# A Hybrid Document Features Extraction with Clustering based Classification Framework on Large Document Sets

S Anjali Devi[1], S Siva Kumar[2]
Department of Computer Science and Engineering
Koneru Lakshmaiah Education Foundation, Vaddeswaram
Guntur, Andhra Pradesh, India -522502

***Abstract*—As the size of the document collections are increasing day-by-day, finding an essential document clusters for classification problem is one of the major problem due to high inter and intra document variations. Also, most of the conventional classification models such as SVM, neural network and Bayesian models have high true negative rate and error rate for document classification process. In order to improve the computational efficacy of the traditional document classification models, a hybrid feature extraction-based document cluster approach and classification approaches are developed on the large document sets. In the proposed work, a hybrid glove feature selection model is proposed to improve the contextual similarity of the keywords in the large document corpus. In this work, a hybrid document clustering similarity index is optimized to find the essential key document clusters based on the contextual keywords. Finally, a hybrid document classification model is used to classify the clustered documents on large corpus. Experimental results are conducted on different datasets, it is noted that the proposed document clustering-based classification model has high true positive rate, accuracy and low error rate than the conventional models.**

***Keywords*—Classification; document feature extraction; document similarity**

## I. Introduction

Machine learning algorithm currently finds wide-spread use in the principles of data mining, where text classification is also a part of it. Work is currently being conducted on the use of machine learning methods to improve efficiency and raising the complexity of computations. Literature in [1] is reviewed about the approaches to machine learning in text classification. The benefit of the suggested solution was that it factored both local and global characteristics, and had the potential to be noise resistant. The suggested solution was shown to work better than traditional SVM methodology, using comparative studies on different datasets. Document representation in vector model is also an essential part in the document clustering algorithms. Several text representations models such as a n-gram models, bag-of-words and feature filtering, etc., have been widely used for large collection of documents. Generally distributed documents share some common context for clustering and categorization [2]. These contexts are represented using documents key terms. Textual data are represented using words and phrases as features in a high dimensional vector space.

Document clustering is the collection of a large number of documents into a set of useful cluster sets where each cluster represents a specific topic or context. The documents within the group should have a high degree of similarity while the degree of similarity among distinct document clusters should be reduced. Traditional clustering methods used to cluster the documents without paying much attention to the contextual information of the document set. For example, if two or more documents are representing same topic using different terminology which are semantically same, the documents are bagged under different clusters. This kind of clustering may lead to inefficient information retrieval. So, document clustering has become an increasingly important for improving the documents sharing and communication in distributed environment. Document clustering has many applications in the area of information retrieval and data mining [3]. For this purpose, different document clustering techniques emerged to better perform the clustering so as to overcome the limitations that are there in the traditional ones.

Clustering methods can be categorized into two main categories. They are Generative approaches (model based) and Discriminative approaches (similarity based). Model based techniques are used to develop extended models from the peer document sets, with each model randomly assigning one particular document cluster. In the similarity based approach [4], function involving pairwise document similarities can be optimized and aiming to optimize the average cluster similarities within the peer overlay clusters.

Moreover, the different levels of analysis are not disjunct. For instance, semantics plays an important role in the syntactic analysis. NLP is a subfield of artificial intelligence and linguistics. In IR, NLP is often used as a pre-processing step. When a system wants to find the most important information in text and then wants to retrieve the information found, it first has to define the most important parts. It is important to note that text mining, IR and NLP are different fields. Sophisticated NLP techniques are frequently used in IR to represent the content of text in an exact way (e.g. noun and verb phrases being the most important ones), extracting the main points of interest, depending on the domain of the IR service. However, NLP is not only used in parsing the documents, but also for handling the user queries. The important information has to be parsed from the user queries in a similar way [5]. NLP techniques are used in almost every aspect of the text mining process, namely in Named Entity Recognition.

The remainder of the paper is described as follows. Section 2 presents detailed information about related work and advances in the field. The proposed hybrid clustering based classification is presented in Section 3. Experimental results are elaborated in Section 4. At the end, conclusion of the paper is detailed in Section 5.

## II. Related Works

Two model architectures are available for representing the text in its vector form. The first predicts the current word from the related terms in the immediate surrounding area while ignoring the order of terms and the second predicts with the aid of the current word the surrounding background words. CBOW is quicker to train when skipping – gram is slower, but better in terms of word weight for Word2Vec. Vectors in Google's Word2Vec are trained from Google News documents on 100 billion words, and are open to all publicly. Such vectors have 300 dimensions and are trained using a continuous bag – of – word model, meaning terms. WordNet is a broad database of English-speaking lexical terms [6]. Here the terms connected with each other are grouped into a collection known as Synset. Lexical categories such as nouns, verbs, adjectives etc. form different synsets and are related through conceptual-semantic relationships and lexical relationships. It is a kind of dictionary and thesaurus which can provide meanings to the terms and comparisons with other terms. Words are thus like nodes, and the connections reflect the relations between them. Word types found in various synsets are of different meanings. WordNet's new online edition is 3.1. Similar to WordNet, ConceptNet is also a broad semantic network composed of the principles relevant to our everyday lives. The ideas apply to the principles of commonsense, and this information is derived from the experiences of average people over the Internet. It is the largest shared information base accessible to the public, consisting of over 2,50,000 relationships. These approaches are generally implemented using domain independent approach in order to result better optimization solutions. Different evolutionary approaches such as genetic algorithms, Rough-set, SVM, etc. are used to classify the document sets from large corpus. Genetic algorithms are implemented to find the complex patterns and classification rules on huge datasets [7]. In some of the hybrid approaches, genetic algorithms are integrated with decision tree schemes to generate an optimized decision tree. Classification models such as Naïve bayes with ensemble decision tree models namely CART, C4.5, Bayesian tree and random forest are used to classify document and feature extraction. They concluded that no single traditional model existed to handle uncertainty for document prediction with large number of attributes set. Hadoop is a software framework used for efficient scalable and parallel programming applications in java and is responsible for processing huge amount of data. It operates on distributed environment with specific clusters, provides results with fault tolerant. It can integrate multiple cluster node's computation and storage data in an efficient manner. Traditional document clustering algorithms are compared in distributed biomedical repositories for efficient document feature extraction [8]. An advanced three-layer biomedical framework has been implemented to cluster the set of documents [9]. This framework is based on a multi-layer neural structure of neighborhood peers. Many overlay peers which act as the representative object of its lower neighborhoods are clustered to form higher level clusters. The basic limitation of this model is selecting an optimal threshold for a dynamic size overlay network. Also, it is very hard to balance the structure size and peer documents.

A model using a parallel approach is implemented to cluster the multiple document collections [10]. The key issue is to find automatic document clusters in large text corpus and it is very high cost to compare documents in a high dimensional vector space. This algorithm tries to minimize the distance computations and cluster size in the training dataset documents, called pivots. They used parallel algorithm in an efficient way to optimize a complex data structure which affords efficient indexing, searching and sorting. Traditional probability estimation techniques such as Naïve bayes, markov model, Bayesian model [11] are used to find the highest probability estimation variance among the gene and its related disease sets in biomedical document sets. Classification is the process of finding and extracting the main contextual meaning of the gene or disease patterns from the distributed document sources and it has become an integral part of day to day activities in all domains like cloud, forums, social networking and medical repositories. Automatic text Classification fulfills certain goals by implementing Classification techniques at the user end to find relevant summaries of the large document sets. Document summaries represent sentences or phrases extracted from different sources without any subjective human intervention or editorial touch and thus making the end product completely unbiased. Classification is a highly interdisciplinary field involving areas like information extraction, text mining, and information retrieval, natural language processing and medical databases. Currently, many scholars at home and abroad have studied the technology of text classification using key methods like conventional machine learning and the deep learning that is currently common. They define the "Clustering of full-subtopic retrieval with keyyphrase-based search results," in that Consider the problem of multiple documents related to the individual subtopics of a Web query, called "complete child retrieval". They present a new algorithm for grouping search results to solve this problem which generates clusters labeled with key phrases [12]. The key phrases are extracted from the search results generic suffix tree and combine into a grouping enhanced by a hierarchical agglomeration process. They also presented a new method to assess the success of complete recovery subthemes, namely "look for secondary duration arguments under adequate documentation". They used a test set explicitly designed to assess the recovery of the subthemes, they found that our algorithm passed all other clustering algorithms of existing research results as a method of redirecting search results underlines the diversity of the results (at least for k>1), that is to say when they are interested in recovering more than one related sub-theme document).

Kostkina et al. [13], they suggested a new approach which expanded the features of short text based Wikipedia and Word2vec. The first phase was the creation of Wikipedia's semantine related definition sets. The semantic relationship between the goal and related concepts was measured; the authors received articles which were highly applicable to the Wikipedia concepts. The author then expanded the applicable notion sets to short texts, and it was noted that this methodology could achieve greater semantine relatedness compared to traditional similarity calculation principles using statistical approach. Experimentally it was shown that the precision of classification could be invented by extending the features of short texts.

Mishra et al. [14], a new system of the Word Embedding Function Extension for Short Text (WEFEST) that extended short texts using word embedding for classification is presented. The proposed WEFEST was embedded in a deep-language model in which word corrections were used to learn a new embedding space. Thanks to the phase the new function vectors space is picked. The use of pre-trained word function embedded in each short text in the training dataset has been enhanced, the authors made use of the nearest neighboring algorithm to achieve short text classification, and the effectiveness of the suggested technique has been validated by the empirical results on Chinese news websites containing title datasets for text classification. They applied the various function extraction methods, feature representation methodology, and text classification approaches. The proposed work was focused on forensic autopsy knowledge to find suitable methods for extraction of features, meaning of features and categorization of texts. From the empirical findings it has been discovered that the unigram features outperformed bigram, trigram, and unigram, bigram, and trigram variants. Compared with normalized TF-IDF structures, the TF and TF-IDF value representation approach works efficiently. LDA was used to extract the thematic details. The authors could add features that were relevant to the subject to the document defined by feature set to enhance the classification of the text. The authors [15] explored various forms of terms frequency and topic-related data, and these were considered traits for supporting vector machine. The experimental results on three companies showed that the accuracy of text classification could be improved by combined features. Unlike the supervised selection technique, which includes category information in the training data, Park et al. [16] proposed an unsupervised feature selection technique in which no information based on categories was needed. This helped the framework to include more framework scenarios, since labeled data was both expensive and not very reliable. Like the other unsupervised methods, this technique made use of embedding terms to identify terms that had virtually the same semantic meaning. The word embedding maps the terms into vectors, preserving the semantine relationships between terms. Many of the words were not used as features to prevent redundancy; the writers chose the most suitable word with similar semantic meaning. Sinoara et al. [17] proposed feature selection technique that was based on Kullback-Leibler (KL) divergence. The purpose of this technique is to evaluate the current association between each class and subclass through KL divergence. The Mutual information method was used for calculating the correlation between each feature and subclass; Term frequency probability was used for measuring the importance of subclass characteristics, so that, for parent class node, a superior discrimination set of features could be selected. The authors used hierarchical feature selection techniques and SVM classifiers on two organizations for purposes of hierarchical text classification tasks. Experiments showed that the proposed algorithm was successful compared to the cchi square statistics (CHI), information gain (IG), and shared knowledge (MI) directly used to pick hierarchical features.

Jiang et al. [18] proposed a novel text classification algorithm, based on the Ant Colony Optimization (ACO). It abused the discreteness of the features of the text document and the value the ACO provides in addressing discrete issues. The behavior of the ant population having the class information was used for classifying the text in order to find a suitable route matching during the process of iterating the algorithm. A score of connectedness between two concepts was high if there were several paths between them (which consisted of direct / indirect hyperlinks). Now TD and WD were concatenated vertically to form the TD&WD matrix, which was used for classification purposes. Of the grouping, they used majority voting methodology. At Reuters (0.9331), 20 Newsgroup (0.7563), and RCV1 (0.5198), their scheme registered appropriate classification accuracies.

Song et al. [19] used NPE for selection of features and applied the PSO classifier for classification of documents. NPE is a better feature-selection scheme than Latent Semantic Indexing, they stated. LSI has proven to be a powerful tool for various information retrieval tasks but it may not be a successful discriminating feature selector for classifying documents into different categories. Feature extraction is a core concept in the text classification process. To build lexical chains, Ravindran tap semantine tools such as synonyms and identity [20]. Based on lexical chains, a two-pass algorithm generates feature vectors, first generating all possible lexical chains and then selecting the longest chains. Through removing unimportant strings, they achieve a reduction of 30 per cent in function vector dimensions and an increase of 74 per cent in execution time. Apart from English, it is also used in the classification of sentiments in the Chinese language text. Likewise, it was used for recurrent neural networks and the findings outperform the standard techniques in both cases. A further attempt was made to identify document using word2vec in combination with the LDA method [21], which also provided better results. In addition to text classification, word2vec has been used in many other areas of application such as improving medical knowledge through unsupervised medical corporate learning [22], answer selected from possible collection, good, poor in a question – response method [23], etc. Word2Vec is an unsupervised model of writing, writing the semantic context associated with the text. They developed a framework by using Information Retrieval (IR) strategies to extract information in biomedical domain [24]. According to the relevance degree, their framework can rank the documents. Their framework can extract relevant documents as well as can diversify the information. The authors presented two labelling methods and merged some IR models. They validated their theory by experimenting on TREC Genomics datasets and result enhanced performance.

Ma et al. [25] presented an algorithm for biomedical documents classification by using Medical Subject Headings (MeSH) and MEDLINE indexing. The author considered 50 articles from MEDLINE and classified these documents by the above said algorithm which uses Natural Language Processing scheme. After calculating precision and recall manually for individual documents, he calculated average precision and recall. The author also identified three major flaws for this approach. Those are: 1) Precision and recall are decreased significantly because of the exact matching. 2) The given algorithm is unable to classify an abstract of biomedical documents. 3) Because blogs terminologies are not MeSH headings, the algorithm can't be applied to the articles of blogs. To overcome these flaws, future research and work is necessary for this method. Park et al. [26] proposed an algorithm that results the top search results for IR queries. For their approach Boolean interface is used without ranking functions. Through conjunction of queries ranked documents are resulted using relevance metric. By the efficient use of probabilistic modelling, the researchers formed their algorithm. The above algorithm sets off a minimum cut-off for the documents to be categorized under high ranking. They argued that, their technique supports monotonic ranking of various keywords and the respective interface uses Boolean expression of keywords. The authors validated their methodology by experimenting on PubMed database and TREC dataset and got enhanced results.

Rashid et al. [27], introduced two new deep learning approaches. They calculated the embedded words and analyzed that with other modelling approaches. As compared to other deep learning approaches of biomedical documents mining, the above said algorithm results better performance. They categorized their research into three sub-categories: 1) Various domain-specific representation are analyzed and a new word embedding approach is proposed. 2) DBN-base DDIE model and RNN-based NER model are introduced through which the process of word embedding is done, and it is compared with skip-gram, CBOW, GolVe, etc. 3) This technique shows significant results in word embedding with better recall. Extraction of keywords from text data is an important technique used by search engines and indexing services to quickly categorize and locate relevant data based on keywords explicitly or implicitly provided. In this section, the literature review involves the various methods used to locate and identify keywords in the individual papers, social networking sites, lecture audio archives, speech transcripts, website database etc. It is important to note that most of the algorithms to be considered in the analysis used an external corpus of documents to check and assess the algorithms' performance. Similarly, these algorithms relied on a weighted function that combined some measure of the presence of a word or phrase within a text with a similar measure from the body. The most common measurements used were word frequency, word distance, document position of terms, co-occurrence with other terms, word-to-word relationship (lexical chains), key phrases, etc. They suggested a system for extracting keywords that would work on individual documents. They followed a text-oriented approach in which, irrespective of the current state of a corpus, the same keywords are extracted from a text The DIKpE algorithm was evaluated on a publicly accessible keyyphrase extraction dataset containing 215 full-length documents from various computer science subjects for its effectiveness and performance. DIKpE was evaluated by measuring the number of matches automatically extracted between the key phrases attached to the text and the keyphrases. DIKpE was found to have clearly outperformed the other two algorithms in extracting the keyphrases, although no training activity was undertaken. They discussed many techniques for automated (unsupervised) keyword extraction for voice transcripts. He found the multiparty meeting domain in particular, and explored the suitability of certain algorithms that were successfully used in meeting transcripts for automated keyword extraction of written text. To test these keyword extraction algorithms, they used transcripts from the ICSI meeting corpus. They also integrated POS filtering, word clustering, sentence salience score into the TF-IDF system and evaluated the outcomes. The accuracy and efficiency of the thematic classification were determined. Two unsupervised discriminative terms were used to automatically classify transcriptions which were extremely incomplete. Term Frequency – Inverse Document Frequency (TF-IDF) using the Gini Purity criteria approach was used to identify the transcription themes. They discovered that the Wikipedia page redirects to automatically gain language-independent variations in morphological character. Four languages have been used for research, namely 3.83,000 Arabic documents, 50 million English documents, 50,000 Hungarian documents and 2.11,000 Portuguese documents. For performance measurement, standardized discounted cumulative benefit and mean average precision were used. The authors in [29] performed Arabic stemming responsive material and substantial progress in English retrieval, outperforming words and stems. Classified news articles using the KNN approach to machine learning. Naïve Bayes term graph model, K-nearest Neighbours (KNN), was used by the authors as a hybrid approach to obtain accurate results. The authors used Reuters dataset with 21578 articles, in which 9603 were training papers and 3299 were test papers. Specific pre-processing methods were used to achieve better results for the documents. A model program for the Vector space was developed and relevant documents were collected for the query. The authors clarified the methods used in text classification such as, K-Nearest Neighbors, Regression Models, Decision Trees, Decision Rules, Naïve Bayes and Bayesian Networks. Significant division of news articles based on classifying documents into various categories, the relevant document was displayed when entering keywords. For each document Association Rule Mining algorithm was applied to find the frequently co-occurring terms and then mapped to a weighted and guided graph. Unsupervised approaches typically include assigning each candidate's sentence a saliency score by considering various features. Supervised machine learning algorithms have been proposed to identify a candidate's phrase either into a main phrase or not using features such as occurrence frequency, POS details and position of the term in the text. Both of the above methods make use of the document text only to produce key phrases and cannot (as-is) be used to produce label-specific key phrases.

The keyword extraction model was developed using both statistical as well as pattern features inside words. The algorithm is independent of language and does not require a semantic dictionary to obtain the semantic features [30] suggested an improved extraction method for the keyword (Extended TF). Document clustering by consensus and classification (DCCC) model is implemented in [28] to perform cluster based classification on the limited high dimensional datasets with limited dimensions.

## III. A Hybrid Document Clustering based Classification Framework

Fig. 1 describes the proposed cluster-based document classification framework on large document sets. Initially, document pre-processing is applied on the input document sets. Here, doc-1, doc-2, …, doc-n represents the input documents for document filtering and feature extraction. Each document is filtered using the Stanford NLP library. In the pre-processing phase, each document is tokenized for word vector generation, stop word removal, stemming and n-gram processing. After the document pre-processing phase, each filtered document is given to hybrid Glove optimization model. The main and contextual key phrases of the glove optimization function are given to similarity measures for key phrase ranking. These main contextual ranked key phrases are given as input to clustering based KNN model and hybrid probabilistic based naïve Bayesian models. These models are used to improve the prediction rate or to minimize the error rate on the large documents sets.

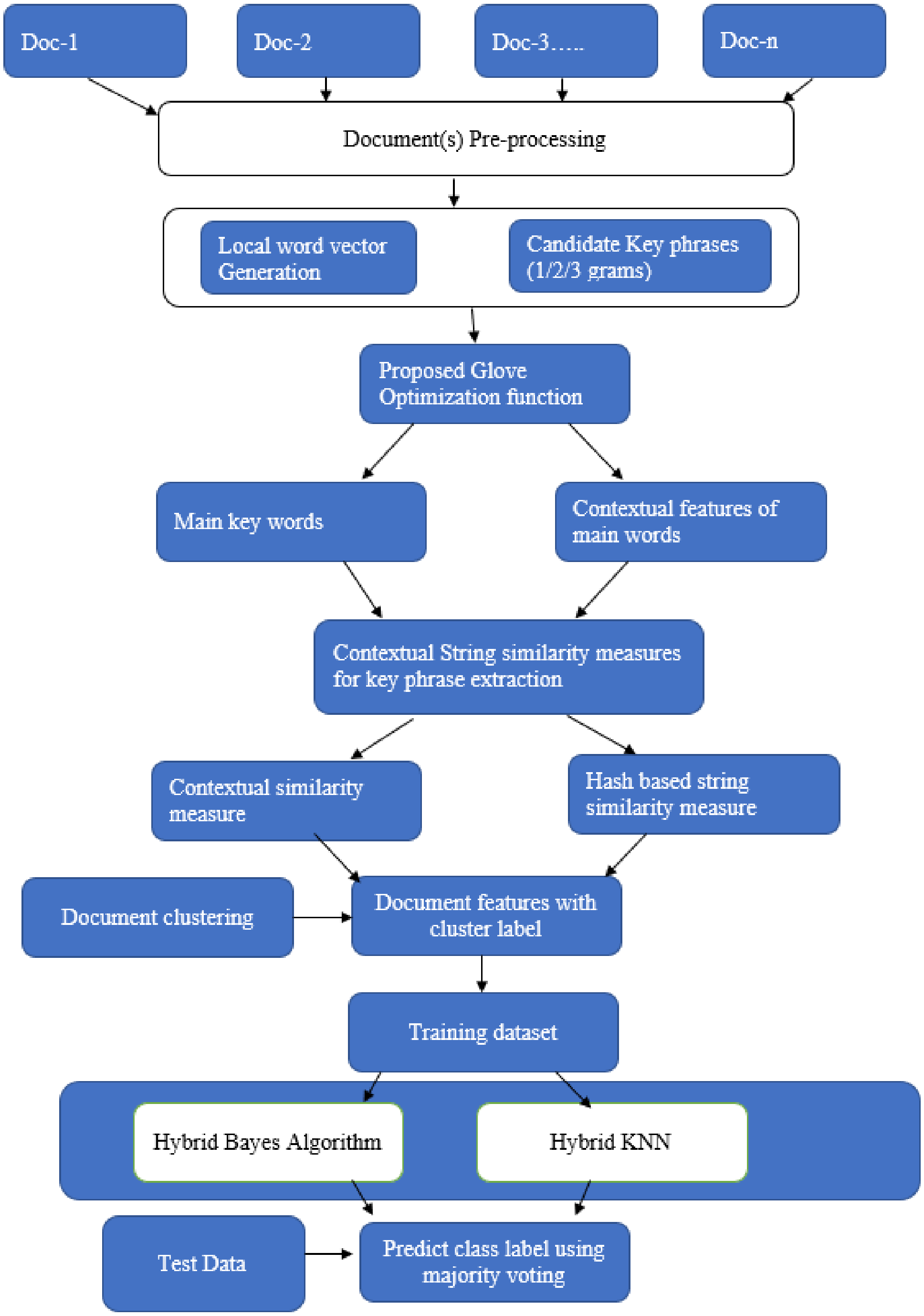
Doc-1
Doc-2
Doc-3…..
Doc-n
Document(s) Pre-processing
Local word vector Generation
Candidate Key phrases (1/2/3 grams)
Proposed Glove Optimization function
Main key words
Contextual features of main words
Contextual String similarity measures for key phrase extraction
Contextual similarity measure
Hash based string similarity measure
Document clustering
Document features with cluster label
Training dataset
Hybrid Bayes Algorithm
Hybrid KNN
Test Data
Predict class label using majority voting

Fig. 1. Proposed Model.

In this work, an advanced cluster-based classification model is designed to improve the document cluster quality and classification accuracy. Most of the traditional document clustering-based classification models are independent of multi-class document classification due to high computational time and accuracy. In this work, a hybrid clustering measure for document classification problem is proposed to minimize the runtime(ms) and classification accuracy.

Most of the information content available on the internet is in the form of text data, so handling of text data is imperative. The method of extracting useful and non-trivial information and knowledge from unstructured text is commonly referred to in data mining. Categorization of text is a key area of study within the field of text mining. The basic purpose of categorizing text is to identify, grasp and organize volumes of text data or documents. The key problems are the complexity of natural languages, and the incredibly high dimensionality of the document feature space that solves this question of classification. Machine learning thus has a dual role: Firstly, we need an efficient data representation to store and process the vast amount of data, as well as an effective learning algorithm to solve the problem. Secondly, to identify unknown documents the accuracy and efficiency of the learning model should be high. The aim is to reduce the dimensionality curse to produce better classification accuracy as well as time consumption due to excessive processing. For the purpose of classification of text documents, the methods for sub-set selection of features employ an evaluation mechanism that is applied to each single word often known as words. There are a variety of factors in assessing the classifier's performance, such as training time, testing time, precision, precision, recall, etc. Proposed model selects a document class based on analyzing the words in the text, which consists mostly of nouns, verbs, and adjectives associated with those nouns. A similar method has been proposed with the use of POS (part-of-speech) tagging where the POS tagger can classify terms in documents by computer via the tags attached to them. A drawback of the Doc2Vec model is the high computational cost of the model construction for each document compared with Word2Vec, GloVe, and fastText. The Doc2Vec model creates a one-time only model for the n-gram text representation. Every term of the document representation vector is a collection of two or more adjacent words in a repository of documents. Another similar approach is to use a fixed section of letters, in which single pieces of letters reflect elements of every document's function vector.

Glove optimization model is used to extract n-gram local word vectors on the filtered data. This model extracts main words and its associated contextual features of main words. Finally, these main words and contextual key word vectors are given to adaptive contextual similarity and string similarity measures. GloVe encodes significance as vector offsets in an embedded space. This model measures the frequency of word co-occurrences in a broad text corpus within a specific window to produce linear significance and uses the factorization of global matrixes and local window modes. The model also has a local cost function and a weighting function to offset uncommon co-occurrences.

**Proposed Glove algorithm consists of following steps:**

*1) Parameter initialization:* Let X is the word co-occurrence matrix and each element Xij represent how often word i appears in context of word j.

$$\mathrm{w}_{\mathrm{i}} : \text{Main word}$$

$$\mathrm{w}_{\mathrm{j}} : \text{Context word}$$

$$\mathrm{b}_{\mathrm{i}}, \mathrm{b}_{\mathrm{j}} : \text{main and context bias values}$$

$$\theta = \mathrm{Min}\{\mathrm{b}_{\mathrm{i}}, \mathrm{b}_{\mathrm{j}}\}.\mathrm{D}((\mathrm{w}_{\mathrm{i}}.\mathrm{b}_{\mathrm{i}}), (\mathrm{w}_{\mathrm{j}}, \mathrm{b}_{\mathrm{j}}))$$

*2)* Define soft constraints for each word pair:

$$\begin{aligned} \mathrm{C} &= \mathrm{CostFunction} \\ &= \mathrm{b}_{\mathrm{i}} \mathrm{w}_{\mathrm{i}}^{\mathrm{T}} \mathrm{w}_{\mathrm{j}} + \mathrm{b}_{\mathrm{j}} \mathrm{w}_{\mathrm{i}}^{\mathrm{T}} \mathrm{w}_{\mathrm{j}} + \theta \\ &\quad -(\log(\mathrm{X}_{\mathrm{ij}})/\max\{\|\mathrm{w}_{\mathrm{i}}\|,\|\mathrm{w}_{\mathrm{j}}\|\}) \end{aligned}$$

$$\eta = \mathrm{weight} = \mathrm{f}(\mathrm{X}_{\mathrm{ij}}) = \begin{cases} (\dfrac{\mathrm{X}_{\mathrm{ij}}}{\mathrm{x}_{\max}})^{\alpha} & \text{if } \mathrm{X}_{\mathrm{ij}} < \mathrm{XMAX} \\ 1 & \text{otherwise} \end{cases}$$

*3)* Define a cost function

$$\begin{aligned} \mathrm{J} = \sum_{\mathrm{i}=1}^{\mathrm{V}} \sum_{\mathrm{j}=1}^{\mathrm{V}} \eta.(\mathrm{b}_{\mathrm{i}} \mathrm{w}_{\mathrm{i}}^{\mathrm{T}} \mathrm{w}_{\mathrm{j}} + \mathrm{b}_{\mathrm{j}} \mathrm{w}_{\mathrm{i}}^{\mathrm{T}} \mathrm{w}_{\mathrm{j}} + \theta \\ -(\log(\mathrm{X}_{\mathrm{ij}})/\max\{\|\mathrm{w}_{\mathrm{i}}\|,\|\mathrm{w}_{\mathrm{j}}\|\}))^2 \end{aligned}$$

The Proposed Glove model is optimized by using the following formula.

$$\begin{aligned} \frac{\partial J}{\partial w_i} = b_i w_j C = b_i w_j (b_i w_i^T w_j + b_j w_i^T w_j + \theta \\ -(\log(X_{ij}) / \max\{\| \mathrm{w}_i \|, \| \mathrm{w}_j \|\})) \end{aligned}$$

$$\begin{aligned} \frac{\partial J}{\partial w_j} = b_i w_i C = b_i w_i (b_i w_i^T w_j + b_j w_i^T w_j + \theta \\ -(\log(X_{ij}) / \max\{\| \mathrm{w}_i \|, \| \mathrm{w}_j \|\})) \end{aligned}$$

Update $\mathrm{w}_{\mathrm{i}}$ and $\mathrm{w}_{\mathrm{j}}$ using learning theta.

In the contextual similarity measure, the similarity between the glove features are evaluated to find the contextual phrases in the biomedical or any textual document sets. Here, the dissimilarity index is used to compute the non-correlated features among the large number of candidate patterns. Finally, contextual glove similarity index is computed by using the dissimilarity measure.

**Hash based Similarity Measure**

The hash-based string similarity is given as:

Let p and q are the big integer which are randomly selected, k is the big prime integer and x is the input word vector then the hash integer of each word in the word vector is given as

H(x)=((px+q).m)xor(k)

Hash based similarity measure is used to find the connecting string similarity between the key phrases. Thus, if a sentence starting with the connecting word is included in the key-phrase, its preceding sentence is also included in the key phrase despite of its rank.

**Hybrid Cluster based KNN**

Input: Let k be the number of nearest neighbor documents,

$D_t$ be the input training documents set.

Output: Classified Documents.

Procedure:

*1)* Read 'k' value and input training $D_t$.

*2)* Apply k-means document clustering algorithm by using the optimal weighted term distance. Compute the weighted term distance to each contextual key phrase of hybrid glove method to the test documents.

$$\psi(TF_{t,d}) = \eta . \frac{t\,f_{t,d} \times \log \frac{\sqrt{T_c}}{T_t}}{[(\sum_{t=1}^{n} t\,f_{t,d}^2) \times \log(\frac{|D|}{\sqrt{T_c}})]} \quad (1)$$

Where $T_t = \sum_{d=1}^{D} t\,f_{t,d}$ where $t\,f_{t,d}, \eta > 0$

and $T_c = \sum_{d=1}^{D} \sum_{t} t\,f_{t,d}$

*3)* Compute the contextual cluster similarity index between the two document sets using the cluster similarity measure.

$$S\,(CM(d_i),CM(d_j)_j) = \psi(TF_{t,d}) . \frac{\sum_{k=1}^{n} t_{ik} \times t_{jk}}{\sqrt{\sum_{k=1}^{n} t_{ik}^2} \sqrt{\sum_{k=1}^{n} t_{jk}^2}}$$

*4)* To each test document in the training data, compute classification score using the following formula.

$$KScore\,(D_t, C_k) = \sum_{d \in DK} S\,(CM(d_i), CM(d_j)) \times P(D_i, C_j)$$

$$P(D_i, C_j) = \begin{cases} 1 & D_i \in Cj \\ 0 & D_i \notin C_j \end{cases}$$

*5)* To each test document in the training data, predict the classification accuracy and error rate.

*6)* Sort top k documents in each cluster with high classification accuracy.

**Hybrid Bayesian Estimation model**

In the proposed probabilistic based Naïve Bayes is used to predict the contextual main word in the hybrid glove method in the given cluster C.

$$P(C_m \mid t_{cw}) > P(C_n \mid t_{cw}) \text{ for } n \neq m$$

Using Eq (1), $P(C_m / t_{cw})$ is maximized.

According to Bayes Theory,

$$P(C_m \mid t_{cw}) = \frac{P(t_{cw} \mid C_m) P(C_m)}{P(t_{cw})}$$

In most of the test mining models, main contextual words are independent to each documents cluster. Also, as the feature space is increasing in size, the computation of priori estimations is performed using the following equations:

$$\theta(P(t_{cw} \mid c_i), P(t_{cw} \mid c_j))$$
$$= \sum_{t \in d} P(t_{cw} \mid c_i) . \log_2 (\frac{P(t_{cw} \mid c_i)}{P(t_{cw})} + 1) / P(t_{cw} \mid c_j))$$
$$\psi(t_{cw}) = 1 + \sum_{k} P(t_{cw} \mid c_i)^2 \cdot P(c_i \mid t_{cw})^2 \cdot (P(t_{cw} \mid c_j))^2$$

$$P(t_{cw} \mid C_m) = (\prod_{r=1}^{n} P(t_r \mid C_m).$$
$$\max\{\prod_{p=1}^{m} P(C_m)) /$$
$$\sum_{i,j}^{|D|} \{\theta(P(t_{cw} \mid c_i), P(t_{cw} \mid c_j)) + \theta(P(t_{cw} \mid c_i), P(t_{cw} \mid c_j))\}$$

One of the simplest machine learning algorithms is the nearest neighbor algorithm. The purpose of the education process is to store the vectors and class marks of the training documents. Documents are converted into text-classified representations in the phrase of training. The most frequently used vector space model is document representation. Each document in this model is represented by a vector, which shows the weight of one word in a document in each entry. One weighing approach is tf-idf (duration frequency-inverse frequency of the document) and the wij (duration: frequency - inverse frequency of the document).

## IV. Experimental Results

The experiments are conducted on various data sets for the main sentence extraction. Every dataset is pre-processed by deleting the stop words and word stemming. In this article,we use the Wikipedia 2014 Glove and Gigaword with 5 billion vocabulary tokens. In http:/nlp.stanford.edu/ projects/glove /size, the developers of the Glove provide the term embedding vectors. We have used a window size 15 and a minimum size 10 to know the GloVe vectors. The similarity between objects is determined in the input vectors as a dot product. Co-occurrence is a strong foundation which encompasses many forms of element similarity. For word similarity, we used the stringsim and contextual similarity measures to demonstrate our model's capacity, a well-known dataset for the English evaluation of similarity. For the evaluation of the proposed model, experimental results are being simulated on text documents such as real time biomedical databases, ChEBI, biocause, PHAEDRA corpus. The cluster score and main contextual feature obtained in iteration 1 for ChEBI is furnished in Appendix-I.

Table I illustrates the experimental analysis of proposed clustering-based document classification model to the conventional classification algorithms using true positive rate. In these results, proposed hybrid cluster based naïve Bayesian model has better true positive rate than the conventional models on different text document datasets.

Fig. 2 illustrates the experimental analysis of proposed clustering-based document classification model to the conventional classification algorithms using true positive rate. In these results, proposed hybrid cluster based KNN model has better true positive rate than the conventional models on different text document datasets.

TABLE I. PERFORMANCE RESULTS OF PROPOSED CLUSTERING-BASED HYBRID NAÏVE BAYESIAN ALGORITHM TO THE CONVENTIONAL MODELS USING TRUE POSITIVE RATE

| Models | Reuters-21587 | SemEval | Ohusumed | Biomedical |
|---|---|---|---|---|
| SVM | 0.81 | 0.85 | 0.87 | 0.91 |
| KNN | 0.83 | 0.83 | 0.89 | 0.9 |
| TFID | 0.85 | 0.91 | 0.91 | 0.9 |
| Naive_Bayes | 0.87 | 0.85 | 0.92 | 0.92 |
| DCCC | 0.956 | 0.923 | 0.901 | 0.914 |
| HCNB | 0.96 | 0.94 | 0.95 | 0.97 |

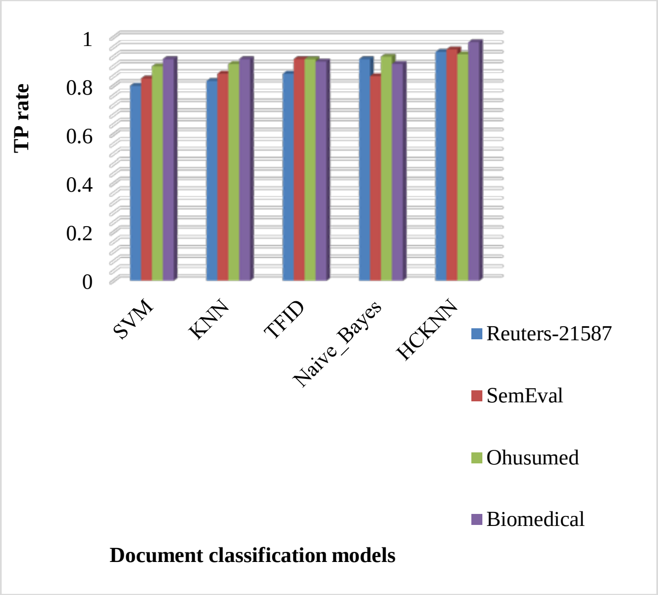

Fig. 2. Performance Results of Proposed Clustering based Hybrid KNN Algorithm to the Conventional Models using True Positive Rate.

TABLE II. PERFORMANCE RESULTS OF PROPOSED CLUSTERING-BASED HYBRID NAÏVE BAYESIAN ALGORITHM TO THE CONVENTIONAL MODELS USING ACCURACY MEASURE

| Models | Reuters-21587 | SemEval | Ohusumed | Biomedical |
|---|---|---|---|---|
| SVM | 0.77 | 0.79 | 0.86 | 0.89 |
| KNN | 0.83 | 0.84 | 0.91 | 0.89 |
| TFID | 0.86 | 0.9 | 0.9 | 0.89 |
| Naive_Bayes | 0.91 | 0.86 | 0.9 | 0.9 |
| DCCC | 0.935 | 0.962 | 0.932 | 0.957 |
| HCNB | 0.95 | 0.97 | 0.94 | 0.97 |

Table II illustrates the performance analysis of proposed clustering-based document classification model to the conventional classification algorithms using accuracy measure. In these results, proposed hybrid cluster based naïve Bayesian model has better average accuracy rate than the conventional models on different text document datasets.

Fig. 3 describes the performance analysis of proposed clustering-based document classification model to the conventional classification algorithms using accuracy measure. In these results, proposed hybrid cluster based KNN approach has better average accuracy rate than the conventional models on different text document datasets.

Fig. 4 illustrates the performance analysis of proposed clustering-based document classification model to the conventional classification algorithms using error rate measure. In these results, proposed hybrid cluster based naïve Bayesian model has better average accuracy rate than the conventional models on different text document datasets.

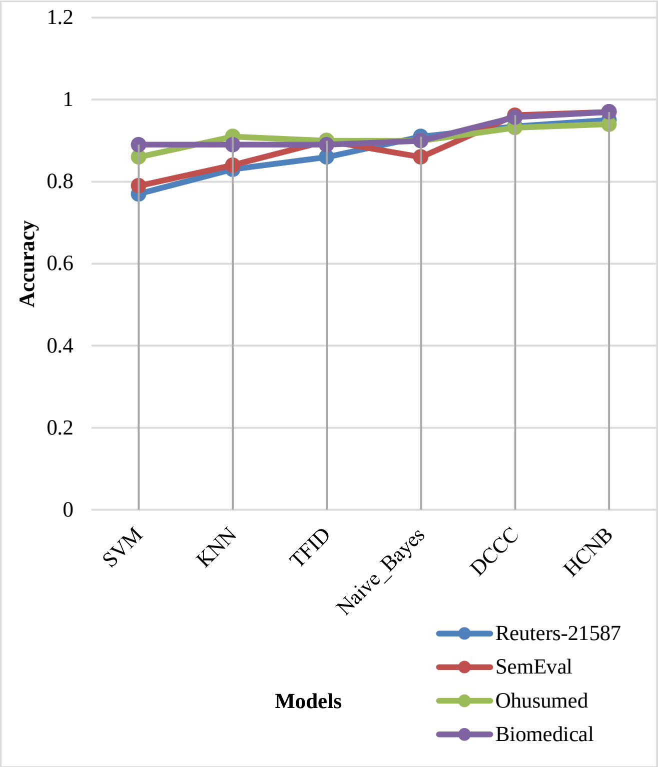

Fig. 3. Performance Results of Proposed Clustering-based Hybrid KNN Model to the Conventional Models using Accuracy Measure.

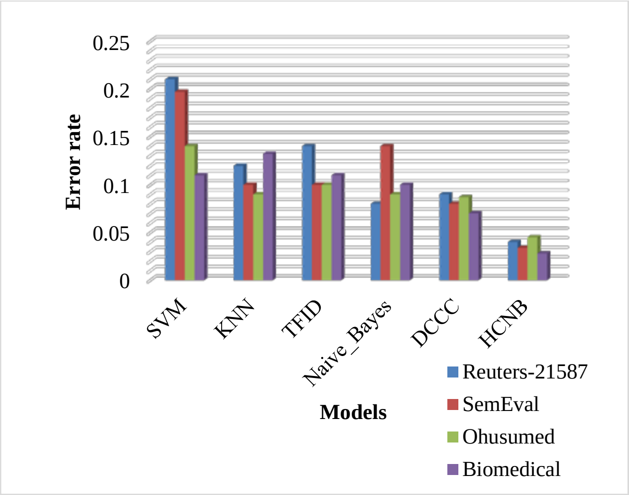

Fig. 4. Performance Results of Proposed Clustering-based Hybrid Naïve Bayesian Model to the Conventional Models for Error Rate Estimation.

TABLE III. COMPARATIVE ANALYSIS OF PRESENT TECHNIQUE TO THE CONVENTIONAL TECHNIQUES BY USING ERROR RATE ESTIMATION ON DIFFERENT MICROARRAY DATASET

| Models | Reuters-21587 | SemEval | Ohusumed | Biomedical |
|---|---|---|---|---|
| SVM | 0.21 | 0.197 | 0.14 | 0.11 |
| KNN | 0.12 | 0.1 | 0.09 | 0.132 |
| TFID | 0.14 | 0.1 | 0.1 | 0.11 |
| Naive_Bayes | 0.08 | 0.14 | 0.09 | 0.1 |
| DCCC | 0.09 | 0.08 | 0.087 | 0.07 |
| HCNB | 0.04 | 0.034 | 0.045 | 0.028 |

Table III describes the performance analysis of proposed clustering-based document classification model to the conventional classification algorithms using error rate measure. In these results, proposed hybrid cluster based KNN approach has better average accuracy rate than the conventional models on different text document datasets.

## V. CONCLUSION

Document classification is one of the major problems in large and high dimensional feature space. As the size of contextual features in the documents sets increases, it is difficult to classify the documents using the traditional glove, TF-D and word2vec methods. In this paper, an advanced document clustering-based classification model is implemented on the large inter and intra feature variation document sets. In this work, a hybrid document clustering similarity index is optimized to find the essential key document clusters based on the contextual keywords. Experimental results show that the clustering-based document classification models have better statistical performance than the conventional approaches on large document sets.

## REFERENCES

[1] A. K. Abasi, A. T. Khader, M. A. Al-Betar, S. Naim, S. N. Makhadmeh, and Z. A. A. Alyasseri, “Link-based multi-verse optimizer for text documents clustering,” Applied Soft Computing, vol. 87, p. 106002, Feb. 2020, doi: 10.1016/j.asoc.2019.106002.

[2] S. N. B. Bhushan and A. Danti, “Classification of compressed and uncompressed text documents,” Future Generation Computer Systems, vol. 88, pp. 614–623, Nov. 2018, doi: 10.1016/j.future.2018.04.054.

[3] S. A. Curiskis, B. Drake, T. R. Osborn, and P. J. Kennedy, “An evaluation of document clustering and topic modelling in two online social networks: Twitter and Reddit,” Information Processing & Management, vol. 57, no. 2, p. 102034, Mar. 2020, doi: 10.1016/j.ipm.2019.04.002.

[4] F. F. dos Santos, M. A. Domingues, C. V. Sundermann, V. O. de Carvalho, M. F. Moura, and S. O. Rezende, “Latent association rule cluster based model to extract topics for classification and recommendation applications,” Expert Systems with Applications, vol. 112, pp. 34–60, Dec. 2018, doi: 10.1016/j.eswa.2018.06.021.

[5] M. Franco-Salvador and L. A. Leiva, “Multilingual phrase sampling for text entry evaluations,” International Journal of Human-Computer Studies, vol. 113, pp. 15–31, May 2018, doi: 10.1016/j.ijhcs.2018.01.006.

[6] M. Fu, H. Qu, L. Huang, and L. Lu, “Bag of meta-words: A novel method to represent document for the sentiment classification,” Expert Systems with Applications, vol. 113, pp. 33–43, Dec. 2018, doi: 10.1016/j.eswa.2018.06.052.

[7] Z. Gero and J. Ho, “PMCVec: Distributed phrase representation for biomedical text processing,” Journal of Biomedical Informatics: X, vol. 3, p. 100047, Sep. 2019, doi: 10.1016/j.yjbinx.2019.100047.

[8] R. Janani and S. Vijayarani, “Text document clustering using Spectral Clustering algorithm with Particle Swarm Optimization,” Expert Systems with Applications, vol. 134, pp. 192–200, Nov. 2019, doi: 10.1016/j.eswa.2019.05.030.

[9] D. Ji, P. Tao, H. Fei, and Y. Ren, “An end-to-end joint model for evidence information extraction from court record document,” Information Processing & Management, vol. 57, no. 6, p. 102305, Nov. 2020, doi: 10.1016/j.ipm.2020.102305.

[10] R. Joshi, R. Prasad, P. Mewada, and P. Saurabh, “Modified LDA Approach For Cluster Based Gene Classification Using K-Mean Method,” Procedia Computer Science, vol. 171, pp. 2493–2500, Jan. 2020, doi: 10.1016/j.procs.2020.04.270.

[11] D. Kim, D. Seo, S. Cho, and P. Kang, “Multi-co-training for document classification using various document representations: TF–IDF, LDA, and Doc2Vec,” Information Sciences, vol. 477, pp. 15–29, Mar. 2019, doi: 10.1016/j.ins.2018.10.006.

[12] L. Kong et al., “Leveraging multiple features for document sentiment classification,” Information Sciences, vol. 518, pp. 39–55, May 2020, doi: 10.1016/j.ins.2020.01.012.

[13] A. Kostkina, D. Bodunkov, and V. Klimov, “Document Categorization Based on Usage of Features Reduction with Synonyms Clustering in Weak Semantic Map,” Procedia Computer Science, vol. 145, pp. 288–292, Jan. 2018, doi: 10.1016/j.procs.2018.11.061.

[14] N. K. Mishra and P. K. Singh, “FS-MLC: Feature selection for multi-label classification using clustering in feature space,” Information Processing & Management, vol. 57, no. 4, p. 102240, Jul. 2020, doi: 10.1016/j.ipm.2020.102240.

[15] B. Mutlu, E. A. Sezer, and M. A. Akcayol, “Multi-document extractive text summarization: A comparative assessment on features,” Knowledge-Based Systems, vol. 183, p. 104848, Nov. 2019, doi: 10.1016/j.knosys.2019.07.019.

[16] J. Park, C. Park, J. Kim, M. Cho, and S. Park, “ADC: Advanced document clustering using contextualized representations,” Expert Systems with Applications, vol. 137, pp. 157–166, Dec. 2019, doi: 10.1016/j.eswa.2019.06.068.

[17] R. A. Sinoara, J. Camacho-Collados, R. G. Rossi, R. Navigli, and S. O. Rezende, “Knowledge-enhanced document embeddings for text classification,” Knowledge-Based Systems, vol. 163, pp. 955–971, Jan. 2019, doi: 10.1016/j.knosys.2018.10.026.

[18] F. Yi, B. Jiang, and J. Wu, “Topic Modeling for Short Texts via Word Embedding and Document Correlation,” IEEE Access, vol. 8, pp. 30692–30705, 2020, doi: 10.1109/ACCESS.2020.2973207.

[19] Y. Song, S. Upadhyay, H. Peng, S. Mayhew, and D. Roth, “Toward any-language zero-shot topic classification of textual documents,” Artificial

Intelligence, vol. 274, pp. 133–150, Sep. 2019, doi: 10.1016/j.artint.2019.02.002.

[20] Y. Wu, S. Zhao, and W. Li, "Phrase2Vec: Phrase embedding based on parsing," Information Sciences, vol. 517, pp. 100–127, May 2020, doi: 10.1016/j.ins.2019.12.031.

[21] M. P. Akhter, Z. Jiangbin, I. R. Naqvi, M. Abdelmajeed, A. Mehmood, and M. T. Sadiq, "Document-Level Text Classification Using Single-Layer Multisize Filters Convolutional Neural Network," IEEE Access, vol. 8, pp. 42689–42707, 2020, doi: 10.1109/ACCESS.2020.2976744

[22] Y. A. Alhaj, J. Xiang, D. Zhao, M. A. A. Al-Qaness, M. Abd Elaziz, and A. Dahou, "A Study of the Effects of Stemming Strategies on Arabic Document Classification," IEEE Access, vol. 7, pp. 32664–32671, 2019, doi: 10.1109/ACCESS.2019.2903331.

[23] S. Eken, H. Menhour, and K. Köksal, "DoCA: A Content-Based Automatic Classification System Over Digital Documents," IEEE Access, vol. 7, pp. 97996–98004, 2019, doi: 10.1109/ACCESS.2019.2930339.

[24] G. Li, Z. Wang, and Y. Ma, "Combining Domain Knowledge Extraction With Graph Long Short-Term Memory for Learning Classification of Chinese Legal Documents," IEEE Access, vol. 7, pp. 139616–139627, 2019, doi: 10.1109/ACCESS.2019.2943668.

[25] Y. Ma, P. Zhang, and J. Ma, "An Ontology Driven Knowledge Block Summarization Approach for Chinese Judgment Document Classification," IEEE Access, vol. 6, pp. 71327–71338, 2018, doi: 10.1109/ACCESS.2018.2881682.

[26] E. L. Park, S. Cho, and P. Kang, "Supervised Paragraph Vector: Distributed Representations of Words, Documents and Class Labels," IEEE Access, vol. 7, pp. 29051–29064, 2019, doi: 10.1109/ACCESS.2019.2901933.

[27] J. Rashid et al., "Topic Modeling Technique for Text Mining Over Biomedical Text Corpora Through Hybrid Inverse Documents Frequency and Fuzzy K-Means Clustering," IEEE Access, vol. 7, pp. 146070–146080, 2019, doi: 10.1109/ACCESS.2019.2944973.

[28] A. M. Sheri, M. A. Rafique, M. T. Hassan, K. N. Junejo, and M. Jeon, "Boosting Discrimination Information Based Document Clustering Using Consensus and Classification," IEEE Access, vol. 7, pp. 78954–78962, 2019, doi: 10.1109/ACCESS.2019.2923462.

[29] T. Temel, "High-accuracy document classification with a new algorithm," Electronics Letters, vol. 54, no. 17, pp. 1028–1030, 2018, doi: 10.1049/el.2018.0790.

[30] R. Zhao and K. Mao, "Fuzzy Bag-of-Words Model for Document Representation," IEEE Transactions on Fuzzy Systems, vol. 26, no. 2, pp. 794–804, Apr. 2018, doi: 10.1109/TFUZZ.2017.2690222.

## APPENDIX-I

```
Cluster Score : which its vector value :0.96
Cluster Score : were its vector value :0.952
Cluster Score : separated its vector value :0.961
Cluster Score : by its vector value :0.95
Cluster Score : a its vector value :0.938
Cluster Score : washout its vector value :0.941
Cluster Score : period its vector value :0.926
Cluster Score : of its vector value :0.934
Cluster Score : 4 its vector value :0.972
Cluster Score : weeks, its vector value :0.931
Cluster Score : was its vector value :0.923
Cluster Score : used. its vector value :0.974
Cluster Score : its vector value :0.935
Cluster Score : In its vector value :0.958
Cluster Score : each its vector value :0.95
Cluster Score : phase its vector value :0.97
```

```
Cluster Score : ten its vector value :0.942
Cluster Score : healthy its vector value :0.922
Cluster Score : volunteers its vector value :0.973
Cluster Score : took its vector value :0.961
Cluster Score : 200 its vector value :0.946
Cluster Score : mg its vector value :0.946
Cluster Score : itraconazole its vector value :0.964
Cluster Score : or its vector value :0.933
Cluster Score : matched its vector value :0.923
Cluster Score : placebo its vector value :0.969
Cluster Score : orally its vector value :0.97
Cluster Score : once its vector value :0.959
Cluster Score : daily its vector value :0.952
Cluster Score : for its vector value :0.958
Cluster Score : 4 its vector value :0.951
Cluster Score : days its vector value :0.947
Cluster Score : according its vector value :0.941
Cluster Score : to its vector value :0.965
Cluster Score : a its vector value :0.925
Cluster Score : randomization its vector value :0.95
Cluster Score : schedule. its vector value :0.952
Cluster Score : its vector value :0.926
Cluster Score : On its vector value :0.94
Cluster Score : day its vector value :0.929
Cluster Score : 4, its vector value :0.949
Main-Contextual Works List :===> 1) 2) The 3) 0.040703042514465194
 Main-Contextual Works List :===> 1) 2.6-fold 2) 2.0-fold 3)
0.02630626620960846
 Main-Contextual Works List :===> 1) cerivastatin, 2) major 3)
0.004520272696979648
 Main-Contextual Works List :===> 1) half-life 2) cerivastatin 3)
0.010059523995966884
 Main-Contextual Works List :===> 1) days 2) for 3) 0.030376280419389695
 Main-Contextual Works List :===> 1) days 2) according 3)
0.03075857667846457
 Main-Contextual Works List :===> 1) were 2) two 3)
0.018200219160994166
 Main-Contextual Works List :===> 1) metabolite 2) active 3)
0.02934636541187912
 Main-Contextual Works List :===> 1) metabolite 2) M-23, 3)
0.03636079620998193
 Main-Contextual Works List :===> 1) increased 2) 1.8-fold 3)
0.002126193197175103
 Main-Contextual Works List :===> 1) concentrations 2) its 3)
0.022965243547397464
 Main-Contextual Works List :===> 1) concentration 2) of 3)
0.042554880058576515
 Main-Contextual Works List :===> 1) HMG-CoA 2) reductase 3)
```

```
0.012201337576894449
 Main-Contextual Works List :===> 1) 3-hydroxy-3-methylglutaryl 2)
competitive 3) 0.02601678012887241
 Main-Contextual Works List :===> 1) cerivastatin 2) half-life 3)
0.018770381479517345
 Main-Contextual Works List :===> 1) leads 2) pathway 3)
0.04422042394663061
 Main-Contextual Works List :===> 1) decreased 2) AUC(0-24h) 3)
0.01774289121084388
 Main-Contextual Works List :===> 1) a 2) period 3)
0.0014494343061637062
 Main-Contextual Works List :===> 1) 2.4-fold, 2) 0.001) 3)
0.019349273848956093
 Main-Contextual Works List :===> 1) effects 2) itraconazole, 3)
0.013057016278279616
 Main-Contextual Works List :===> 1) A 2) competitive 3)
0.02601678091115786
 Main-Contextual Works List :===> 1) increased 2) < 3)
0.016245777323349198
 Main-Contextual Works List :===> 1) period 2) 4 3) 0.00628237514390879
 Main-Contextual Works List :===> 1) inhibitors 2) were 3)
0.003972892951397503
 Main-Contextual Works List :===> 1) 0.001) 2) by 3) 0.01717860815440692
 Main-Contextual Works List :===> 1) the 2) by 3) 0.0171786176013269
 Main-Contextual Works List :===> 1) 24 2) h. 3) 0.01858117239581096
 Main-Contextual Works List :===> 1) 0.001) 2) itraconazole. 3)
0.031160433362267235
 Main-Contextual Works List :===> 1) cerivastatin 2) 28% 3)
0.045072254995085534
 Main-Contextual Works List :===> 1) zero 2) time 3)
0.027113291660797015
 Main-Contextual Works List :===> 1) 2.4-fold, 2) 1.1-fold 3)
0.043091828461683
 Main-Contextual Works List :===> 1) days 2) to 3) 0.001815575624543022
 Main-Contextual Works List :===> 1) < 2) itraconazole. 3)
0.03084041553411316
 Main-Contextual Works List :===> 1) the 2) cerivastatin 3)
0.009743819202678772
 Main-Contextual Works List :===> 1) active 2) inhibitors 3)
0.031801292511466094
 Main-Contextual Works List :===> 1) 0.05), 2) 28% 3)
0.044967902949240274
 Main-Contextual Works List :===> 1) time 2) curve 3)
0.019054502825359795
 Main-Contextual Works List :===> 1) increased 2) by 3)
0.017746488507059024
 Main-Contextual Works List :===> 1) of 2) elimination 3)
0.021782216414505815
```

```
 Main-Contextual Works List :===> 1) with 2) design 3)
0.019750504038797516
 Main-Contextual Works List :===> 1) 3.2-fold 2) lactone 3)
0.01955636998633503
 Main-Contextual Works List :===> 1) hydroxyitraconazole 2) measured 3)
0.013681591494622555
 Main-Contextual Works List :===> 1) (Cmax) 2) serum 3)
0.013239864690434963
 Main-Contextual Works List :===> 1) zero 2) curve 3)
0.018842142113686126

Main-Contextual Works List :===> 1) in 2) resulting 3)
0.025708380609795373
 Main-Contextual Works List :===> 1) concentration 2) mean 3)
7.985778292819061E-4
 Main-Contextual Works List :===> 1) orally 2) or 3)
0.022261480766409793
 Main-Contextual Works List :===> 1) and 2) reductase 3)
0.012228039361377256
 Main-Contextual Works List :===> 1) 4 2) to 3) 0.00328600130796709
 Main-Contextual Works List :===> 1) (P 2) increased 3)
0.03118034855179173
Iteration #1 , cost = 244.0632399474855
 Avg Features 48
 Runtime(ms) of Features 4617.0
 Runtime(ms) of Similarity rank 4927.0
Classification accuracy :98.4
```