International Journal of Engineering & Technology, 7 (1.1) (2018) 466-471

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

An novel cluster based feature selection and document classification model on high dimension trec data

P. Lalitha Kumari[1]*, Ch.Satyanarayana [2]

Research Scholar[1], Professor[2]
[1,2] Department of Computer Science and Engineering
[1,2] JNTUK, Kakinada ,A.P., India

Abstract

TREC text documents are complex to analyze the features its relevant similar documents using the traditional document similarity measures. As the size of the TREC repository is increasing, finding relevant clustered documents from a large collection of unstructured documents is a challenging task. Traditional document similarity and classification models are implemented on homogeneous TREC data to find essential features for document entities that are similar to the TREC documents. Also, most of the traditional models are applicable to limited text document sets for text analysis. The main issues in the traditional text mining models in TREC repository include :1) Each document is represented in vector form with many sparsity values 2) Failed to find the document semantic similarity between the intra and inter clusters 3) High mean squared error rate. In this paper, novel feature selection based clustered and classification model is proposed on large number of different TREC repositories. Traditional latent Semantic Indexing and document clustering models are failed to find the topic relevance on large number of TREC clinical text document sets due to computational memory and time. Proposed document feature selection and clustered based classification model is applied on TREC clinical benchmark datasets. From the experimental results, it is proved that the proposed model is efficient than the existing models in terms of computational memory, accuracy and error rate are concerned.

Keywords: *TREC Datasets, Information Retrieval, Document Clustering And Classification.*

1. Introduction

With the tremendous growth of structured and unstructured textual defect data in the past decade, context based TREC analysis has become equal importance in text mining applications. When there is only a sparse proportion of a labelled minority class sample, many classifiers tend to over predict the majority class, essentially ignoring the minority class. A large number of text similarity prediction models have been implemented on the textual datasets for document classification. The text localization technique, however, can be an efficient document classification method because it employs a sequence of pre-processing methods other than a single technique to generate better results for TREC datasets. Most of the localization models are implemented on learning techniques for feature prediction in text documents. Prediction of document features in TREC data is an essential task for the prediction of new documents.

The problem of traditional feature prediction has been implemented in three phases. Identification of the TREC documents in the datasets using the keyword based matching. Consistency and conciseness are the main features of the similarity measure. The judgment of text feature selection quality through the use of information retrieval techniques indicates the third measure.. The relationship between the features and the measures are identified through the use of linear mixed-effects regression models [1].

TREC document analysis is essential to developers especially in large-scale text analysis systems because they are used to fix feature extraction based document classification. The identification of a new TREC documents comes along with two vital tasks. The first task is the problem of identification of features in the TREC training data[2-4].

The second task is called the feature based document clustering and classification.

However, the establishment of how the historical TREC report relates to the new TREC report still poses challenges. A decision of how accurate the semi-automatic fixer recommendation and prediction of the levels of context similarity is made from the similarity between queries, their features, and historical reports. TREC reports can be classified into their corresponding topics through topic modelling. TREC reports which share a topic are categorized under the same classification. The REP algorithm which was proposed by [4] is a similarity function that introduces topic modelling as an additional feature in comparison of the TREC report for similarity. K-Nearest Neighbour classification is used along with the enhanced REP to find similarity between new TRECs document categories[5].

Initially, the document data give rise to features, and these features are evaluated in the process of document clustering. Mostly high-dimensional document space's hard to handle, pre-process and cluster due to large amounts of document sets. To improve the learning of the clustering algorithm, the numbers of samples are required to be learned according to its dimension. Conceptually, this document space is a sub-space of low dimensionality, and it is wrapped with ambient space. Due to this dimensionality issue, many dimension reduction methods were developed to resolve the above problem. The main objective of this method is, to decrease the document dimensions and enhance the performance and efficiency. Thus, through dimensionality reduction methods, dimensional feature spaces of high-dimensional documents are minimized so the conventional Clustering schemes are used to achieve the better clustering performance. Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) are two most used techniques for feature selection and dimension reduction [6]. These algorithms are implemented in the various fields such as pattern recognition, text mining and gene extraction and data analysis. In supervised machine learning algorithm, training data are required for the process of estimation or prediction.

Classification can be defined as a special kind of learning model which is responsible for categorization of different gene-disease datasets. These datasets are classified into set of finite or infinite classes. Apart from supervised and unsupervised machine learning approaches, there are two other machine learning techniques generally used for classification are:- regression and clustering. A learning function generally maps original data into their real-value variable in the process of regression. This technique can estimate the predictive variable for every individual sample. The most frequent applications of document classifications are noticed in the field of TREC and research domain[7]. The process of classification is broadly categorized into two parts, those are: training and testing phase. Classification algorithm is responsible to create a classification model with the help of the training set. Later the model performance is evaluated in the testing phase. Many research works have been proposed since years in order to develop a classification algorithm with optimized performance. Some of the popular classification models[3] are briefly described below.

Clustering is categorized under the category of unsupervised learning and here groups are formed according to the similarity of data items. The groups which are built in the process of clustering are known as clusters. Data items having high similarity are included in the same cluster, whereas data items having no similarity or least similarity are included in different clusters[8].

Document clustering groups similar documents using statistical measures on term frequencies, phrase frequencies and sentence frequencies.

The majority of the documents clustering techniques that are in use today are centered on the feature vector spaces, which are broadly used to train document model for text clustering and classification [9]. Each featured vector space specifies documents as a characteristic vector of the terms that occur in all the document collection set. Each document feature vector includes word/ phrase frequencies of the words appearing in that sentence or document. Document Similarity between sentences/documents are examined using one of document similarity measures that are based on such a feature vector or word frequencies, for instance, Jaccard measure and the cosine measure. Clustering techniques based on these vector spaces make use of single word i.e., one gram interpretation only. They do not make use of any word neighborhood or phrase based clustering[10].

Document clustering involves partitioning a set of documents into a specified set of document clusters. The main aim behind the clustering is to find the inherent structure in the data objects and the degree of similarity within the cluster should be minimized. Most of the algorithms are categorized into two groups i.e., partitioning method and non-partitioning method. The partitioning method seeks to partition a document collection into non-overlapping clusters. In recent years, it has been observed that a partitional clustering algorithm is well suited for clustering and summarization of a large set of documents due to its low computational complexity.

The process of feature extraction has high significance in the field of classification. All features are divided into two groups, those are:-

1) According to the first group, features extraction using noisy attributes and contextual information.

2) The second group contains correlated features. Traditional feature extraction models discard noisy features in order to decrease the high dimensional features to a lower dimensional feature.

2. Related work

Most of the text mining models have been used for document analysis, feature identification, and contextual identification using query or keywords. IR based TREC localization techniques are considered better than spectrum based feature localization or change impact analysis because they have better accuracy and a computational cost on high dimensional datasets. The classification scheme bagging is categorized under a special kind of Bootstrap aggregation. Furthermore, the process of bagging also supports all characteristics of machine learning and meta-algorithm[7]. Meta-algorithm can be defined as a specific algorithm which is developed for improvement of stabilization factor. Bagging has wide range of applications in the fields of statistical classification and regression. The process of bagging not only reduces variance, but also limits over fitting. Besides these, there exists another application of bagging classification i.e., decision trees. Some common factors are generally responsible for errors of machine learning algorithms, those are:- noise, bias and variance. Noise is generally defined as an error occurs by the target function. Biases are the targets which are not qualified to be learnt by the classification algorithms.

Hachenberg et.al, proposed a decision tree to classify TRECs into either fast or slow decision tree construction measures[7]. [8] proposed the filtering of TREC reports for analysis of document classification and feature extraction. They developed a technique for predicting features based on the temporal sequence of the activities of the contextual meaning. The method used a hidden Markov model for prediction. They stated that the quantity of fixable TRECs in the future could be predicted by using the Markov, chain model. They also used the Monte Carlo simulation to predict the time taken in feature selection procedure. TREC

document feature selection and classification has numerous IR methods of different complexities with limited document sets. Latent Dirichlet Allocation and Latent Semantic Indexing are some of the intensive computational semantic methods. Vector Space Model is an example of a simple lexical matching method. Due to noisy information, the Information Retrieval methods such as the Latent Dirichlet Allocation (LDA) and Latent Semantic Indexing (LSI) may perform inaccurate on high dimensional datasets. This means that Information Retrieval methods still have a long way to go for them to get the accurate result in their applications. These methods also need to get many false positives to achieve high recall. The Naive Bayes model is more efficient than that of the logistic regression scheme, nearest neighbour, decision tree and neural network, according to Receiver Operating Characteristic (ROC) curve, which is more significantly implemented in the research domain. The model is a simple, less parameterized and efficient one in terms of performance[11].

KNN is a special type of instance-based classification model, in which approximation function is computed locally and it continues until classification occurs. It is also termed as lazy learning because there is no need of training phase like other conventional approaches. These training data are tested in the testing phase. For large datasets, training data are split into smaller partial datasets[12].

Support Vector Machine is a construction-based classification model. It also follows the idea of statistical learning[13]. This algorithm achieves enhanced performance as compared to other existing approaches. With the help of hyperplanes, SVM defines decision boundaries. It also distinguishes data points of different classes, which can solve both linear and nonlinear classification problems. A mapping function is invoked for mapping of the original data points from the input space to a high-dimensional or infinite-dimensional feature space through a kernel function. Relevance Vector Machine (RVM) is an enhanced version of SVM. This model provides better performance rate when compared with other models including SVM. It emphasizes on sparsity and compressed sensing. This approach uses subsets of the training data of SVM which are fewer than that of support vectors.Many methods can be used to achieve this such as the TF-IDF, the VSM (Vector Space Model), and statistical model. Hiew used VSM to achieve more effective duplicate TREC report detection. They used sentence extraction strategy and the clustering approach are used to eliminate information redundancy resulting from the multiple original documents. Clustering is one of the vital tools in data mining and knowledge discovery [20-29] . Due to massive amounts of data collected in databases, cluster analysis has been recently become a highly active topic in data mining. Mostly K-means clustering algorithm is used to group similar objects together. But, it requires the number of clusters to be specified in advance which is considered to be one of the major drawbacks of this algorithm. The ability to automatically cluster similar items together, allows one to determine hidden similarities and key concepts. It also summarizes a large amount of information into a smaller number of clusters.

LDA successfully attempts to improve data modelling over other methods by allowing for documents within corpora to be modelled as collections of topics. The unique and revolutionary idea behind this model is that the topic variable in the model is selected repeatedly within each document allowing for documents to be comprised of multiple topics. This is also the intention of the Probabilistic Latent Semantic Indexing (PLSI) method, but LDA uses a hidden random variable that predicts new documents using the previously unseen documents without over fitting. For the discussion of LDA within this section, main focus is on documents that are comprised of text corpora.

3. Proposed model

The proposed model concentrated on finding document features identification with the document clustering and classification. This model finds quality text document's information from the unstructured data. By merging rich document representations, it resolves textual classification issues in machine learning. Document based vector representation helps as a measure of semantics. The proposed model optimizes the probability based features clustering and classification, which is responsible for construction of predictive classification models. In this model, feature based relational clusters are used to find the probability of a document belonging to a specific TREC category.

Static and dynamic methods are the two categories of automated methods of TREC topic localization. The dynamic approach takes a look at the execution trace of the system, breakpoints and program data to locate feature topics. This approach works by examining whether the program runs successfully under certain input or fails, and it determines the differences. Dynamic methods, therefore, may not work with errors which span over small datasets because they require the features to be run and it is not always feasible. Static methods are based on information retrieval in their function to detect the feature in lines of textual documents. The content may describe events, domain concepts, system attributes and features or exceptions. The advantage of static methods includes independence from specific languages and their low cost of computation.

TREC repositories are the source of the text reports in which data extraction and pre-processing have been done on large data size. In the proposed model, Stanford NLP and Lucene libraries are used as pre-processing. Splitting of sentences and terms, filtering the stop words and stemming all constitute the pre-processing step. Stemming serves to reduce the size of vocabulary and avoid duplication that may constitute the verbs. Pronouns and prepositions represent noise, along with special characters. The list of stop words was acquired from Mallet.

IR-based feature extraction: The probability that an Information Retrieval approach will accurately rank the training files which need to be fixed is highly dependent on the goodness of interpretation of the training files and the TREC report. The three-step pre-process of an information system consists the normalization of texts, removal of stopwords and stemming. The TREC report and the training files are pre-processed, terms that can be analyzed are created, and a calculation of similarity then follows.

Text normalization is responsible for removing the punctuation symbols, tokenizing the terms and splitting the identifiers. Identifiers are parsed using the Abstract Syntax Tree in the source files. A method named "combine Analyzed Score," for example, is divided into "combine," "analyzed" and "score." Filtration of non-essential words from a stopword list then follows, and it serves to reduce cases of noisy

matches and increase the accuracy. An example could be the words "goes" and "going" whose original forms are the word "go." Stemming represents like words with a common root form are represented using the same word. The Porter stemming algorithm3 comes in handy for this step. After pre-processing, indexing of the training files follows through observation of the statistics that are collected and stored such as how many documents bear that term (DF), and how many times in a document the term under analysis appears (TF). The resemblance between the source files' contents and the TREC reports are calculated through the inverse of DF, which is termed as IDF. IDF and TF are useful for vector representation of the training files and the TREC report.

Algorithm steps:

TREC topic and its feature list matrix is tabulated as

Features / Topics	FL_1	FL_2	……	FL_n
T_1	T_1FL_1			
……	……	……	……	……
T_m	T_mFL_1	T_mFL_2	T_mFL_n	T_mFL_n

Each topic weight can be defined by collaborating the topic queries and its feature hits.

$$T_iFL_j = \max\{U_k, \max\{H(T_i\}, \max\{H(FL_j\}\}; \quad (1)$$

T_iFL_j can be obtained by maximizing the Kth TREC topic hit rate.

T_i is the accessed topic relevance.

FL_j is the selected topic feature list.

Procedure:

Select query from the TREC topics list TL.
for each query q in TL
do
if($q \in$ TREClist)
then
extract query q related topics along with feature
TM(topic matrix)$\leftarrow \{q_i, FL_j\}$
End if
Enf for
For each topic query q in TL
Do
Compute the topic weight using (1) and topic relevance ratio using probability estimation as shown below.

Compute Predictive correlation between the two features as

$$\text{Topic Relevance Ratio(TRR)} = \frac{Pro(TF_i / q_i).Pro(TF_i / q)}{N(N-1)}$$

Where N is the total queries
Total topic relevance ratio(TTRR)=TW/TRR.
If(TTRR> λ)
Then
Predictive Correlation PC=Corr(F[i],F[i+1]) /

$$\sum_{i=i}^{N} \text{Prob}(F[i] / F[i+1])$$

If(PC>0)
Then
D' =addFeature(F[i],F[i+1],PC);
End if
Else
Continue
End for
Enf if

D: TREC dataset
m: Required number of clusters for cluster initialization
Step 1: for i=1 to m do

$\mu_i \leftarrow$ Mean of initial random data objects.

$\sigma_i^2 \leftarrow 1$

$\phi_i \leftarrow 1/m$

End for
For k=1 to N do
For i=1 to m do

$$z(n,i) \leftarrow \phi_i [2.\pi.\sigma_i]^{-D^2/2}.e^{-\frac{1}{2.\sigma_i^2}\|x_n-\mu_k\|^2/\min\{D_k\}}$$

End for

$$z(n) \leftarrow \frac{z(n,i)}{\sum_k z(n,i)}$$

End for
// Update cluster parameter
For i=1 to m do

$$\phi_i \leftarrow z(n,k)/N$$

$$\mu_i \leftarrow \frac{\sum_n [z(n,k).x_n]}{\sum_n z(n,k)}$$

$$\sigma_i^2 \leftarrow \frac{\sum_n [z(n,k).\| x_n - \mu_i \|]}{\sum_n z(n,k)}$$

End for
Until convergence of initial m clusters.

Step 2:

D' $\leftarrow$ Optimal initial clusters in step-1
G $\leftarrow$ buildKD-tree(D')
H $\leftarrow$ buildHeap(D')
m $\leftarrow$ Number of clusters
For k=1 to m clusters do
if size(H)>m then
p $\leftarrow$ min(H)
q $\leftarrow$ p.nearest;
remove(H,q)
s $\leftarrow$ OptimalMerge(p,q)

```
Remove_Represent(G,p)
Remove_Represent(G,q)
insert_represent(G,s)
s.nearest ← x  //arbitary cluster in H
for each object x ∈ H do

 if distance(S,x)< distance (s,s.nearest_pt)
        then
          if s.nearest_pt ∈ p or x.nearest ∈ q
              if distance(x,x.nearest_pt)<distance(x,s)
                 x.nearest_pt ← nearest_nearest(G, x, distance(x,s))
              else
                          x.nearest ← s
                Relocate(H,x)
            end if
          end if
        else if distance(x,x.nearest)> distance (x,s)
           then
                x.nearest ← s
          Relocate(H,x)
        end if
   Insert(H,s)
end for
```

Step 3: Classify using naïve Bayesian classifier using the given prediction formula:

$$\text{DocClassify Prob}(H(k), d_{(i)}) = \prod_{j=1}^{N} f_{ij} \cdot \frac{\text{Prob}(H(k)) * \text{Prob}(d_{(i)} / H(k))}{\text{Prob}(d_{(i)})}$$ where

Algorithm 2: OptimalMerge(p,q)

$s \leftarrow p \cup q$

$$s.\text{meanprob} \leftarrow \sum_{\min(p,q)} |p_i - q_i| \{\frac{|p| \cdot \bar{p}}{|p| + |q|}\}$$

```
Temp ← {}
for i=1 to |R| do
   maxD ← 0
   for each object obj in cluster s do
      if i=1
       then
          minD ← distance(obj,s.mean)
       else
          minD ← min{distance(obj,t) : t ∈ Temp
       if (minD>=maxD) then
            maxD ← minD
            maxPt ← obj
       end if
   end for
   Temp ← Temp ∪ {max Pt}
   end for
for each object obj in Temp do
```

$s.\text{represent} \leftarrow s.\text{represent} \cup \{|(1-\eta)| \text{obj} + \eta(s.\text{mean} - \text{obj})\}$

```
return s
end
```

$$\text{Distance}(U,V) \leftarrow \frac{\max |\mu_U(x_i) - \mu_V(x_i)|}{\min |\sigma_U^2(x_i) - \sigma_V^2(x_i)|} . \text{Manhattan}(u, v)$$

4. Experimental results

Experimental results are evaluated on large collection of TREC document sets taken from the TREC website [13]. Experimental results are performed on TREC Medline biomedical repository for document ranking process. Different measures such as document ranking and error rates are used to find the performance of the proposed algorithm on the document extraction algorithm.

ROC and F-measure are the commonly used performance metrics in ranking models. However, due to noise and imbalanced problems, traditional Receiver Operating Characteristics (ROC) and F-measure may not be a good choice. In this ensemble model, a novel phase wise accuracy measures such as geometric mean (GM) and Sum True positive rate are used to evaluate the performance of the proposed model to the traditional ensemble models.

Accuracy Rate $A(D_c, F_i) = |D_c \cap F_i| / |D_c|$

Recall Rate $R(D_c, F_i) = |D_c \cap F_i| / |F_i|$

F-Measure Rate= $\sum_{i=1}^{k} |F_i| . \varphi(F_i) / \sum_{i=1}^{k} |F_i|$

where $\varphi(F_i) = \max_{i=1}^{k} (2.A(D_c, F_i).R(D_c, F_i) / (A(D_c, F_i) + R(D_c, F_i))$

Clinical TREC XML document preprocessing

The most widely used application of text mining is in biomedical domain, due to a large number of medical document sets. In the traditional text mining approaches, the search process depends only on "sorting and motif" in the case of open access articles. Biomedical open access articles are extracted in full text and licensed under creative common license. Around 1GB of articles are detected in the first step of the document extraction algorithm. Unambiguous structured data are essential for the better performance of extraction model. Besides XML tags, generally, all the documents from the TREC medline repository have unstructured information. The document clustering approach is responsible for pre-processing and clustering of full-text articles. This approach emphasizes on abstract, unlike other text mining approaches in the biomedical domain.

In recent days, XML is used as a standard format for information sharing on the web. The most common and frequently implemented approach is clustering. Here, clustering represents merging of similar types of XML data & applications of XML clustering are: information retrieval, data integration, document ranking, web mining as well as query processing.

The major issues in XML data preprocessing for ranking are given below[2] :

- Initially, the clustering process calculates the similarity index among numbers of different XML data. But, this is a major problem to evaluate the similarity function because of the heterogeneity property of XML documents.
- Implicit dimensionality has increased to a great extent.

Biomedical documents, phrases, sentences are used in the feature extraction to extract the main features of the original documents. The graph-based feature extraction generates the features by extracting phrases or sentences from the set of key peer nodes of the overlay network. Finally, key phrases or sentences are extracted by computing the ranking scores and then selecting the highest scored phrases or sentences.

Table 1: Data preprocessing results on TREC clinical documents.

#Documents	#MeSH	#Feature terms	#noise symbols	#non-functional characters	#Filtered documents
1Lakh	334824	283784	23433	14546	92744
2Lakh	456733	485937	37365	24343	188956
3Lakh	637496	598768	46374	47265	267690
4Lakh	967386	698366	59783	64534	359885
5Lakh	1247836	893771	78653	97366	43859
1Million	3153353	2453645	183253	252573	74235

Table 2: Runtime Comparison of Proposed Model with the Traditional document cluster based classification models.

Models	Medline 100000(documents)	Medline 200000(documents)	Medline 300000(documents)	Medline 500000(documents)
LDA Model	64.75	119.3	162.65	198.24
SVM+K means	55.36	112.93	158.24	184.65
Proposed Model	33.26	104.65	142.93	179.24

Table 1 describes the runtime of the proposed model with the existing ranking models on TREC clinical data . From the Table 2, it can be observed that the proposed model has less average runtime for search operation when compared to existing models.

$$Liklihood = \frac{N_C}{N_C + N_{IC}}$$

In the above formula N_C is TREC reports that predict correctly and N_{IC} is TREC reports that predict incorrectly. Precision shows the number of the files that predict correctly over the number of files that is recommended by our method. Recall shows the number of files that predicted correctly over the TREC repository. We donate the F_B set of fixed file to and F_R the number of recommended file for precision and recall(accuracy).

$$precision = \frac{|F_B \cap F_R|}{|F_R|}$$

$$Recall = \frac{|F_B \cap F_R|}{|F_B|}$$

5. Conclusion

As the size of the text documents in the TREC repository increases, it becomes difficult to process the feature extraction and document representation in large corpus data. Also, the sparsity and finding the essential feature vectors in large clinical databases is important for decision making systems. Traditional latent Semantic Indexing and document clustering models are failed to find the topic relevance on large number of TREC clinical text document sets due to computational memory and time. Proposed document feature selection and clustered based classification model is applied on TREC clinical benchmark datasets. From the experimental results, it is proved that the proposed model is efficient than the existing models in terms of computational memory, accuracy and error rate are concerned.

References

[1] M. Rojcek, "System for Fuzzy Document Clusterng and Fast Fuzzy Classification", "15th IEEE International Symposium on Computational Intelligence and Informatics ", pp.39-42, 2014.

[2] A. Aïtelhadj, M. Boughanem, M. Mezghiche and F. Souam, "Using structural similarity for clustering XML documents", pp.109-139, 2011.

[3] S. W. Chan and M. W Chong, "Unsupervised clustering for nontextual web document classification", "Decision Support Systems", pp.377-396, 2004.

[4] D. Curtis, V. Kubushyn, E. A. Yfantis and M. Rogers, "A Hierarchical Feature Decomposition Clustering Algorithm for Unsupervised Classification of Document Image Types", "Sixth International Conference on Machine Learning and Applications", pp.423-428, 2007.

[5] W. Dai, G. Xue, Qi. Yang and Y. Yu, "Co-clustering based Classification for Out-of-domain Documents", "Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM", pp.210-219, 2007.

[6] I. Diaz-Valenzuela, V. Loia, M. J. Martin-Bautista, S. Senatore and M. A. Vila, "Automatic constraints generation for semisupervised clustering: experiences with documents classification", "Soft Computing 20, no. 6 ", pp. 2329-2339, 2016.

[7] C. Hachenberg and T. Gottron, "Locality Sensitive Hashing for Scalable Structural Classification and Clustering of Web Documents", "Proceedings of the 22nd ACM international conference on Information & Knowledge Management. ACM", pp.359-363, 2013.

[8] S. Jiang, J. Lewris, M. Voltmer and H. Wang, "Integrating Rich Document Representations for Text Classification", "IEEE Systems and Information Engineering Design Conference (SIEDS '16)", pp.303-308, 2016.

[9] W. Ke, "Least Information Document Representation for Automated Text Classification", "roceedings of the American Society for Information Science and Technology 49.1", pp.1-10, 2012.

[10] B. Lin and T. Chen, "Genre Classification for Musical Documents Based on Extracted Melodic Patterns and Clustering", "Conference on Technologies and Applications of Artificial Intelligence", pp. 39-43, 2012.

[11] L. N. Nam and H. B. Quoc, "A Combined Approach for Filter Feature Selection in Document Classification", "IEEE 27th International Conference on Tools with Artificial Intelligence ", pp.317-324, 2015.

[12] S. Shruti and L. Shalini, "Sentence Clustering in Text Document Using Fuzzy Clustering Algorithm", "International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)", pp.1473-1476, 2014.

[13] http://www.trec-cds.org/2017.html