2_33.

BIOGRAPHIES OF AUTHORS






Ranganathbabu Kasturi Rangan    obtained B.E. in information science & engineering in 2014, M.Tech. in software engineering in 2016, both from Visvesvaraya Technological University, Karnataka, India. He worked in Intel India Pvt Ltd and Alten Calsoft Labs. Totally he is having 3 years of industry experience and 4 years of teaching experience. Presently working as an Assistant Professor in the Department of Information Science and Engineering, Vidyavardhaka College of Engineering, Mysore. He is a lifetime member of ISTE, and member of ACM association. His article is awarded best paper. His areas of interest include natural language processing, machine learning, and document categorization. He can be contacted at email: rkrangan@vvce.ac.in or rkrangan3@gmail.com.



Dr. Bukahally Somashekar Harish    obtained his Ph.D. in computer science from University of Mysore, India. Presently he is working as a Professor in the Department of Information Science and Engineering, JSS Science and Technology University, India. He was a visiting researcher at DIBRIS - Department of Informatics, Bio Engineering, Robotics and System Engineering, University of Genova, Italy. He has been invited as a resource person to deliver various technical talks on data mining, image processing, pattern recognition, and soft computing. He is serving as a reviewer for international conferences and journals. He has published articles in more than 100+ international reputed peer reviewed journals and conferences proceedings. He successfully executed AICTE-RPS project, which was sanctioned by AICTE, Government of India. His area of interest includes machine learning, text mining, and computational intelligence. He can be contacted at email: bsharish@jssstuniv.in.



Dr. Chaluvegowda Kanakalakshmi Roopa    received her B.E. degree in information science and engineering and M.Tech. degree in computer engineering from Visvesvaraya Technological University, Belagavi, Karnataka, India. She completed her Ph.D. from University of Mysore, India. She is currently working as an Associate Professor at JSS Science and Technology University. She is serving as reviewer for many conferences and journals. She is a lifetime member of ISTE and CSI. Her area of research includes medical image analysis, biometrics, and text mining. She can be contacted at email: ckr@jssstuniv.in.