Analysis of Similarity Metrics Through Clustering using WordCount,TF_IDF and Probability

Aranga Arivarasan, Dr.M.Karthikeyan

[1] Aranga Arivarasan, with Division of Computer and Information Science, Faculty of Science, Annamalai University of India.

[2] Dr.M.Karthikeyan with Division of Computer and Information Science, Faculty of Science, Annamalai University of India.

[3]

Abstract – **The text documents are very important in the usage of www. Many users require so much text information to gather essential knowledge in their required field of interest. To serve the user internet the quite relevant required topic related documents are to be retrieved. To satisfy this requirement as well as indexing and retrieving the documents the researchers tend to produce many new algorithms in the field of text document mining. The proposed system carries out the problem by performing two different operations. Fist one is the feature extraction operation. The second on is the clustering operation. In the first operation to extract the features from the text document various operations like preprocessing, tokenization, Stop word removal, streaming and bag of Words were performed. By performing these operations the Document representing features namely WordCount, TF_IDF and probability of word occurrence were determined. The second operation is performed with K-means clustering algorithm. In the clustering phase the three features and some of the similarity measures were used to conclude the overall performance. The proposed method yields better performance for Spearman Similarity compared with other two Cosine Similarity and Correlation Similarity metrics.**

Keywords – ***TFIDF, Word Frequency, Probability, pre-processing, Clustering, K-Means***

I. INTRODUCTION

The task of categorizing electronic document automatically in to their corresponding category is the main purpose of Document clustering. The fast increased internet usage leads to handling of enormous terabyte of electronic documents. Since the www efficient usage for several decades made text document classification very wide spread as well as implementation in numerous application like web mail spam filtering, web user emotion analysis, customer commodity searching requirements etc.

Text clustering is executed by representing the as a set of terms of indexes associated with some numerical weights. The goal is always to cluster the given text documents, in a way that they get clustered by means of the similarity measures with certain accuracy. There are many approaches are available for classification of text documents Naïve bayes, Support Vector Machines, DBSCAN, K-medoids, k-means and expectation maximization. The performances of the above said algorithms highly rely on the datasets provided to them for training. Before going to execute the text clustering the document representation approaches suffix tree representation of document analysis of similarity or distance metrics and most importantly the correct clustering approach are to be considered very carefully.

In some cases Clustering is wrongly referred as automatic classification. Because the clusters found are not known prior to processing. The distribution and the nature of data will determine the cluster members. But in classification classes are always pre-defined as well the classifier learns the relationship between objects and classes from trainingset. The trainingset is nothing but a set of data correctly labeled by human, and then used to the learning behavior of an unlabeled data. Many decades of study is going on document clustering but still it is far from consideration in solving problems. The most promising challenges lie in selection of features that should be used for clustering. Suitable similarity measure to perform the clustering operation, appropriate clustering method for utilizing selected similarity measure, implementing the clustering algorithm in an efficient way to make the clustering feasible will assist to achieve the quality of clustering performance. To solve these issues there are several clustering techniques are available namely Distribution based methods**,** Centroid based methods**,** and Connectivity based methods Density Models and Subspace clustering

The rest of the paper is organized as seven sections. In Section 2, the referred related works regarding the Document clustering is elaborately. The Section 3 describes the various distance metrics used in this paper. In Section 4, the system overview is elaborated briefly. Section 5, evaluates the experimental results in detail and section 6 produces the conclusion of the paper and section 7 gives the references made.

II. SIMILARITY METRICS

In document clustering, similarity is usually determined using associations and commonalities between features, where features are in general words and phrases. Two documents are considered as comparable if they contribute to similar topic or knowledge. When clustering is in employment on documents we are very much involved in clustering the component documents according to the type of information that is present in the documents. Accurate clustering requires a clear-cut description of the distance between a pair off substance in terms of either the pair wise resemblance or distance. A variety of similarity or distance measures have been proposed and broadly applied such as Spearman similarity correlation similarity cosine similarity Jaeeard coefficient Euclidean distance and so on.

A. Spearman Similarity

When documents are represented as term vectors the similarity of two documents correspond to the correspondence between the vectors. Spearman Correlation determines the correlation between two sequences of values. The two sequences are ranked individually and the difference in rank are calculated at every position, *i*. The distance between sequences
X = (***X1***, ***X2***, etc.) and ***Y*** = (***Y1***, ***Y2***, etc.) is computed using the following formula:

$$1 - \frac{6 \sum_{=1} (\quad -)^2}{(^2 - \quad 1)}$$

Where, ***Xi*** and ***Yi*** are ith values within sequence ***X*** and ***Y*** respectively. The range of Spearman Correlation is from -1 to 1. Spearman Correlation can sense certain linear and non-linear correlations.

B. Cosine Similarity

The most universally used measure in Document Clustering is Cosine Similarity. For any two documents d_i and d_j the relationship can be calculated

$$\cos(\ ,\ ) = \frac{\ .\ }{\| \ \| \| \ \|}$$

where, d_i, and d_j are m-dimensional vectors over the term set T= $\{t_i, t2, \ldots t_m\}$. All dimensions represent a term -with its influence in the document, which is non negative. As a result the cosine similarity is non-negative and restricted between [0, 1]. The cosine similarity is autonomous of document length. When the document vectors are of unit length the above equation is simplified to:

$\cos(d_i, d_j) = di.\ d_j$

When the cosine value is 1 the two documents are identical and if not anything in common between them then 0. Since document vectors are orthogonal to each other.

C. Correlation Similarity

Correlation is a procedure for investigate the association between two quantitative continuous variables. There are different forms of Pearson Correlation Coefficient formula is given by

$$, = \frac{\ }{[\quad ^2 \quad - \quad ^2 | \quad ^2 - \quad ^2 \quad]}$$

Where=and TF1 =

The measure ranges from +1 to -1. Positive correlation indicate that both variables increase or decrease together where as negative correlation indicates that as one variable increases so the other
decrease and vice versa. Two documents are matching when Pearson similarity is ±1. The spearman distance is a distance measure while the cosine similarity and Pearson coefficient are similarity measures. We apply a simple renovation to alter the similarity measure to distance values. Because cosine similarity bounded in [0, 1] and monotonic we take D = 1- SIM as the related distance value. For Pearson coefficient which ranges from -1 to +1, we take D = 1 - SIM when SIM >0 and D = | SIM | when SIM <0.

III. SYSTEM OVERVIEW AND METHODOLOGY

A. Preprocessing

The cluster process depends on different preprocessing technique to achieve best possible excellence and performance. Here we talk about some of the familiar preprocessing methods. The main intention of preprocessing is to symbolize the data in a form that make the most of it for clustering.

Quite a few traditions of representing the documents are Vector-Model graphical model TFIDF Probability keyword word count etc. Weighing of the documents and their similarities are calculated by implementing a variety of techniques. The significance of a word within a given document is typically represented in a vector illustration where for each word a statistical significance is stored. The text mining approach extremely relies on set of words a bag-of-words that a text document can be proficiently represented. The text processing phase involves a subsequence to reading textual documents and divides the characteristic into tokens, words, terms, or attributes. The weight obtained from the occurrence of the terms of each text document followed by the removal of uninformative attributes such as stop words numbers and special characters. The rest of the characteristics are then unvarying by dropping to the root through the error rectification process. Despite removing uninformative features the size of a text document space may be too large. Certain constraints to decrease the size of the character space of every document as input in addition with the regularity of the

characteristic of each document. The purpose of this phase is to improve the quality of features extracted to represent the document and simultaneously reducing complexity of mining procedure.

B. Tokenization

Tokenization in our common sense not only divides the tokens but also interprets and groups individual tokens to generate higher rank of understanding. Tokenization converts a stream of characters into a progression of tokens. A token is an illustration of a sequence of characters in some specific document that are grouped simultaneously as a constructive semantic unit for processing. A category is the classes of all tokens contain the same character succession. The data must be processed through all three operations. The first operation is to renovate the documents into the quantity of words one and the same to BOW. The second operation is to eliminate a blank succession that is known as cleaning and filtering. At last every text input document is split into a directory of characteristics also called token, term, vocabulary, or attribute.

C. Stop words

Stop words is a listing of normally frequent tokens which appear in each text document. The frequent tokens such as conjunctions and pronouns need to be removed because it does not have any effect in the form of tokens and these words include a very modest or uninformative on the categorization process of a document representation. Some exceptionally universal words that would emerge to be of slight value in serving document matching the user needs are barred from the vocabulary entirely. For the same reason if the tokens are special character or number then those tokens must be eliminated. In order to discover the stop words we can arrange our list of terms by regularity and pick the high frequent ones according to their lack of semantics importance.

D. Stemming

Stemming is the practice of eliminating prefixes and suffixes as of tokens. The practice is conceded out for dropping to the resultant word of the stream. The stream need not be identical to the original morphological root of the word and it is usually satisfactorily related from beginning to the end of word map similar to the stream. This progression is used to minimize the count of tokens in the feature space and advance the performance of the clustering when variety of forms of features is stemmed into a distinct feature. The streaming process is carried using the following algorithm

Step 1: Eliminate plurals (-s) and suffixes (-ed or - ing).
2: If the vowel occurs in the previous step, replace y to i on the next word.
Step 3: From the step 3, Map double suffixes to single ones (-ization,-ational).
Step 4: Additionally, reduces the suffixes like (-full, -ness) etc.
5: Deducts (-ant, -ence) etc.
Step 6: If a word ends with a grammatical verb ending, then it has been removed.
Step 7: Finally, removes a (-e).

E. Word frequency count

An important set of metrics in text mining relates to the frequency of word count (or any token) in a certain corpus of text documents. However, one can also use an additional set of metrics in cases where each document has an associated numeric value describing a certain attribute of the document. One will first go through the process of creating a simple function that calculates and compares the absolute and weighted occurrence of words in a corpus of documents. This can sometimes uncover hidden trends and aggregates that aren't necessarily clear by looking at the top ten or so values. They can often be different from the absolute word frequency as well. Then It is simple to do the basic analysis and find out that your words are split 50:50 to measure the absolute frequency of words, and try to infer certain relationships. In this case, you have some data about each of the documents. The key word exists: in which case the assignment is done (adding one). Now the key exists, its value is zero, and it is ready to get assigned an additional 1 to its value.

Although the top word was in the first table, after counting all the words within each document we can see that other words are tied for the first position. This is important in uncovering hidden trends, especially when the list of documents you are dealing with, is in the tens, or hundreds, of thousands. With counted occurrences of each word in the corpus of documents, the weighted frequency can be obtained. This reflects how many times the words appeared to readers; compared to how many times used them.

F. Bag of Words (BOW)

BOW is simplified version used in data mining for information retrieval and document clustering. Bag of Word is a simplest method for feature identification and representation of text document. BOW process consists of the following steps,

Step 1: Every document is indexed by means of the bag of vocabulary by a vector with one document for each term taking place in the whole gathering of tokens in document. Each vector has a corresponding value representing the count of appearance of the term that appeared in the document.

2: All documents are represented as a point in a vector space with one entry for every term in the vocabulary.

3: If a word do not come out in a feature document that particular vector is set to the importance nil.

G. TF—IDF

Feature selection is an essential process in document clustering to produce better accuracy, efficiency, and scalability of a text documents, compared to other techniques. Several procedures are available to group the text documents namely information gain, mutual information, term Frequency, Chi-square process, cross entropy, the term weighting methods of text, index based process.

Enhanced TF-IDF is used for dimensionality reduction. The feature selection and weighting methods contain the following steps:

In Term Frequency Inverse Document Frequency the term weights are set as the uncomplicated numeral of occurrence of the terms in the documents. This clearly shows the ability of accepting that the terms occurring frequently within a document may identify its meaning more robustly than terms occurring not as much of frequently and should be given higher weights. Each document d is identified as a vector in the term-space and represented by the term frequency (TF) vector value.

The document vector "d" is represented by,

D= {Term X $freq_1$Term X $Freq_2$ … … Term X $Freq_n$}

Where i= {1, 2 . . . , n} is the term frequency for whole documents. Depending on the Vector Space Model, the weight matrix is calculated by using the matrix derivation.

Term Frequency (TF) is a score of the occurrence of the word in the existing document. Given that all documents are dissimilar in span it is possible that a term would appear many times in lengthy document than shorter ones. The term frequency is often determined by the document length to standardize.

=

To give a privileged influence to words that occurs merely in a little document the words that occur frequently across the entire collection are not helpful. Terms that are restricted to few documents are useful to differentiate those documents from the remaining of the collection. The inverse document frequency term weight is one way of conveying higher privilege to more discriminative words. IDF is defined by fraction N/ni, where N is the total number of documents in the collection and the number of documents in which term i occurs. Due to the large quantity of documents in many collections this appraise is usually compressed with a log function. The resultant description of IDF is thus:

Among these methods, Information Gain, TF, DF and $_{log}$ IDF, Chi-square (Statistical term and entropy based), term weighting methods *TSW* and TDW are useful methods to manage the feature selection process. The

combined term frequency with IDF results are determined as TF-IDF weighting.

,= , 1

The TF-IDF version of the Document d is

–

= 1 log , 2 log , , … , log ,

1 2

Normalized unit vector to all document vector is

| – | = 1

Centroid vector is

= 1

|| ∈

Inverse Document Frequency (IDF) is a score of how exceptional the word is across documents. IDF is a quantity of how extraordinary a term is rarer the term more is the IDF score.

= ( )

The Term Frequency (TF) for each word is normalized by the inverse document frequency that helps to find TF/DF of the documents. TF/DF describes coordinates of the term weights that are given by the term frequencies as well as to calculate Cosine similarity for all vector space models. The cosine similarity equation is as

, , ,

, = 2 . 2

, ,

Q – Query of term frequency
I - Inverse document frequency
W – Weight
J – Term frequency
D – Document vector
Thus,

– = *

From the above formula If a document containing 100 words wherein the word *India* appears 5 times. The term frequency for *India* is (5/
100) = 0.05. Now, assume we have 10 million documents and the word *India* appears in one

thousand of these. Then, the inverse document frequency is calculated as log(10,000,000 / 1,000) = 4. Now, the Tf-idf weight is the product of these determined values: 0.05 * 4 = 0.20.

H. Probability

The important contribution in this proposed method is to find the probability distribution of similar documents to perform the clustering process. The proposed method determines a unique probability distribution equation to achieve the most
significant accurate clustering process. In the document clustering phase each document is selected from the dataset and by using the probability distribution function the corresponding probability of each unique word in that document is calculated for the purpose of clustering. For each word in the document the relationship between the selected document and the corresponding cluster is determined. Depending on the probability values the document which has the overall maximum probability value is assigned to that cluster.

, = *)

, = _____

, = _____

IV. EXPERIMENTS AND RESULTS

For our experimental purpose the proposed system collected 300 documents for the five categories Business, Entertainment, Politics, Sports and Technology. It is very important to perform the very effective preprocessing task to generate the unique words. The outcome of our proposed system to generate the unique words were shown in Table1.

TABLE 1. TOTAL WORD AND UNIQUE WORD COUNT

Category	Total Number Of Unique Words	Total Number of words
Business	6544	60786
Entertainment	8929	64622
Politics	7422	76277
Sports	6919	59930
Technology	8207	87550

Initially the proposed system splits the entire documents in the corpus in to individual tokens. Then all tokens are calculated to find the keyword occurrence. With table. 1. Results the WordCount, TF_IDF frequency and probability of keyword occurrence were calculated. The determined values are formed in to clusters by means of K-Means clustering algorithm. To perform the clustering the K-means uses three Similarity metrics Spearman Similarity, Cosine Similarity and the Correlation Similarity. From the clusters the confusion matrix is determined. The confusion matrix is used to find the accuracy, precision, recall and F-measures.

The precision, Recall and F-measures were calculated by using the following formulas

$$= \frac{}{+} \quad \frac{}{*}$$

$1 = 2$

The calculated values are shown in Table 2. The fig.1. Graphically represent the accuracy of the three similarity metrics for the WordCount, TF_IDF and the Probability values. It clearly shows that the clustering operation gives better results by using the

Probability values for all the three similarity metrics. Among the similarity metrics the Spearman similarity gives better results than the other two

=

Cosine Similarity and Correlation Similarity for all three WordCount, TF_IDF and Probability value.

	Distance Measures	Spearman Similarity	Cosine Similarity	Correlation Similarity
WordCount	Accuracy	94.46	76.87	72.33
	Precision	9.20	7.81	6.44
	Recall	9.23	7.50	6.76
	F-Measures	9.21	7.65	6.60
TF_IDF	Accuracy	94.80	80.80	74.40
	Precision	9.37	8.50	6.59
	Recall	9.43	8.53	6.76
	F-Measures	9.40	8.19	6.67
Probability	Accuracy	95.7	88.4	86.8
	Precision	9.45	8.84	8.77
	Recall	9.33	8.66	8.33
	F-Measures	9.39	8.75	8.54

TABLE 2. RESULTS USING WORDCOUNT TFIDF and Probability of words

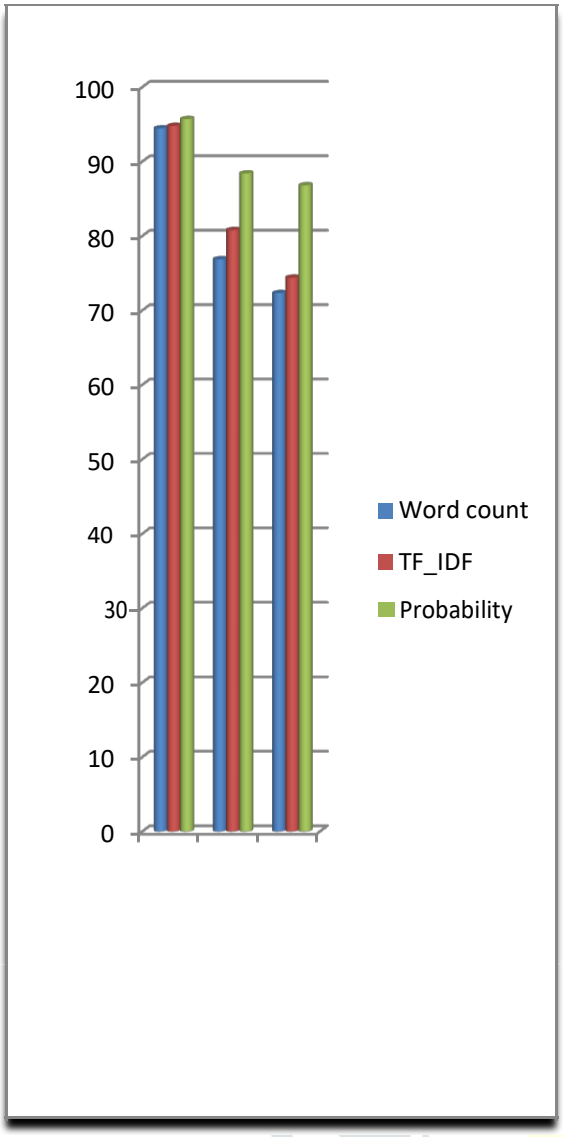

Fig.1. Bar chart showing the Accuracy

For our better understanding and demonstration purpose all the calculated results were shown in Fig.1. Our proposed system calculates Accuracy, Precision, Recall and F-Measures by using the WordCount, TF_IDF and Probability values through K-Means clustering. The clusters are determined by using the three similarity metrics Spearman Similarity, Cosine Similarity and the Correlation similarity.

From the Fig.2. we can clearly understand that the Spearman Similarity measures achieves the best overall performance than the other two similarity measures.

Fig.3. describes the line chart of Precision, Recall and F-Measures determined by our proposed system. Precision, Recall and F-Measure are the external measures to analysis the performance of the system. Precision retrieves the number of correct assignments out of the number of total assignments made by the system.

Recall retrieves the number of correct assignments made by the system, out of the number of all possible assignments. F-measure is a combination of the precision and recall measures used in system. From Fig.3 shows our system yields better performance.

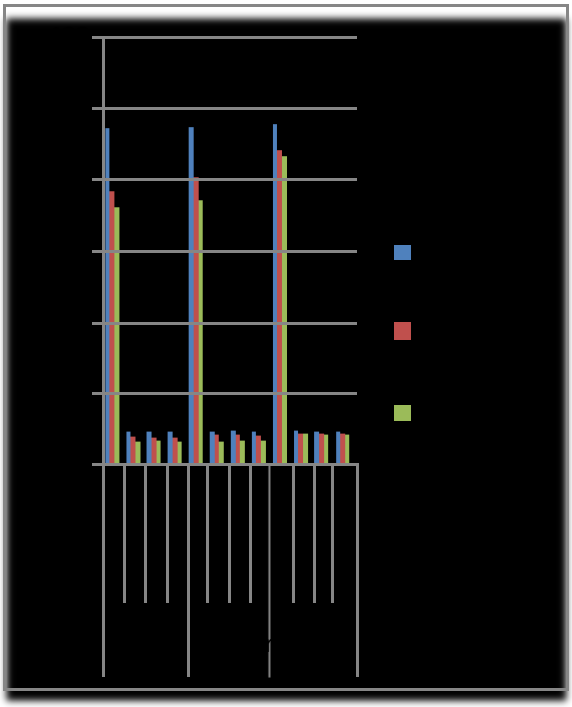

Fig.2. Accuracy, Precision, Recall and F-Measures.

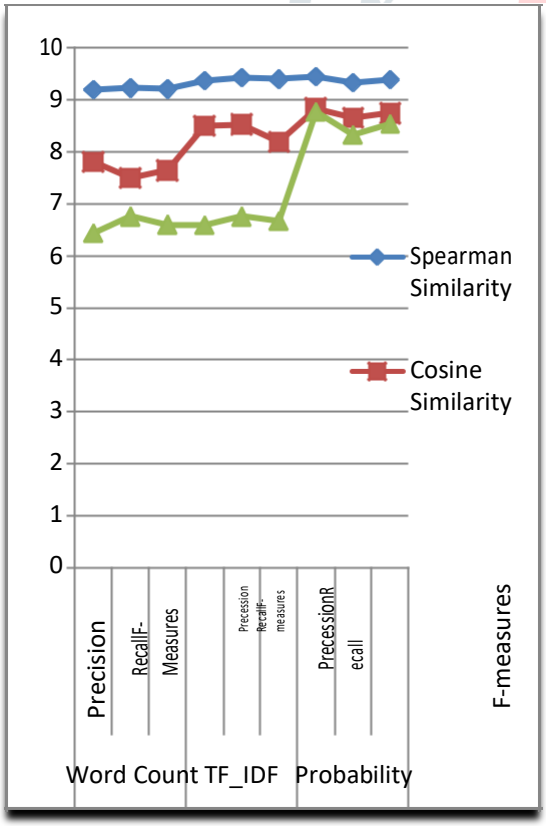

Fig.3. Performance Analysis chart.

V. Conclusion

In this paper the proposed clustering approach is implemented using the WordCount , TF-IDF and Probability of word occurrence features of the documents of five different categories. For each category we have taken 300 documents. The proposed system retrieves 6544 key words out of 60786 unique words for the business category. The Entertainment category documents retrieve 8929 key words out of 64622 unique words. The Politics category retrieves 7422 key words out of 59930 unique words. The sports category retrieves 6919 key words out of 59930 unique words. The Technology category retrieves 8207 key words out of 87550 unique words. With these results the K-Means algorithm is adapted to perform the clustering. The proposed model uses three similarity measures to compute the clustering operation. The Spearman Similarity measure yields an accuracy of 94.46, 94.80, 95.70 for WordCount,TF-IDF and Probability respectively. Cosine Similarity measure yields an accuracy of 76.87, 80.80, 88.40 for WordCount,TF-IDF and Probability respectively. Correlation Similarity measure yields an accuracy of 72.33, 74.40, 86.80 for WordCount, TF-IDF and Probability respectively. The proposed method yields better results for Spearman Similarity compared with other two Cosine Similarity and Correlation Similarity metrics. In future the model may be extended by increasing the number of categories as well as the number of documents also. The extracted features used hearer can also be examined by other clustering algorithms like DBSCAN, KNN, ANN and GMM so on.

REFERENCES

[1] Saqib Alam, Nianmin Yao, "Big Data Analytics, Text Mining and Modern EnglishLanguage" journal of Grid Computing 2018

[2] Vladimer B. Kobayashi1, Stefan T. Mol1, Hannah A. Berkers1, Ga´bor Kismiho´k1and Deanne N. Den Hartog

[3] Robert Wing Pong Luk, Kam-Fai Wong, Kui-Lam Kwok "Interpreting TF-IDF Term Weights as Making Relevance Decisions", ACM Transactions on Information Systems, Vol. 26(3) 2008.

[4] Dibyendu Mondal Pushpak, Raksha Sharma, "Comparison Among Significance Tests and Other Feature Building Methods for Sentiment Analysis: A First Study", International Conference on Computational Linguistics and Intelligent Text Processing, pp 3-19, 2017.

[5] Kasula Chaithanya Pramodh, Dr.P.Vijayapal Reddy, "A Novel approach for Document Clustering using concept extraction", International Journal of Innovative Research in Advanced Engineering, 05(3),pp 59-65, 2016.

[6] Charu C. Aggarwal, ChengXiang Zhai. "A survey of text classification algorithms", Mining text data. pp. 163–222, (2012)

[7] Borovikov, E. "A survey of modern optical character recognition techniques", Computer Vision and Pattern Recognition (2014).

[8] Bsoul, Q., Salim, J., Zakaria, L. Q. "An intelligent document clustering approach to detect crime patterns", Procedia Technology, 11, pp.1181–1187, 2013.

[9] Cohen Priva, U., Austerweil, J. L., "Analyzing the history of cognition using topic models", Cognition, 135, pp.4–9, 2015.

[10] Aranzabe, M. J., A. D. de Ilarraza & I. Gonzalez-Dios . "TransformingComplex Sentences using Dependency Trees for Automatic Text Simplificationin Basque", SEPLN, pp. 61–68. 2012

[11] Matthew Honnibal and Ines Montani. spacy " Natural language understanding with bloom embeddings", convolutional neural networks and incremental parsing. 2017

[12] Sowmya Vajjalla and Detmar Meurers "Readability assessment for text simplification: From analysing documents to identifying sentential simplifications", International Journal of Applied Linguistics, 165(2):194– 222, 2015.

Aranga Arivarasan is a Research Scholar who is working as Assistant Professor in Division of Computer and Information Science, Annamalai University, India. He completed his B.Sc[Computer Science] and M.Sc[Computer Science] From Madras university in 1998 and 2000 respectively, the M.B.A and M.Phil[Computer Science] from Annamalai University in 2005 and 2007 respectively.

Dr.M.Karthikeyan is an assistant professor in Division of Computer and Information Science, Annamalai University, India. He completed his M.Sc[Computer Science] from Bharather University in 1993 and M.Phil[Computer Science] and Phd from Annamalai University in 2005 and 2014 respectively. His area of interest is Data Mining, Digital Image Processing, and Artificial Neural Networks.