An Efficient Machine Learning-based Text Summarization in the Malayalam Language

Rosna P Haroon[1*], Abdul Gafur M[2], Barakkath Nisha U[3]

[1]Assistant Professor, Department of CSE, Ilahia College of Engineering and Technology, APJ Abdul Kalam Technological University, Kerala, India.
[1]rosna.haroon@gmail.com
[2] Professor&Principal, Ilahia College of Engineering and Technology, APJ Abdul Kalam Technological University, Kerala, India.
[2]abdulgafurm@gmail.com
[3]Associate Professor, Department of IT, Sri Krishna College of Engineering and Technology, Tamilnadu, India.
[3]ubnisha@gmail.com
*Corresponding author : Rosna P Haroon

Received January 29, 2022; revised March 24, 2022; revised April 23, 2022; accepted May 11, 2022; published June 30, 2022

Abstract

Automatic text summarization is a procedure that packs enormous content into a more limited book that incorporates significant data. Malayalam is one of the toughest languages utilized in certain areas of India, most normally in Kerala and in Lakshadweep. Natural language processing in the Malayalam language is relatively low due to the complexity of the language as well as the scarcity of available resources. In this paper, a way is proposed to deal with the text summarization process in Malayalam documents by training a model based on the Support Vector Machine classification algorithm. Different features of the text are taken into account for training the machine so that the system can output the most important data from the input text. The classifier can classify the most important, important, average, and least significant sentences into separate classes and based on this, the machine will be able to create a summary of the input document. The user can select a compression ratio so that the system will output that much fraction of the summary. The model performance is measured by using different genres of Malayalam documents as well as documents from the same domain. The model is evaluated by considering content evaluation measures precision, recall, F score, and relative utility. Obtained precision and recall value shows that the model is trustable and found to be more relevant compared to the other summarizers.

Keywords: Malayalam Text Summarization, Supervised Machine Learning, SVM, Text Mining, Sentence Extraction, Summary Generation.

http://doi.org/10.3837/tiis.2022.06.001 **ISSN : 1976-7277**

1. Introduction

Summarization plays an important role in our day-to-day life. As all of us are living a very busy scheduled life, most of them wish to get the important data at their fingertips without reading a very large text. With the introduction of Artificial intelligence, the computer can automate any type of human activities and the same will reduce the risk of time management of humans. So, obviously in summarization task also computer is playing a role. As we are dealing with languages in summarization, this is coming under the subarea of AI known as Natural Language Processing. Usually, English is accepted as the universal language for communication. So the majority of the NLP works are focused on the English language. Because of that dataset and corpus availability are more in such languages. But when we consider the other languages it is not the case.

Here we have implemented a learning based extractive summarizer for Malayalam language which one is providing better precision and recall rates compared to other summarizers implemented so far.Moreover a trained summarizer like this is lagging in Malayalam.Here comes the importance of our work.Eventhough the technology have advanced and everything is getting into our finger prints within seconds, still the technology is not reached to the common people in our state due to the language gap.This is one of the main motivation behind the topic.

Text summarization is one of the predominant applications of natural language processing. It is nothing but here the system is finding out the gist of the text given in the document. It can be performed mainly in two ways[19]. One is known as extractive summarization whereby we would be able to get the shortened version of the document by picking the most important sentences from the document. The other is named abstractive summarization where the sentences are regenerated with the help of paraphrasing and natural language generation techniques.

The following example[20] will illustrate the difference between extractive and abstractive summarization:

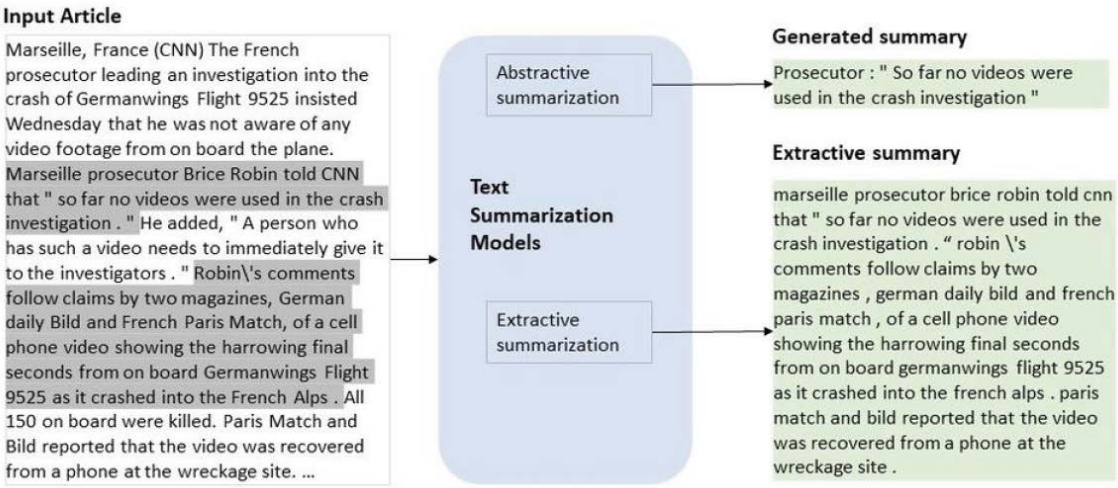

Fig. 1. Extractive and Abstractive summary of a sample text.

As seen in the above example, abstractive summarization produces a more abstractive summary which consists of a sentence conveying the important information in the paragraph given. But in extractive summary, it is producing a summary by extracting the important sentences from the given paragraph. The majority of research works in Malayalam are adopting extractive summarization rather than abstractive. Several approaches including statistical score, semantic graph, etc are used there to find out the most leading sentences from the document. A machine learning-based algorithm is lagging in Malayalam text summarization and we are trying to perform the same by using a supervised machine learning technique.

The major contributions providied by this paper are:

- An efficient extractive summarizer for Malayalam language.
- Machine learning based extractive summarizer.
- An extractive summarizer with better precision and recall rates.

Here we are implementing Support Vector Machine(SVM)-based learning to perform the summarization process. Support Vector Machine is a supervised machine learning classifier which is proved to be an efficient one in many classification tasks.SVMs can be categorized into linear and nonlinear SVMs based on how the hyperplane segregates the data. If it is possible to separate the data by using a straight line it is known as linear type, otherwise, it is nonlinear[18].

2. Related Work

An extensive set of literature about text summarization using machine learning approaches is available for the English language. But for a language like Malayalam, it is not possible to apply the same methods invented for other languages so far. There are lots of syntactic differences between these two languages. So, from the preprocessing phase to the final step, the complication is more in the case of languages like Malayalam. Here we are reviewing machine learning method text summarizers in languages other than Malayalam and some text summarizers available in the Malayalam language.

Joel Larocca Neto et.al.[1] in their paper “Automatic text summarization using machine learning approach” proposing an ML-based classifier for the English language by incorporating the features like mean Term Frequency-Inverse Frequency(TF-ISF), Sentence length, Sentence positionetc.They have employed two classification algorithms namely Naïve Bayes and C4.5 for the training purpose.When comparing the Naïve Bayes and C4.5, Naïve Byes produced better results in compression rates and C4.5 prediction seems to be poor.

Nikitha Desai and Pranchi shah [2] implemented a supervised machine learning model for the Hindi language whereby they have tried to analyze the summarizer system with a different experimental setup. Based on the different combinations of the feature vectors selected, accuracy was calculated and the system shows an average score of 72% in accuracy when taking more features in the feature vector. As the number of features taken into consideration for summarizing the document is increased the system accuracy is also being incremented.

Chintan Shah and Anjali Jivani [3] in “An Automatic Text Summarization on Naive Bayes Classifier Using Latent Semantic Analysis” describe a summarization based on latent semantic analysis and trained using Naïve Bayes classifier. The semantic similarity between text fragments has been measured using Latent Semantic Analysis. Here they are using statistical methods like SVD (Singular Value Decomposition) to show the relationship among

words and sentences. Important concepts are being selected from the SVM model by using recursive feature elimination. Based on the order of elimination, concepts will be ranked. The model is trained using Naïve Bayes classifier.

Nedunchelian Ramanujan et al. proposed a timestamp-based approach with a naïve Bayes classifier for multi-document summarization [4]. Based on the chronological position of the sentences in the document a value is assigned to each sentence and this is taken as the timestamp. Based on the score obtained using the features selected a number of relevant sentences are selected in the summary and the same is ordered using this timestamp value. This will result in an ordered and coherent summary. They have also done a comparative based study of proposed methods by using MEAD platform including this timestamp approach.

In [5], authors have implemented an extractive text summarizer using deep learning modified neural network classifier. Here entropy value is calculated for each relevant feature and the value is classified into two classes namely the highest entropy value and the lowest entropy value. Those sentences coming in the highest entropy class are taken in summary output. The dataset used for the performance analysis is Document Understanding Conference (DUC) Dataset and the performance is varying depending on the file size they have taken. The result shows that this method scores a higher accuracy rate compared to other Artificial Neural Network schemes. Different methods for machine learning approaches for text summarization have been discussed in [6][7]. Authors have listed out different methodologies used so far in a tabular form along with the dataset used and remarks.

For Malayalam documents, summarization works are very few. Implemented works have been focused on statistical scoring[8] and graph-based approaches[9]. Vector space model for Malayalam Summarizer [10] proposed a statistical method for extractive summarization by prioritizing the sentences with the help of cosine similarity. The highest scored sentences will be sorted out in the summary. A graph-based method for Malayalam documents has been proposed in [11] where the sentences are represented as nodes and vertex weight is calculated using similarity measures. Minimum spanning tree Malayalam summarizer [12] creates a semantic graph from the input document and thereby graph reduction is performed using minimum spanning tree concept by creating repetitive subgraphs.

A clustering technique using self-organizing maps are also been implemented for Malayalam summary in paper [13] whereby an extractive summarizer has developed by scoring the sentence based on relevance analysis and context-aware measures and formed a cluster using SOM. Relevant sentences are selected from the clusters using the algorithm proposed by the researcher. Both theoretical and practical evaluations are done in this method to check the accuracy of the model.

Evaluation of text summarizer is also important in determining the accuracy of the output generated. There are intrinsic and extrinsic measures for summarization. Text quality evaluation and content evaluation are coming under intrinsic and task-based evaluation schemes like question answering, information retrieval, etc. are coming under extrinsic. Quality in terms of grammar, non-redundancy, referential clarity, structure, and coherence is being considered in text quality evaluation techniques [14][15]. For a summarizer system, the most important evaluation measure to be considered is its content evaluation. The measures like precision, recall, F-score, relative utility, Rouge N-gram matching, etc. are the most frequent measures taken by the researchers to evaluate their system. ROUGE (Recall Oriented Understudy of Gisting Evaluation) is an often-used evaluation strategy where consecutive tokens are considered for comparison.Overlap of N-grams in human evaluated summary and system computed summary are taken into account and computing the ROUGE score. If a high overlap is there, the score will be more. Here N may be 1, 2 or more and based on this the

measure will be ROUGE-1,ROUGE-2 etc.But this is not suited well for abstractive summarizers since semantic meaning and factual accuracy are not been considering in Rouge. Other than N-grams, alternatives like the longest common subsequence can also be considered in Rouge evaluation [19].

A multi document summarization system with statistical score features incorporated with modified page ranking algorithm is proposed in[21]. After getting the summary of each document,it is subjected to Maximum Marginal Relevance to get the final summary.An abstractive summarizer for Malayalam with the help of attention mechanism is proposed in [22]. Here it produces regenerated sentences in the summary,but it doesn't support long range dependency between sentences.

A robust document similarity metric is proposed in[23], by which they are doing the clustering of documents.For the similarity measure of documents this may contribute in summarization works also. Three way clustering scheme is used in[24] to find out the relationship between data items and clusters.A multi view clustering technique by customizing the K-means algorithm is also suggesting in this paper.[25] also describing a multi view data clustering scheme with the help of non negative matrix factorization and a solution is proposed from diverse views by preserving the geometrical structure of the data.

From the literature works done, it is evident that no one tried a trained model for Malayalam extractive text summarization. The proposed model focuses on such a training model using an SVM classification algorithm[16][17] to select the prioritized sentences for summary output. From the related work study,it is seen that support vector machine provides better performance compared to other classification algorithms.Evaluation measures using relative utility is also been considering here to determine the correct accuracy of the output, which was not been done by other Malayalam NLP researchers.

Table 1. Summary of Text summarization Papers in Literature Review

Methodology Proposed	Datasets Used	Measurement Mertic Used
Machine learning based method(Naïve Bayes,SVM) for English/Hindi language [1][2][3][4]	Manual(200 Documents)[1] HindiNews domain (130 articles)[2] Manual(Text corpus from different articles,10Nos)[3] Manual(20 Documents)	Precision and Recall[1][4] Accuracy by counting correctly classified sentence[2] ROUGE 2.0 Evaluation kit
Sentence Ranking Method[8]	50 selected News Articles in Malayalam	ROUGE-1 and ROUGE-2
Graph based method/Minimum Spanning Tree[9][11][12]	Manual[9][11][12]	Precision,Recall and F-score[9][11][12]
Vector Space Model[10]	Manual	Precision,Recall
Self organizing maps and entity recognition[13]	Manual(Articles from Manoramaonline)	Sentence rank evaluation,question game evaluation, keyword association
Hierarchical encoder/decoder architecture[19]	CNN/Daily mail Data Set	ROUGE-1,ROUGE-2 and ROUGE-L

Multi document summarization with statistical score and MMR[21]	Manual	ROUGE-1 and ROUGE-2
Attention based Mechanism for abstractive summary[22]	Translated BBC News Repository	ROUGE-1,ROUGE-2 and ROUGE-L

3. Proposed Methodology

A machine learning-based text summarizer for the language Malayalam is proposed here. The framework we are discussing is for the single input document. As far as a machine learning model is concerned, the accuracy will depend on the quality of the training we have given to the model as well as the learning algorithm we have adopted. The system is trained with the Support Vector Machine algorithm.

The following diagram shows the architecture of the machine learning-based text summarizer. The input can be given as a text document and the system outputs an extractive summary of the input concerned. The document given as input firstly undergoes a preprocessing phase which includes the process of text segmentation, tokenization, stop word removal, and stemming. The following mentioned features are extracted from the segmented units and the machine is trained with those features in order to predict the exact output summary as a human is doing.

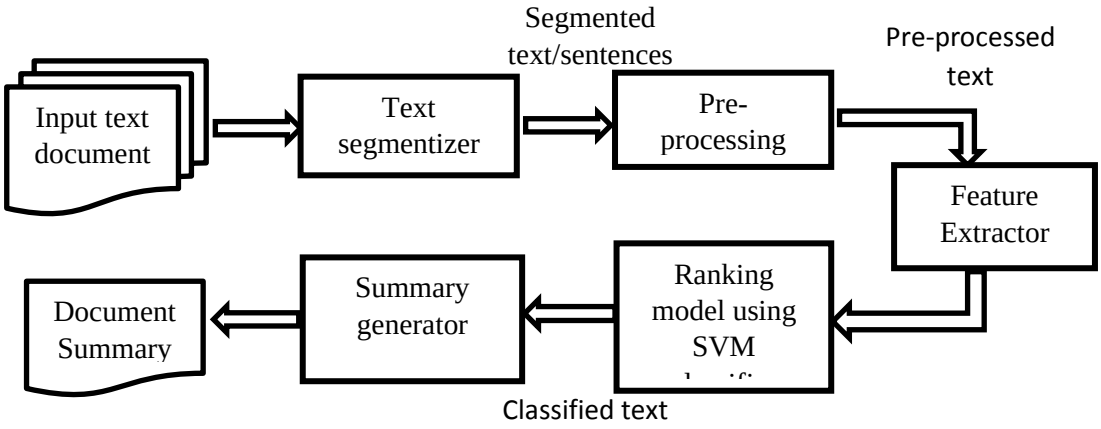

Fig. 2. Architecture of the proposed model

3.1 Text Segmentizer

The accepted input document is segmentized into different sentences here. The sentences can be identified from the document by giving a sentence boundary condition.

ഹിന്ദു, ബുദ്ധ ഐതിഹ്യങ്ങളിൽ കാണുന്ന ഭീമാകാരമായ പക്ഷിയാണ് ഗരുഡൻ. ഹിന്ദുപുരാണങ്ങളിലെ വിഷ്ണുവിന്റെ വാഹനമാണ് ഗരുഡൻ. കൃഷ്ണപ്പരുന്തിനെ ഗരുഡന്റെ ഒരു രൂപമായും കണക്കാക്കുന്നുണ്ട്.കാശ്യപൻ തന്റെ പത്നിമാരായ കദ്രുവിനോടും വിനതയോടും എങ്ങനെയുള്ള സന്താനങ്ങളെയാണവശ്യമെന്ന് തിരക്കുന്നു. ശക്തൻമാരായ ആയിരം പുത്രന്മാരെ കാംക്ഷിച്ച കദ്രുവിന് ജനിക്കുന്ന സന്താനങ്ങളാണ് വാസുകി, അനന്തൻ തുടങ്ങിയ നാഗങ്ങൾ. വിനതയാവട്ടെ കദ്രുവിന്റെ ആയിരം പുത്രന്മാരെക്കാൾ ശക്തരായ രണ്ടു മക്കളെ ആവശ്യപ്പെടുന്നു. തുടർന്ന് വിനതയ്ക്കുണ്ടാവുന്ന രണ്ട് അണ്ഡങ്ങൾ കാലമേറെയായിട്ടും വിരിയാത്തതു കണ്ട് അക്ഷമ മൂത്ത് വിനത ഒരു മുട്ട പൊട്ടിക്കുന്നു.അതിൽ നിന്നും പകുതി മാത്രം വളർന്ന അരുണൻ ജന്മമെടുക്കുന്നു. അക്ഷമ കാണിച്ച അമ്മയെ കദ്രുവിന്റെ ദാസിയാവട്ടെ എന്നു ശപിച്ച് അരുണൻ പോകുന്നു. രണ്ടാമത്തെ മുട്ട വിരിഞ്ഞുണ്ടായ പുത്രനാണ് ഗരുഡൻ. അതിനിടെ കദ്രുവുമായി ഒരു പന്തയത്തിലേർപ്പെടുന്ന വിനത നാഗങ്ങളുടെ ചതിപ്രയോഗം കാരണം തോറ്റ് കദ്രുവിന്റെ ദാസിയായി മാറുന്നു. ദാസ്യം ഒഴിവാക്കാനായി നാഗങ്ങളുടെ ആവശ്യപ്രകാരം അമൃത് കൈക്കലാക്കാൻ ശ്രമിക്കുന്നതിനിടെ ഇന്ദ്രനെ തോൽപ്പിക്കുന്നു. അമ്മയെ ദാസ്യവ്യത്തിയിൽ നിന്നും മോചിപ്പിക്കുന്നു. പിന്നീട് ഗരുഡൻ ഭഗവാൻ വിഷ്ണുവിന്റെ വാഹനമാകുന്നു.

Fig. 3. sample input to text segmentizer

ഹിന്ദു, ബുദ്ധ ഐതിഹ്യങ്ങളിൽ കാണുന്ന ഭീമാകാരമായ പക്ഷിയാണ് ഗരുഡൻ.

ഹിന്ദുപുരാണങ്ങളിലെ വിഷ്ണുവിന്റെ വാഹനമാണ് ഗരുഡൻ.

.....

.....

പിന്നീട് ഗരുഡൻ ഭഗവാൻ വിഷ്ണുവിന്റെ വാഹനമാകുന്നു.

Fig. 4. output of text segmentizer for the input given in **Fig 3.1.1**

3.2 Pre-Processing Module

Sentences extracted from the text segmentizer are subjected to preprocessing tasks. The following tasks are performed in preprocessing stage:

1. **Tokenization:** Tokens are nothing but it is the basic building units of the natural language. This may be words, sub-words, or characters. For example, when considering the sentence,
   ഹിന്ദുപുരാണങ്ങളിലെ വിഷ്ണുവിന്റെ വാഹനമാണ് ഗരുഡൻ.
   Word-level tokens are:
   ഹിന്ദുപുരാണങ്ങളിലെ , വിഷ്ണുവിന്റെ , വാഹനമാണ് , ഗരുഡൻ ,
   . .
   sub word token can be like: വാഹനം, ആണ്
   character level will be like: വാ-ഹ-ന-മാ-ണ്
   word-level tokenization is performed here for the proposed work.

2. **Stop word Removal:** There may be certain words in the document which may not provide valuable meaning to the sentence and more often come as a grammatical construct. These types of words may remove from the text since this will cause extra storage and also more processing time. The words such as ആണ്, അത്, ഇത്, അങ്ങനെ etc. are removed from the text for reducing the complexity in space and time.

3. **Stemming:** Words may come in different inflected forms. Stemming helps us to find the base form of a word without any inflections. For example, വാഹനമാണ് will result in the stem വാഹനം.

In the case of the Malayalam Language, stemming is difficult since it is an agglutinative language. That means we can append more and more affixes to a particular word and by removing one affix still, it must be a syntactic word. Here we are using a separate data corpus to find out the stem of different words.

3.3 Feature Extractor

After the preprocessing task has been done, the features mentioned below are extracted from the text. The following section deals with the features used to train the model. As text summarization is concerned, we can take so many features including text as well as statistical features for training the text summarizer model. Our text summarizer is trained with the following features:

Step 1: *Number of key phrases in the Sentence*

The most occurring words/phrases in the sentences are called key phrases. A ranking is given to the sentences based on the key phrases present. The ranking can be computed by taking the ratio of the number of key phrases in the sentences to the total number of words/phrases in the longest sentence occurring in the text document.

Step 2: *Position of the sentence in the input document*

The locality of the sentence within the document has a significant role in the case of the summarization process. Usually, human beings have a nature that the important concept will be organized in the initial paragraph positions and the conclusion part will be given in the last paragraph of the document. More weight will be given to those sentences which are coming under this category.

Step 3: *Position of the sentence in the paragraph*

The other feature we have taken considers the overall position of sentences within the document. But in this feature, we are considering the locality of the sentence within the paragraph itself. As we mentioned earlier, the initial sentences in the paragraph will reflect conceptually more, so the training model will give more importance to those sentences which are coming first in the paragraph.

Step 4: *Numerical information in the sentence*

When taking as index terms we are not giving good credit to numbers since they are hazy without a surrounding context. But regarding text summary, numerical data plays a major role since that may represent a date, year, or any important count. In such cases that should be included in the summary report definitely. The segmented sentences from the document which contains the numerical information are ranked as a ratio of the number of numerical data in the sentence concerned to the total number of words in the sentence concerned.

Step 5: *Presence of guillemets in the sentence*

The presence of quotation marks is also a salient feature in producing text summaries. For a language like Malayalam, the important concepts are usually been quoted and the same cannot be avoided in the output of a summary document. So, such units are also been ranked based on how many words are quoted out of total words in the sentence.

Step 6: *Length of the sentence*

A score is being added to each sentence by examining the number of words in the sentence taken and the number of words in the longest sentence of the document. Generally, the shorter sentences may not convey more information. Similarly, the sentences with more length also give a short weight since they may contain more unnecessary extravasation.

3.4 Ranking Model & Summary Generator

Based on the features extracted from the previous step, the model will be trained to classify the sentences into different groups namely VVI (Very-Very Important), VI (Very Important) I (Important), and LI (Least Important). Support Vector Machine is the algorithm used here for classification. A score is being calculated based on the feature obtained from the feature extractor module. The relevance of the sentences can be found out by using this score and sentences will be clustered into four classes in view of the priority of the text segment. The summary document is produced with the most relevant sentences formed by the ranking module. Percentage of the summary to be produced can be given from the user end. Based on this threshold value that many numbers of sentences will be selected from the four classes VVI, VI, I, LI respectively ranked by the training model. The following algorithm describes the entire process in detail.

Algorithm 1

Input: Text document of any genres.

Method: SVM classifier-based algorithm for training the model

Output: Extracted text document summary

```
Begin
        Input the text document
        Perform preprocessing:
                Break the document into separate sentences
                        For (sen=1; sen<last sentence;sen++)
                                Split the sentence into tokens
                                If (token=stopword)
                                        Remove from tokens.
                                Perform stemming.
                                Sentence ranking ( );
                                Classify ( );
                                Summarygen ( );
End
```

Subroutine 1- Sentence ranking ()

```
Begin
        Int length, w, max_length=m, score;
                        If ( key phrases) then
                                score=score+1;
                        If(position= “top”)
                                score=score+1;
                        If(numeric terms(‘0 to 9’)
                                score=score+1;
                        if( “ “ or  ‘ !’)
                                score=score+1;
                        w= count(space)+1;
length=  w/m;
score=score+length;
End
```

Subroutine 2- Classify ()

```
Begin
        Sort the sentences in descending order of score;
        Assign the sentence to different classes VVI, VI, I, and LI based on the score obtained.
End
```

Subroutine 3- Summarygen()

```
Begin
        float cr;
        int ns,ts;
        ts=total number of sentences in input document;
        Accepting compression ratio cr from the user;
        Calculate the number of sentences to be extracted as
                ns=ts/cr;
        select the sentences from class VVIP until ns>=ns in VVIP class.
        If ns is not reached the limit, select the sentences from the VIP class
        If ns is not reached the limit, select the sentences from the IP class
If ns is not reached the limit, select the sentences from the LIP class
End
```

4. Implementation of Summarizer

The summarizer is implemented here by using Python programming language. An interface is also developed by using a web framework Django which enables us to summarize the document in the simplest way. Python language is one of the best options for natural language processing tasks since it contains so many NLP tools and libraries which helps the programmer to pre-process the unstructured input text in an easy way. It also gives the support to integrate with other languages and moreover the syntax of the language is so easy and the same can be easily understandable for anyone including a beginner in the programming field.

Initially, we are selecting the document which is to be summarized. Consider the following document as the input document.

വിഷയദാരിദ്ര്യവും വിഭവദാരിദ്ര്യവുംകൊണ്ട് പ്രതിപക്ഷത്തിന് എത്രത്തോളം തരംതാഴാനാകും എന്നതിന്റെ ഉദാഹരണമാണ് അവർ സ്പീക്കർക്കെതിരെ അവതരിപ്പിച്ച പ്രമേയം. കേട്ടുകേൾവികളുടെയും അപവാദങ്ങളുടെയും പിൻബലത്തിൽ ഒരു നിയമസഭയുടെ അധ്യക്ഷവേദിക്കെതിരെ ഇന്ത്യയിൽ ആദ്യമായി പ്രമേയം കൊണ്ടുവന്ന പ്രതിപക്ഷമെന്നും പ്രതിപക്ഷ നേതാവെന്നുമാണ് ചരിത്രം രേഖപ്പെടുത്തുക. അപകടമായ ഈ രീതി ഭാവിയിലും ആവർത്തിച്ചാൽ അത് ജനാധിപത്യത്തെ ദുർബലപ്പെടുത്തും.

നിയമസഭയുടെ അധ്യക്ഷനായി തെരഞ്ഞെടുക്കപ്പെട്ടപ്പോൾ ഞാൻ നെഞ്ചോളിയന്റെ വാക്കുകൾ പറയുകയുണ്ടായി. "വെൻ ഐ വാസ് ഇൻ ഈജിപ്ത് ഐ വാസ് എ മൂസ്‌ലിം. നൗ ഐ ആം ഇൻ ഇസ്രയേൽ ഡഫ്നിറ്റ്‌ലി ഐ ആം ക്രൂസ്ത്യൻ" ഈ വാക്കുകൾ പാലിക്കാൻ പരമാവധി ശ്രമിച്ചു. നാലേമുക്കാൽ വർഷം അധ്യക്ഷനിൽനിന്ന് നീതി കിട്ടിയില്ല എന്ന ഒരു പരാതിയും ഈ സഭയിൽ വന്നിട്ടില്ല. പ്രതിപക്ഷത്തിന്റെ ശബ്ദത്തിന് കൂറവുണ്ടാകാതിരിക്കാൻ ആകുന്നവിധം പരിശ്രമിച്ചു. ഇപ്പോൾ "അങ്ങാടിയിൽ തോറ്റതിന് അമ്മയോട്" എന്ന പഴഞ്ചൊല്ലിനെ അനുസ്മരിപ്പിക്കുന്ന രീതിയാണ് ഓർമവരുന്നത്. ഈ സർക്കാരിനെതിരെ ഒന്നും പറയാനില്ലാത്ത പ്രതിപക്ഷത്തിന്റെ ഈ അടി സ്വയം തിരിച്ചടിക്കുന്ന ബൂമറാങ്ങ് ആണ്.

എന്താണ് സഭാധ്യക്ഷൻ ചെയ്ത തെറ്റ്? ലെജിസ്ലേച്ചർ ആക്ടിവിസം എന്ന തലത്തിലേക്ക് നമ്മുടെ നിയമസഭയെ ഉയർത്താൻ ശ്രമിച്ചതോ? മാറുന്ന കാലത്തോടു ചേർന്നുനിൽക്കാൻ പദ്ധതികൾ ആവിഷ്കരിച്ചതോ? ഭരണഘടന വെല്ലുവിളികൾ നേരിടുമ്പോൾ ഭരണഘടനാ ക്ലാസുകളിലൂടെ ബദൽ പ്രതിരോധം ഉയർത്തിക്കൊണ്ടുവന്നതോ? ഇന്ത്യയിൽ ആദ്യമായി നിയമനിർമാണത്തിൽ ജനകീയ പങ്കാളിത്തത്തിന് തുടക്കം കുറിച്ചതോ? നിയമസഭയ്ക്ക് ഇന്ത്യയിൽ ആദ്യമായി ബദൽ മാധ്യമം കൊണ്ടുവന്നതോ? നിയമസഭാ പ്രവർത്തനങ്ങളെ എക്കാലത്തും ലഭ്യമാക്കാൻ മാധ്യമരംഗത്തെ പുതിയ അനുഭവമായ ഒടിടി പ്ലാറ്റ്ഫോം രാജ്യത്ത് ആദ്യമായി തുടക്കം കുറിച്ചതോ? നിർമിച്ച നിയമങ്ങളുടെ അനുഭവങ്ങളെയും പരിമിതികളെയും വിശകലനം ചെയ്യുന്ന ഇംപാക്ട് സ്റ്റഡിക്ക് തുടക്കം കുറിച്ചതിനോ? 21 ഗ്രന്ഥത്തിലൂടെ നിയമനിർമാണത്തിന്റെ ചരിത്രംവിശകലനത്തിന് വിധേയമാക്കിയതോ? ചട്ടങ്ങളില്ലാത്ത നിയമങ്ങൾ എന്ന അവസ്ഥയ്ക്ക് അറുതിവരുത്താൻ നേതൃത്വം കൊടുത്തതോ? ഗ്രീൻ പ്രോട്ടോകോളിലൂടെ നിയമസഭാ സമുച്ചയത്തെ ജൈവജീവിതത്തിന്റെ ഭാഗമാക്കിയതോ? ഇതെല്ലാം തെറ്റാണെങ്കിൽ ആ തെറ്റ് ഏറ്റെടുക്കാൻ തയ്യാറാണ്.

ജനാധിപത്യത്തിന്റെ അന്തസ്സത്തയ്ക്ക് ഇടിവു പറ്റുന്ന ഒരു കാലത്തിന്റെ കരിനിഴൽ ഇന്ന് ഇന്ത്യയിലുണ്ട്. ഭരണസംവിധാനത്തിന്റെ എല്ലാ ഘടകത്തെയും കൈപ്പിടിയിൽ ഒതുക്കുന്ന അസാധാരണമായ നീക്കം. ജനാധിപത്യത്തിന്റെ മധുരത്തെത്തന്നെ ഇത് ഇല്ലാതാക്കുമെന്ന് ഏത് രാഷ്ട്രീയ വിദ്യാർഥിക്കും മനസ്സിലാകും. ഈ സാഹചര്യത്തിൽ നിയമനിർമാണ സഭകൾക്ക് പരമ്പരാഗത ചുമതലകൾ അല്ലാതെ പലതും ചെയ്യാൻ കഴിയും. പലതും ചെയ്യാൻ കഴിയണം. ജനാധിപത്യത്തിൽ ശരിയായ മതനിരപേക്ഷ രാഷ്ട്രീയത്തിലുള്ള വിശ്വാസം പൊതുസമൂഹത്തിനും യുവതലമുറയിലും വളർത്തിയെടുക്കാൻ ബോധപൂർവം ശ്രമിക്കേണ്ട കാലമാണ് ഇത്. അതൊന്നും അൽപ്പംപോലും പ്രതിപക്ഷം പരിഗണിച്ചില്ല. ഇവിടെയാണ് സോക്രട്ടീസിനെ ഓർമ്മിക്കേണ്ടത്. അദ്ദേഹത്തെ കുറിച്ചുള്ള കുറ്റപത്രത്തിൽ പറയുന്നു.

"ഈ മനുഷ്യൻ ചിന്താപരമായ പ്രകോപനം കൊണ്ട് യുവത്വത്തെ വഴിതെറ്റിച്ചു. അതിനാൽ ഇയാൾ ശിക്ഷ അർഹിക്കുന്നു. സോക്രട്ടീസിന്റെ വാക്കുകൾ "ബുദ്ധിമാന്മാർ ആശയങ്ങളെക്കുറിച്ച് പറഞ്ഞു കൊണ്ടേയിരിക്കും, സാധാരണക്കാർ സംഭവങ്ങളെക്കുറിച്ച് പറഞ്ഞു കൊണ്ടേയിരിക്കും, നിലവാരമില്ലാത്തവർ വ്യക്തികളെക്കുറിച്ച് പറഞ്ഞുകൊണ്ടേയിരിക്കും. ആ പുലഭ്യത്തിൽ അവർക്ക് ആനന്ദ നിർവൃതി ഉണ്ടാകും". ഇതിൽ തങ്ങളേത് പക്ഷത്തിൽപ്പെടുമെന്ന്, ഏത് വിഭാഗത്തിൽപ്പെടുമെന്ന്, ഓരോരുത്തരും സ്വയം ചിന്തിച്ചാൽ മതി. ഇന്ത്യയുടെ ഈ സാഹചര്യത്തെക്കുറിച്ചൊന്നും പ്രതിപക്ഷം ചിന്തിച്ചില്ല. ഓർത്തില്ല.

ഫെസ്റ്റിവൽ, ഓൺ, ഡെമോക്രസി, ദളിത്, ആദിവാസി, ന്യൂനപക്ഷസമ്മേളനം, നാഷണൽ വിദ്യാർഥി പാർലമെന്റ് എന്നിവയെല്ലാം ധൂർത്തും വ്യർഥവും അനാവശ്യവുമാണെന്ന് പ്രതിപക്ഷനേതാവ് പലവുരു പറഞ്ഞു. അതും ആ ചടങ്ങിൽ പങ്കെടുത്ത് പ്രസംഗിച്ചതിനുശേഷം. ശശി തരൂരും അബ്ദുസമദ് സമദാന്നിയും അനിൽകുമാറും സജീന്ദ്രനും കെ എസ് ശബരീനാഥനും ആബിദ് ഹുസൈൻ തങ്ങളും ഒ രാജഗോപാലും വി എം സുധീരനും സി പി ജോണും എല്ലാവരും പലഘട്ടങ്ങളായി അതിൽ പങ്കാളികളായിരുന്നു. അത് സംഘടിപ്പിച്ച സാഹചര്യത്തെക്കുറിച്ച് അൽപ്പമെങ്കിലും പ്രതിപക്ഷം ചിന്തിച്ചില്ല.

ഇന്ത്യൻ ജനാധിപത്യത്തിനും ഭരണഘടനയ്ക്കും മുകളിൽ അധികാര പ്രമത്തതയുടെ പിടിവീണുകൊണ്ടിരിക്കുന്ന ഒരു ദേശീയ സാഹചര്യത്തിലാണ് ഇന്ത്യൻ ക്യാമ്പസുകളിലെ നേതൃത്വം ഇവിടെ ഒത്തുകൂടിയത്. ഇന്ത്യ എന്ന ആശയം ഈ സമയത്ത് വലിയ ചർച്ചാ വിഷയമായിരുന്നു. യുവത്വത്തിന്റെ നിലപാടുകൾ ഏകോപിക്കുന്ന ഒരു അനുഭവമായി ഫെസ്റ്റിവൽ ഓൺ ഡെമോക്രസി മാറി. ഫെസ്റ്റിവൽ ഓൺ ഡെമോക്രസിയുടെ കേരള മോഡൽ രാജ്യം മുഴുവൻ കൊണ്ടുവരണമെന്നും സംസ്ഥാന നിയമസഭകളും ലോക്സഭാ സെക്രട്ടറിയറ്റും അതിന് നേതൃത്വം കൊടുക്കണമെന്നും നിർദേശിച്ചതും രാജസ്ഥാനിലെ സ്പീക്കറും മുൻ കേന്ദ്രമന്ത്രിയുമായിരുന്ന കോൺഗ്രസ് നേതാവ് സി പി ജോഷിയാണ്. അതുപോലെ ഡിജിറ്റൽ അസംബ്ലിയും സഭാ ടിവിയും ലോഞ്ച് നവീകരണവും എല്ലാം അനാവശ്യവും ധൂർത്തും അഴിമതിയുമാണെന്നും പറയുന്നു. ഇതൊന്നും സ്പീക്കറുടെ പോക്കറ്റിൽനിന്ന് തന്നിഷ്ടംപോലെ ചെലവഴിക്കുന്ന കാര്യങ്ങൾ അല്ലെന്നും എല്ലാ ക്രമവും പാലിച്ചുകൊണ്ടാണെന്നും എല്ലാവർക്കും അറിയാം. അതിനുപുറമേ വിവിധ തലത്തിലുള്ള മോണിറ്ററിങ് സമിതികളും രൂപീകരിച്ചാണ് എല്ലാം ചട്ടപ്രകാരം ചെയ്യുന്നത്.

Fig. 5. Sample Input document

The given document will be subjected to text pre-processing first.

Sentences were separated by a process of sentence splitter and removing the stop words like അത്, ഇത്, അവ, അവിടെ etc. After the text will pass through the stemmer and now the text is in pre- processed form so that we can perform the learning operations.

Feature scores were calculated next based on the feature vectors mentioned in section 3.3. The following table illustrates the feature value obtained for the above text document.

Table 2. Feature values obtained for sample document

Paragraph Number	Number of sentences	Feature values of the sentences
1	3	f1-[1.0,0.6666666666666666,0.3333333333333333] f2-[0.9821428571428571,0.9642857142857143,.9464285714285714] f3-[0.0,0.0,0.0] f4-[0.0,0.0,0.0] f5-[0.6024590163934426, 0.7581967213114754, 0.30327868852459017] f6-[0.0,0.0, 0.3333333333333333]

2	**7**	f1-[1.0, 0.8571428571428571, 0.7142857142857143, 0.5714285714285714, 0.4285714285714285, 0.2857142857142857, 0.1428571428571285] f2-[0.9285714285714286, 0.9107142857142857, 0.8928571428571429, 0.875, 0.8571428571428571, 0.8392857142857143, 0.8214285714285714] f3-[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0] f4-[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0] f5-[0.36475409836065575, 0.18442622950819673, 0.36475409836065575, 0.3729508196721312, 0.30327868852459017, 0.4016393442622951, 0.4016393442622951] f6-[0.0, 0.1428571428571285, 0.1428571428571285, 0.1428571428571285, 0.0, 0.1428571428571285, 0.0]
3	**12**	f1-[1.0, 0.9166666666666666, 0.8333333333333334, 0.75, 0.6666666666666666, 0.5833333333333334, 0.5, 0.4166666666666667, 0.3333333333333333, 0.25, 0.16666666666666666, 0.08333333333333333] f2-[0.8035714285714286, 0.7857142857142857, 0.7678571428571429, 0.75, 0.7321428571428571, 0.7142857142857143, 0.6964285714285714, 0.6785714285714286, 0.6607142857142857, 0.6428571428571429, 0.625, 0.6071428571428571] f3-[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.16666666666666666, 0.0, 0.0, 0.0] f4-[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0] f5-[0.13114754098360656, 0.319672131147541, 0.2336065573770492, 0.36885245901639346, 0.3114754098360656, 0.24180327868852458, 0.5409836065573771, 0.4426229508196721, 0.2950819672131475, 0.3114754098360656, 0.29918032786885246, 0.22540983606557377] f6-[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.08333333333333333]
4	**8**	f1-[1.0, 0.8888888888888888, 0.7777777777777778, 0.6666666666666666, 0.5555555555555556, 0.4444444444444444, 0.3333333333333333, 0.2222222222222222, 0.1111111111111111] f2-[0.5892857142857143, 0.5714285714285714, 0.5535714285714286, 0.5357142857142857, 0.5178571428571429, 0.5, 0.48214285714285715, 0.4642857142857143, 0.44642857142857145] f3-[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0] f4-[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0] f5-[0.39344262295081966, 0.3155737704918033, 0.3975409836065574, 0.3360655737704918, 0.0860655737704918, 0.5819672131147541, 0.1885245901639344, 0.16393442622950818, 0.1885245901639344] f6-[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
5	**7**	f1-[1.0, 0.8571428571428571, 0.7142857142857143, 0.5714285714285714, 0.4285714285714285, 0.2857142857142857, 0.1428571428571285] f2-[0.4285714285714285, 0.4107142857142857, 0.3928571428571285, 0.375, 0.35714285714285715, 0.3392857142857143, 0.32142857142857145] f3-[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0] f4-[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0] f5-[0.26229508196721313, 0.12704918032786885, 0.819672131147541, 0.1885245901639344, 0.4057377049180328, 0.26639344262295084, 0.04098360655737046] f6-[0.0, 0.0, 0.1428571428571285, 0.0, 0.1428571428571285, 0.1428571428571285, 0.1428571428571285]
6	**4**	f1-[1.0, 0.75, 0.5, 0.25] f2-[0.30357142857142855, 0.2857142857142857, 0.26785714285714285, 0.25], f3-[0.0, 0.0, 0.0, 0.0] f4-[0.0, 0.0, 0.0, 0.0] f5-[0.6844262295081968, 0.1885245901639344, 0.7295081967213115, 0.3237704918032787] f6-[0.75, 0.25, 0.0, 0.0]
7	**7**	f1-[1.0, 0.8571428571428571, 0.7142857142857143, 0.5714285714285714, 0.4285714285714285, 0.2857142857142857, 0.1428571428571285] f2-[0.23214285714285715, 0.21428571428571427, 0.19642857142857142, 0.17857142857142858, 0.16071428571428573, 0.14285714285714285, 0.125] f3-[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0] f4-[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0] f5-[0.6844262295081968, 0.22131147540983606, 0.3360655737704918, 1.0, 0.45491803278688525, 0.5901639344262295, 0.4057377049180328] f6-[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
8	**7**	f1-[1.0, 0.8571428571428571, 0.7142857142857143, 0.5714285714285714, 0.4285714285714285, 0.2857142857142857, 0.1428571428571285] f2-[0.10714285714285714, 0.08928571428571429, 0.07142857142857142, 0.05357142857142857, 0.03571428571428571, 0.017857142857142856, 0.0] f3-[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0] f4-[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0] f5-[0.0942622950819672, 0.29918032786885246, 0.639344262295082, 0.48360655737704916, 0.4672131147540984, 0.47950819672131145, 0.2581967213114754] f6-[0.14285714285714285, 0.0, 0.0, 0.0, 0.14285714285714285, 0.0, 0.0]

The SVM classifier predicts the class of the sentences and groups the same in four different classes namely VVI (Very-Very Important), VI (Very Important), I (Important), and LI (Least Important). The user is allowed to provide a compression ratio of a particular percentage and the system produces the summary according to the compression factor given. For example, if the compression is given as 25% only 1/4th of the original document will result

in the summary module. The highest preferences are given to those sentences in the SVM class named VVI. The next preference is in the order VI, I, and LI. To train the model, thousands of documents from different genres are selected. We have created our own dataset to train the model since there are no such training datasets in Malayalam. Data are collected from different news portals, travel vlogs, historical vlogs, etc.

Followed by a compression ratio of 30%, the above sample document brings out the text summary as shown below:

വിഷയദാരിദ്ര്യവും വിഭവദാരിദ്ര്യവുംകൊണ്ട് പ്രതിപക്ഷത്തിന് എത്രത്തോളം തരംതാഴാനാകും എന്നതിന്റെ ഉദാഹരണമാണ് അവർ സ്പീക്കർക്കെതിരെ അവതരിപ്പിച്ച പ്രമേയം. നിയമസഭയുടെ അധ്യക്ഷനായി തെരഞ്ഞെടുക്കപ്പെട്ടപ്പോൾ ഞാൻ നെഹ്രോളിയന്റെ വാക്കുകൾ പറയുകയുണ്ടായി. "വെൻ ഐ വാസ് ഇൻ ഈജിപ്ത് ഐ വാസ് എ മുസ്‌ലിം. നൗ ഐ ആം ഇൻ ഇസ്രയേൽ ഡഫ്നിറ്റ്‌ലി ഐ ആം കൃസ്ത്യൻ'' ഈ വാക്കുകൾ പാലിക്കാൻ പരമാവധി ശ്രമിച്ചു. എന്താണ് സഭാധ്യക്ഷൻ ചെയ്യ തെറ്റ്? ലെജിസ്ലേച്ചർ ആക്ടിവിസം എന്ന തലത്തിലേക്ക് നമ്മുടെ നിയമസഭയെ ഉയർത്താൻ ശ്രമിച്ചതോ? മാറുന്ന കാലത്തോടു ചേർന്നുനിൽക്കാൻ പദ്ധതികൾ ആവിഷ്കരിച്ചതോ? ഭരണഘടന വെല്ലുവിളികൾ നേരിടുമ്പോൾ ഭരണഘടനാ ക്ലാസുകളിലൂടെ ബദൽ പ്രതിരോധം ഉയർത്തിക്കൊണ്ടുവന്നതോ? ഇന്ത്യയിൽ ആദ്യമായി നിയമനിർമാണത്തിൽ ജനകീയ പങ്കാളിത്തത്തിന് തുടക്കം കുറിച്ചതോ? ജനാധിപത്യത്തിന്റെ അന്തസ്സത്തയ്ക്ക് ഇടിവു പറ്റുന്ന ഒരു കാലത്തിന്റെ കരിനിഴൽ ഇന്ന് ഇന്ത്യയിലുണ്ട്. ജനാധിപത്യത്തിൽ ശരിയായ മതനിരപേക്ഷ രാഷ്ട്രീയത്തിലുള്ള വിശ്വാസം പൊതുസമൂഹത്തിനും യുവതലമുറയിലും വളർത്തിയെടുക്കാൻ ബോധപൂർവം ശ്രമിക്കേണ്ട കാലമാണ് ഇത്. അതൊന്നും അൽപ്പംപോലും പ്രതിപക്ഷം പരിഗണിച്ചില്ല. "ഈ മനുഷ്യൻ ചിന്താപരമായ പ്രകോപനം കൊണ്ട് യുവത്വത്തെ വഴിതെറ്റിച്ചു. അതിനാൽ ഇയാൾ ശിക്ഷ അർഹിക്കുന്നു. സോക്രട്ടീസിന്റെ വാക്കുകൾ "ബുദ്ധിമാന്മാർ ആശയങ്ങളെക്കുറിച്ച് പറഞ്ഞു കൊണ്ടേയിരിക്കും, സാധാരണക്കാർ സംഭവങ്ങളെക്കുറിച്ച് പറഞ്ഞു കൊണ്ടേയിരിക്കും, നിലവാരമില്ലാത്തവർ വ്യക്തികളെക്കുറിച്ച് പറഞ്ഞുകൊണ്ടേയിരിക്കും. ആ പുലഭ്യത്തിൽ അവർക്ക് ആനന്ദ നിർവൃതി ഉണ്ടാകും". ഫെസ്റ്റിവൽ ഓൺ ഡെമോക്രസി, ദളിത്, ആദിവാസി, ന്യൂനപക്ഷസമ്മേളനം, നാഷണൽ വിദ്യാർഥി പാർലമെന്റ് എന്നിവയെല്ലാം ധൂർത്തും വ്യർഥവും അനാവശ്യവുമാണെന്ന് പ്രതിപക്ഷനേതാവ് പലവുരു പറഞ്ഞു. ഇന്ത്യൻ ജനാധിപത്യത്തിനും ഭരണഘടനയ്ക്കും മുകളിൽ അധികാര പ്രമത്തതയുടെ കിടിവീണുകൊണ്ടിരിക്കുന്ന ഒരു ദേശീയ സാഹചര്യത്തിലാണ് ഇന്ത്യൻ കൂമ്പസുകളിലെ നേതൃത്വം ഇവിടെ ഒത്തുകൂടിയത്. ഇന്ത്യ എന്ന ആശയം ഈ സമയത്ത് വലിയ ചർച്ചാ വിഷയമായിരുന്നു.

Fig. 6. Output document corresponds to **Fig. 5**

5. Experimental Classification Results and Analysis

As mentioned in section 4, Python language is used to do the implementation side. For the training and testing purpose, we have created our own dataset for the Malayalam language which includes documents from different genres like news articles, travel vlogs, historical, geographical documents, etc. For evaluation purposes also, Malayalam documents from different genres are collected. Summarizer is evaluated using the correlation measures like Precision, Recall, and F1-Score. The statistics about the inclusion of ideal sentences in the generated summary can be found out by using these evaluation measures. For this purpose, we have taken the summary generated by our machine learning-based summarizer and also the summary generated by a human being.

Consider,

N (System Summary) = Number of sentences occurring in the final summary generated by the system.

N (Manual Summary) = Number of sentences in the summary generated by a human.

N (System Summary ∩ Manual Summary) =Number of sentences which are common in system generated summary and human generated summary.

Precision can be computed as,

$$\text{Precision (P)} = \frac{\text{N(System Summary} \cap \text{Manual Summary)}}{\text{N(System Summary)}} \quad (1)$$

Recall can be computed as,

$$\text{Recall(R)} = \frac{\text{N(System Summary} \cap \text{Manual Summary)}}{\text{N(Manual Summary)}} \quad (2)$$

F1- score is another evaluation figure we are using to predict the accuracy of our system. Here we are measuring the harmonic mean of precision and recall of our model. The model is considered to be so perfect if we are getting an F1-score value of 1. This can be computed by using the following formula:

$$\text{F1} - \text{Score} = \frac{2.\text{P.R}}{\text{P} + \text{R}} \quad (3)$$

It is possible to adjust the F-score by giving more weightage to precision or recall based on our model. This is called by the name Fβ measure and can be computed from the following formula:

$$\text{F}\beta = \frac{1 + \beta 2\ (\text{Precision} * \text{Recall})}{(\beta 2.\,\text{Precision})\ + \text{Recall}} \quad (4)$$

Here β is the weighting factor which is giving high weightage to precision when β>1 and favours recall when β<1. But, in our model, we are giving equal weightage to precision and recall. So, only the F1- score is having relevance here.

5.1. Comparison Study with Existing Summarizers

The proposed system is compared with the existing offline and online summarizers. **Table 3** deals with the values obtained for those summarizers for the same Malayalam input document we have used for measuring the accuracy of our model. Precision and Recall rates are found to be low for online summarizers like text compactor, auto summarizer, etc. The recall rate of some summarizers is low in comparison with other summarizers which is having a high precision rate.

Table 3. A comparison study with available summarizers

Name of Summarizer	Precision	Recall	F1-Score
Text Summarizer(https://textsummarization.net/text-summarizer)	0.625	0.625	0.625
Summary Generator(https://summarygenerator.com)	0.67	0.25	0.364
Open Text Summarizer(https://www.splitbrain.org/services/ots)	0.5	0.375	0.429

Text Compactor(https://www.textcompactor.com)	0.55	0.375	0.446
Auto Summarizer(https://autosummarizer.com)	0.375	0.375	0.375
E Summarizer(http://esummarizer.com)	0.125	0.333	0.182
Minimum spanning tree-based text summarizer(offline)	0.72	0.63	0.672
Summarizer with SOM clustering(offline)	0.81	0.65	0.721
Proposed Method	**0.93**	**0.84**	**0.886**

From the following graph, it is evident that the proposed system is showing much more precision and recall rate when compared to others.

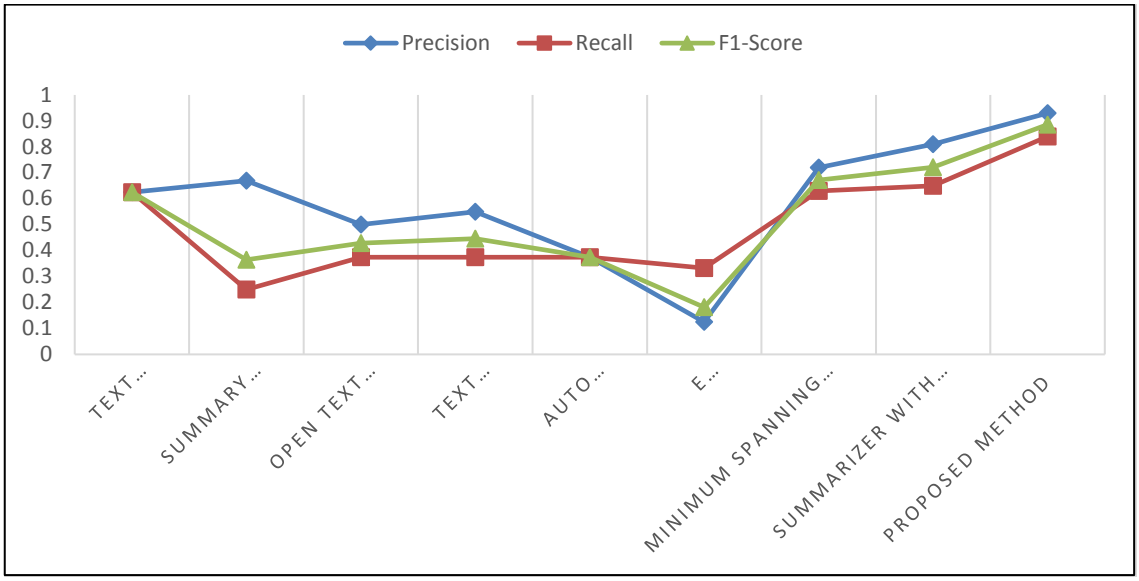

Fig. 7. Relatedness of precision, recall, and F score value

5.2. Result Discussion on Proposed Model

The following statistics will show the results obtained from our model for various datasets. Initially, we are discussing the accuracy of the model with respect to the documents from the sample domain. The following table illustrates the results obtained for sample documents from the history domain.

Table 3. Precision, Recall and F- score obtained for sample documents from history domain

Document	Number of sentences in the original document	Number of sentences the system summary	Number of sentences the human summary	overlapped sentences	Precision	Recall	F1-Score
Document1	137	41	45	38	92.7	84.4	88.4
Document2	250	75	80	69	92.0	86.3	89.0
Document3	380	114	110	102	89.5	92.7	91.1
Document4	20	6	7	6	100.0	85.7	92.3
Document5	264	79	83	70	88.6	84.3	86.4
Document6	43	12	15	11	91.7	73.3	81.5
Document7	32	10	13	10	100.0	76.9	87.0
Document8	110	33	35	30	90.9	85.7	88.2
Document9	33	30	30	28	93.3	93.3	93.3
Document10	26	8	10	8	100.0	80.0	88.9

From **Table 3**, it is seen that the average precision measure we are getting is 93.87.

The following graph glimpses the relatedness between different documents in terms of precision, recall, F1-score, number of sentences in original document, system summary, human summary and number of overlapped sentences.

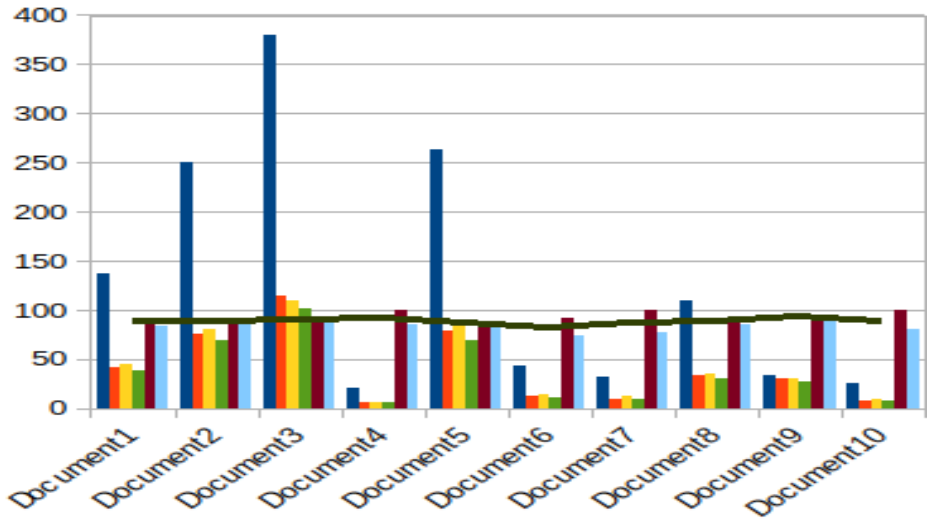

Fig. 8. Relatedness of precision, recall, and F score values.

The ideal summaries are generated by human evaluation. Test documents have been given to Malayalam language experts and they have done the summary manually. System-generated summaries are compared with the manual summary and thereby the measures, precision, recall, and F1-score have been calculated. But when humans are doing the manual summarization, based on the experts some variations may occur. Expert2 may not choose the same sentences as selected by Expert1. To remedied out this problem, we are using a grading scheme for the

sentences in the document. According to the relevance of the sentences, we are assigning a grading factor or confidence value from 5 to 1. This measure is known by the name Relative Utility (RU).

Each document to be manually summarized will be assigned the confidence value as follows: {(S1,5), (S2,3), (S3,4), (S4,3), (S5,1) …….. (Sn,4)}

Here, S1, S2…Sn represents the sentences in the document in the given order. For Example, A document contains 10 sentences and given the grading values like,
{(S1,5), (S2,4), (S3,4), (S4,2), (S5,1), (S6,1), (S7,2), (S8,3), (S9,4), (S10,5)}

Expert 1 have selected the sentences S1, S3, S5, S8 and S10 and Expert 2 have selected the sentences S1, S3, S6, S9 and S10.

Utility point for Expert1 and Expert2 can be measured as,

Utility point of Expert1 = 5+4+1+3+5=18

Utility point of Expert2 = 5+4+1+4+5=19

Here we can see that the utility point is relatively the same even if the experts have selected different sentences. The results have shown that the variation occurring in results by different experts is very less by considering the Relative Utility measure. The statistics obtained from five different experts for the below document with the labeling of RU are shown in **Table 3**.

പഞ്ചാബ് സംസ്ഥാനത്തെ ലുധിയാന ജില്ലയിലെ ഒരു വില്ലേജാണ് ഭാഗ്ഗൂർ. ഭാഗ്ഗൂർ വില്ലേജിന്റെ പരമാധികാരി സർപഞ്ചാണ്. ജനങ്ങൾ തെരഞ്ഞെടുക്കുന്ന പ്രതിനിധിയാണ് സർപഞ്ച്.

2011 ലെ ഇന്ത്യൻ കാനേഷുമാരി വിവരമനുസരിച്ച് ഭാഗ്ഗൂരിൽ 235 വീടുകൾ ഉണ്ട്. ആകെ ജനസംഖ്യ 1268 ആണ്. ഇതിൽ 659 പുരുഷന്മാരും 609 സ്ത്രീകളും ഉൾപ്പെടുന്നു. ഭാഗ്ഗൂരിലെ സാക്ഷരതാ നിരക്ക് 76.97 ശതമാനമാണ്. ഇത് സംസ്ഥാന ശരാശരിയായ 75.84 ലും താഴെയാണ്. ഭാഗ്ഗൂരിലെ 6 വയസ്സിനു താഴെയുള്ള കുട്ടികളുടെ എണ്ണം 121 ആണ്. ഇത് ഭാഗ്ഗൂരിലെ ആകെ ജനസംഖ്യയുടെ 9.54 ശതമാനമാണ്.

2011 ലെ ജനസംഖ്യാ കണക്കെടുപ്പ് രേഖകൾ പ്രകാരം 415 ആളുകൾ വിവിധ തൊഴിലുകളിൽ ഏർപ്പെട്ടിരിക്കുന്നു. ഇതിൽ 373 പുരുഷന്മാരും 42 സ്ത്രീകളും ഉണ്ട്. 2011 ലെ കാനേഷുമാരി പ്രകാരം 89.4 ശതമാനം ആളുകൾ അവരുടെ ജോലി പ്രധാന വരുമാനമാർഗ്ഗമായി കണക്കാക്കുന്നു. എന്നാൽ 32.77 ശതമാനം പേർ അവരുടെ ഇപ്പോഴത്തെ ജോലി അടുത്ത 6 മാസത്തേക്കുള്ള താത്കാലിക വരുമാനമായി കാണുന്നു.

ഭാഗ്ഗൂരിലെ 446 പേരും പട്ടികജാതി വിഭാഗത്തിൽ പെടുന്നു.

Fig. 9. Sample document taken for the calculation of Relative Utility

The language expert labeled the utility measure for the sentences in the document as: {(S1,5), (S2,5), (S3,1), (S4,4), (S5, 3), (S6, 2), (S7,3), (S8,2), (S9,2), (S10,2), (S11, 4), (S12, 2), (S13,3), (S14, 2), (S15, 5)} All the five users are requested to select 8 relevant sentences from this document. Selected sentences by each user and the relative utility measure are depicted in **Table 4**. **Fig. 10** reflects the differences in utility measures by different human evaluators operated on the same document.

Table 4. Relative utility (RU)score of different evaluators

Sentence	Human	Human	Human	Human	Human
S1	5	5	5	5	5
S2	5	5	5		
S3				1	1
S4	4	4		4	4
S5	3	3	3		3
S6			2	2	
S7		3		3	
S8	2		2		
S9	2				2
S10				2	2
S11		4	4	4	
S12	2				
S13					3
S14		2	2	2	
S15	5	5	5		5
RU	**28**	**31**	**28**	**23**	**25**

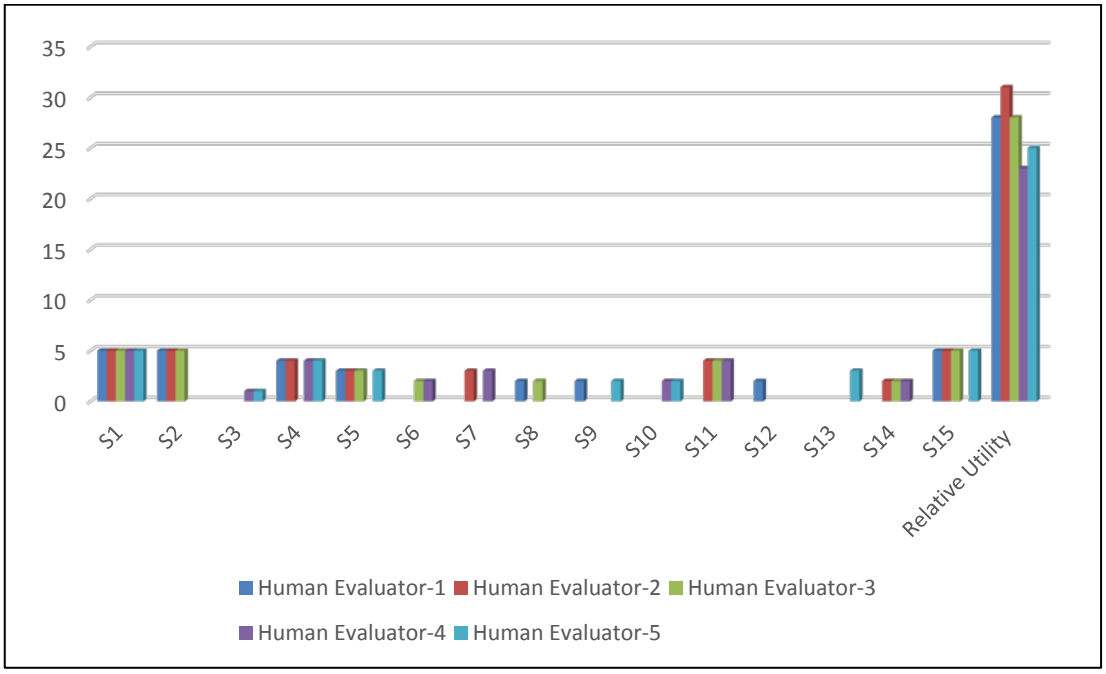

Fig. 10. Comparison chart of RU by 5 different evaluators.

Fig. 11 below shows the variation in accuracy rate as the documents are selected from different domains.

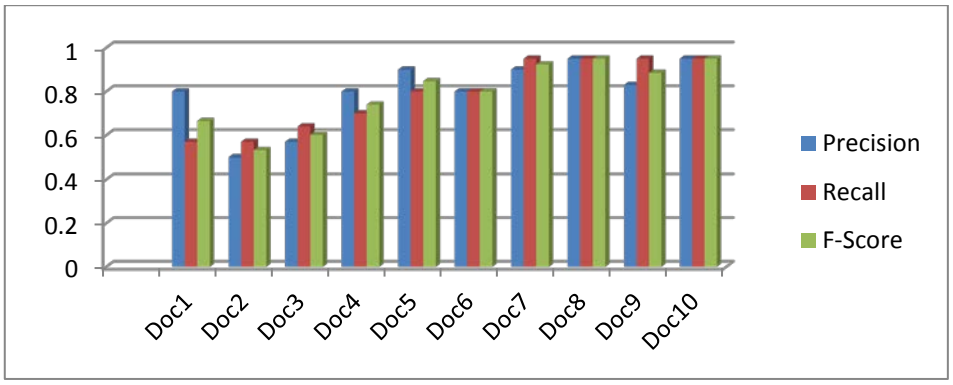

Fig. 11. A comparison chart of Performance metrics with ten documents

The complexity of the proposed methodology can be calculated as follows:

Pre-processing step in the algorithm consists of iterations where the pre-processing tasks have to be performed for each sentence in the document. If there are n sentences in the document the complexity of this iteration can be taken as O(n). In the sentence ranking module also based on the features selected, we can determine the complexity as O(f*n) where f is the number of features selected and n is the number of sentences. We can say that the performance of the algorithm is in direct proportion with the size of the input document taken.

6. Conclusion and Future Work

An SVM-based Malayalam text summarizer is proposed in this work. Compared to other languages, NLP works including summarization are low in Malayalam due to the scarcity of resources and agglutinative nature of the language. An extractive summarizer is created hereby training the Malayalam documents based on the SVM classifier. An average accuracy rate of 93.87 is achieved as a result. The system is tested against documents from different genres and also with different human evaluators. An evaluation measure of relative utility is also incorporated here to assess the accuracy of the summarized document when different human beings are assessing the accuracy rate against the system summary. Sentence wise pre-processing is required here to extract the features. This may lead to higher execution time for longer documents, but it can be compromised by the use of high-speed computers. The precision rate for the documents which contains more guillemets within a paragraph is found to be comparatively lesser since the compression factor may reach in such cases without covering all the paragraphs in the document concerned. But this scenario may come less often. The system can be extended with deep learning training models so that the precision and recall rate can be enhanced to fit the model more accurately. From this extracted summary, an abstractive summary can also be generated by incorporating a natural language generator that will be able to regenerate the short sequence of a sentence from this extracted summary.

Conflict of Interest

The authors declare no conflict of interest.

References

[1] Neto, Joel & Freitas, Alex & Kaestner, Celso, "Automatic Text Summarization Using a Machine Learning Approach," in *Proc. of SBIA 2002: Advances in Artificial Intelligence*, 205-215, 2002.Article(CrossRef Link).

[2] Nikita Desai1, Prachi Shah, "Automatic Text Summarization Using Supervised Machine Learning Technique for Hindi Language," *International Journal of Research in Engineering and Technology* vol. 05, no. 06, pp. 361- 367, Jun-2016. Article(CrossRef Link)

[3] Shah, Chintan & Jivani, Anjali, "An Automatic Text Summarization on Naive Bayes Classifier Using Latent Semantic Analysis," *Data, Engineering and Applications*, pp. 171–180, 2019. Article(CrossRef Link)

[4] Ramanujam, Nedunchelian & Manivannan, Kaliappan, "An Automatic Multi document Text Summarization Approach Based on Naïve Bayesian Classifier Using Timestamp Strategy," *The Scientific World Journal*, 1-10, 2016. Article(CrossRef Link).

[5] BalaAnand Muthu, Sivaparthipan CB, Priyan Malarvizhi Kumar, Seifedine Nimer Kadry, Ching-Hsien Hsu, Oscar Sanjuan, Ruben Gonzalez Crespo, "A Framework for Extractive Text Summarization Based on Deep Learning Modified Neural Network Classifier," *ACM Trans. Asian Low-Resource. Lang. Inf. Process.*, 20(3), 2021, Article No. 45. Article(CrossRef Link)

[6] Rahul, Surabhi Adhikari, Monika, "NLP based Machine Learning Approaches for Text Summarization," in *Proc. of the Fourth International Conference on Computing Methodologies and Communication*, IEEE, 2020. Article(CrossRef Link)

[7] Amita Arora, Akanksha Diwedy, Manjeet Singh and Naresh Chauhan, "Machine Learning Approach for Text Summarization," *International Journal of Database Theory and Application*, Vol.10, No.8, pp.83-90, 2017. Article(CrossRef Link)

[8] Krishnaprasad P, Sooryanarayanan A and Ajeesh Ramanujan, "Malayalam Text Summarization: An Extractive Approach," in *Proc. of IEEE International Conference on Next Generation Intelligent Systems (ICNGIS)*, 2016. Article(CrossRef Link)

[9] Kanitha D K et al, "Malayalam Text Summarization Using Graph Based Method," *International Journal of Computer Science and Information Technologies*, Vol. 9(2), pp. 40-44, 2018. Article(CrossRef Link)

[10] Kanitha D K, D. Muhammad Noorul Mubarak, S. A. Shanavas, "Malayalam Text summarization Using Vector Space Model," *International Journal of Engineering and Techniques (IJET)*, Vol. 4, No. 2, pp. 918-925, 2018. Article(CrossRef Link)

[11] Ajmal E B, Rosna P Haroon, "An extractive Malayalam document summarization based on graph theoretic approach," in *Proc. of 2015 Fifth International Conference on e-Learning (econf)*, IEEE, Manama, Bahrain, 2015. Article(CrossRef Link)

[12] Rahul Raj M and Haroon R P, "Malayalam text summarization: minimum spanning tree-based graph reduction approach," in *Proc. of IEEE 2nd International Conference on Advances in Computing, Communication, & Automation (ICACCA)*,2016. Article(CrossRef Link)

[13] RAHUL RAJ, M., HAROON, R.P. & SOBHANA, N.V., "A novel extractive text summarization system with self-organizing map clustering and entity recognition," *Sādhanā Publications*, Springer, vol. 45, 2020. Article(CrossRef Link)

[14] Steinberger, Josef & Jezek, Karel, "Evaluation Measures for Text Summarization," *Computing and Informatics*, vol. 28, no. 2, pp. 251-275, 2012. Article(CrossRef Link)

[15] Indu M, Kavitha K V, "Review on text summarization evaluation methods," in *Proc. of International Conference on Research Advances in Integrated Navigation Systems (RAINS – 2016)*, IEEE, 2016. Article(CrossRef Link)

[16] Tsutomu HIRAO, Hideki ISOZAKI, Eisaku MAEDA, Yuji MATSUMOTO, "Extracting Important Sentences with Support Vector Machines," in *Proc. of the 19th international conference on Computational linguistics - Volume 1*, pp. 1-7, 2002. Article(CrossRef Link)

[17] Vo Duy Thanh, Danang, Vo Trung Hung, Ho Khac Hung, Tran Quoc Huy, "Text Classification Based on SVM and Text Summarization," *International Journal of Engineering Research & Technology (IJERT)*, Vol. 4, No. 02, February-2015. Article(CrossRef Link)

[18] Jaewoong Moon, Subin Kim, Jaeseung Song, and Kyungshin Kim, "Study on Machine Learning Techniques for Malware Classification and Detection," *KSII Transactions on Internet and Information Systems*, Vol. 15, No. 12, pp. 4308-4325, Dec. 2021. Article(CrossRef Link)
[19] Mengli Zhang, Gang Zhou, Wanting Yu, Wenfen Liu, "KI-HABS: Key Information Guided Hierarchical Abstractive Summarization," *KSII Transactions on Internet and Information Systems*, Vol. 15, No. 12, pp. 4275-4291, Dec. 2021. Article(CrossRef Link)
[20] https://techcommunity.microsoft.com/t5/ai-customer-engineering-team/bootstrap-your-text-summarization-solution-with-the-latest/ba-p/1268809
[21] Manju, K., David Peter, S., Mary Idicula, S., "A Framework for Generating Extractive Summary from Multiple Malayalam Documents," *Information*, vol. 12, issue 1, 2021. Article(CrossRef Link)
[22] Sindya K. Nambair, David Peter S., Sumam Mary Idicula, "Attention based abstractive summarization of a Malayalam Document," *Procedia Computer Science*, vol. 189, pp. 250-257, 2021. Article(CrossRef Link)
[23] Diallo, B., Hu, J., Li, T., Khan, G. A., Hussein, A. S., "Multi-view document clustering based on geometrical similarity measurement," *International Journal of Machine Learning and Cybernetics*, 2021. Article(CrossRef Link)
[24] Khan, G.A., Hu, J., Li, T. et al., "Multi-view low rank sparse representation method for three-way clustering," *International journal of Machine Learning and Cybernetics*, vol. 13, pp. 233–253, 2022. Article(CrossRef Link).
[25] Khan, G.A., Hu, J., Li, T. et al., "Multi-view data clustering via non-negative matrix factorization with manifold regularization," *International journal of Machine Learning and Cybernetics*, vol. 13, pp. 677–689, 2022. Article(CrossRef Link)

Rosna P Haroon has done her MTech in Computer and Information Science from Cochin University of Science and Technology, Kerala, India. She is currently working as assistant professor in Ilahia College of Engineering and Technology, Kerala, India. She is pursuing PhD in APJ Abdul Kalam Technological University, Thiruvananthapuram, Kerala as part time research scholar. Her research interests are Natural Language Processing, Artificial Intelligence and Machine learning. She has published more than 25 papers in international conferences and Journals.

M Abdul Gafur received Master Degree in Computer Science and Engineering from National Institute of Technology Calicut, India in 2007 and PhD Degree in Computer Science and Engineering from Jawaharlal Nehru Technological University Hyderabad in 2015. He is working as Professor in Computer Science and Engineering at Ilahia College of Engineering and Technology Muvattupuzha. From 2017 to 2020 he was working as Head of the Department of Computer Science and Engineering and Dean (Research) at MEA Engineering College. From 2008 to 2017 he worked as a faculty member at University of Tabuk, Kingdom of Saudi Arabia. His research interest includes wireless protocol design, Network Security, Artificial Intelligence and Data Science. Dr Gafur's awards and Honor include Fellow of Institution of Engineers (India) and Senor member of IEEE.

Dr. Barakkath Nisha Usman is a senior member in IEEE, she had contributed technically in various events conducted by IEEE. She has completed her Ph.D. in the area of Wireless sensor Networks from Anna University, Chennai. She had received her Bachelor of Engineering in computer science engineering with distinction from Anna University, Chennai. She got her Master of Engineering in Computer Science Engineering from Anna University, Chennai. She received Gold Medal for securing First Rank in Her post-graduation. She was working as Associate Professor in well-established engineering colleges for the past 15 years. Her area of Interests includes Adhoc networks, Wireless sensor Network, Data mining, Natural Language Processing. She has published various papers in reputed SCI, SCOPUS indexed Journals and she has presented papers in various national and International Conferences. She has filled two patents in the field of IOT and Artificial Intelligence. She is an active member of various professional societies. she reviewed more than 50 papers as a reviewer for international journal and Conferences.