Clustering Textual Documents by Extracting Sequence from Word-of-Graph

Murtaza Munawar Fazal[1], Muhammad Rafi[2]

[1]*Shaheed Zulfikar Ali Bhutto Institute of Science and Technology* (SZABIST) Karachi Pakistan

[2]*Faculty of Computer Science University, FAST – National Karachi, Pakistan*

[1]murtaza.fazal@gmail.com

[2]rafi.muhammad@gmail.com

Abstract: Document clustering is an unsupervised machine learning technique that organizes a large collection of documents into smaller, topic wise homogenous and meaningful sub-collections (clusters). Traditional document clustering approaches use extracted features like: word (term), phrases, sequences and topics from the documents as descriptors for clustering process. These features do not consider the relationship among different words that are used to convey the contextual information within the document. Recently, Graph-of-Word approach is introduced in information research; this approach addresses the problem of independent assumption by building a graph of word from the words that appeared in a document. Hence, the relationships among words are captured in the representation. It is an un-weighted directed graph whose vertices represent unique terms and edges represent co-occurrences between the terms. The representation is simplified by using a sliding window of size = 3 with the text of the document. This paper uses a sequence based-representation of document that is extracted from graph-of-word of the document. A similarity measure is defined over the common sequences between two documents. The proposed approach is implemented and tested on standard text mining datasets. A series of experiments reveal that the proposed approach outperforms the traditional approaches on clustering measures like: Purity, Entropy and F-Score.

***Keywords* ----Document Clustering, Information Retrieval, Unsupervised techniques, Data Mining and Document Graph**

I. INTRODUCTION

Clustering is an unsupervised data mining approach which is widely used in variety of situations. It combines a group of documents into meaningful sub-groups; the word meaningful is rather relative. A data clustering problem that focuses on objects that are in the form of documents is known as Document Clustering. The main concentration of this process is to group similar documents into a single group (cluster) which are identical in some aspects like type of document, contents of document and etc. The challenging parts are first to identify the relevant features for clustering and to identify how many classes of such groups (cluster) exist in the data set. Document Clustering attempts to find grouping among the documents in such a way that the documents belonging to a cluster are similar (i.e. high intra cluster similarity) and are different to documents which are part of other clusters (low inter cluster similarity). It is an unsupervised approach Identification and classification of unknown features of data in a document is highest priority of document clustering method. Traditionally, document clustering algorithms mainly uses features like: words, phrases, and sequences from the documents to perform cluster. Mostly, the feature extracting techniques used by these algorithms are based on frequency distribution of the features and feature counting to decide the similarity between a pair of documents. All these approaches thus, do not consider the context in which the text was used. These approaches purely perform clustering irrespective of the context. Human readable documents comprise of context and the usage of words highlights the context of the text. Recently, few researches have suggested some different document model representation which aims to captures the semantics of the words. Some of the worth mentioning approaches are frequent word sequences, frequent word-meaning sequence and representation of document as a graph-of-word. All these approaches have reported better results than the traditional approaches. In this paper, a new document clustering approach will be introduced that significantly depicts the context of text efficiently than the previous approaches. Document representation in terms of word-of-graph, where each unique word represents the edges and directed vertices among them, is an effective method which retains the sequence in which the words were used originally but comparing graphs is a cumbersome task. In this approach, word-sequences are extracted from this directed graph which not only reduces the amount of words used in the document representation but also the tedious job of comparison of documents is relaxed. As per facts, it cen be safely assumed that this is the very first attempt to represents document in word-sequences depicted from the word-of-graph. With this approach, it can be perceived that the context in which the text was written is better recognized among the various other unsupervised approaches. One of the vital features of this approach is to represent the document in a compact form, which eventually reduces the size of the document. Finally, the Hierarchical agglomerative clustering is used to perform the clustering. The results of the standard clustering measures produced in this study using the standard information retrieval datasets; clearly outperform the results of the other approaches under comparative study. The rest of the paper organized as follow, in the section three, the related works of this study will be discussed. Then the experimental setup, data set, approach to document clustering and the measures of this study will be discussed. Finally, in the last section, the results and conclusion of the work is discussed.

II. LITERATURE REVIEW

Data Clustering [1] is an effective unsupervised data mining technique used to discover knowledge within the data. Unsupervised approaches do not have any prior knowledge of classes to which the data may belong. Document clustering focus on data clustering problems which further focuses on objects which are in the form of documents. The aim of document clustering is to find relevance among the documents and group them together. Documents belonging to a cluster are linked together by some features (like words, meanings, etc.) and are dissimilar to other cluster of documents by the same feature set. The cumbersome part is to determine the similarity among the documents i.e. having higher intra cluster similarity and dissimilarity with other document clusters i.e. lower inter cluster similarity. Identifying the correctness of obtained feature set and grouping of documents without any prior knowledge is a major challenge of document clustering method. Clustering is an effective method for computing search [2]. It allows grouping similar results [3], discovering similarity among the documents [4]. Different clustering methods are presented in [1]. It has two major categories (i) Hierarchical vs. Flat and (ii) Partition vs. Overlapping.

Agglomerative hierarchical clustering [1] (AHC) is a bottom-up approach so it initially treats each document as a cluster, pair of documents are merged together in a cluster after performing similarity measure calculations [5]. It requires extensive amount of calculations for processing the similarity between all the documents. The other category of document clustering is partitioned based algorithms, which create a one level partitioning of the document collect as stated in [3, 6, 7]. Using an algorithm like k-means, it creates k-documents as base level documents for the first round of clustering process. Based on some similarity measure used, the documents that are similar with respect to some feature set will be merged together and the base level will be recalculated based on the result of the clustering process. This process is iterated until there cannot be more base level calculation possible. Steianbach *et al* states the difference between the two categories of document clustering that were mentioned [8]. Traditional document clustering approaches mainly extract features like word, phrases and sequences from the documents. [9-12]. It applies extraction techniques that are based on frequency distribution of features and feature counting to relate the documents. All of these approaches do not retain the context in which the text was written. Thus, it cannot guarantee the theme of the document.

Two new document clustering algorithms that claim to obtain document context better than the traditional approaches are Clustering based on Frequent Word Sequence (CFWS) and Clustering based on Frequent Word Meaning Sequence (CFWMS) [12]. Both of these approaches maintain a list of unique words that contains words that are frequently used in the documents. Let supposed a database D consist of 3 documents d1, d2 and d3. It can be written as D = d1, d2, d3. Each document contains distinct words and the database D has all the distinct words from all the documents. To obtain frequent words sequence, a 2-word sequence is generated among each document. To filter out the less frequent word sequences, minimum occurrence of a sequence is controlled by a threshold value which may be kept as 5% occurrence of a word sequence is required to be part of the document representation sequence list. After filtering out the unnecessary word sequences using the threshold value, the final dictionary can be obtained which can be now written as D' = { d1', d2', d3'}. This not only reduces the word sequences in D' but also improves the clustering among documents. In CFWS, the documents that support the same frequent word sequence are considered to be cluster candidates. The minimum threshold used is 5-15% word sequences. Documents are merged based on k-mismatch concept using Landau–Vishkin (LV) algorithm [13]. The same process is repeated to build all the clusters present in the database of documents. The second algorithm proposed in [12] is CFWMS which uses frequent word meaning sequence to obtain similarity among documents. A word may be used in different aspect and it can be depicted by the same lexical concept that a word form can use to express [14]. WordNet is used to convert word forms to the word meaning they express so that a word that is used in different forms, synonyms, etc. are all represented as one word and the word frequency is properly calculated. For instance different words like "car", "auto" and "vehicle" supports the count of one word meaning. The words that do not match with the entries in WordNet are kept unchanged as they may capture the uniqueness of the document.

Textual documents can be denoted in terms of a graph-of-word [15]. In the works of Mihalcea and Tarau [16] and Erkan and Radev [17], salient vertices of a graph are extracted from a sentence. Document context can be represented by these vertices. Therefore the vertices of the graph represent the unique terms and edges denote the co-occurrence between the terms or a meaningful relationship semantically [18]. Term may be a word [16,19] or even a sentence [16,17] that make up the vertices. As mentioned in [20], edges can be weighted or un-weighted, that is, in weighted graph, the co-occurrence of two terms can be counted and weight of the edges are labelled whereas in un-weighted graph, the frequency of repetitive two terms is not maintained. Further, graphs can be directed (ordered pairs of vertices) or un-directed (unordered pairs of vertices). The approach proposed in [15] uses the un-weighted directed graph as the un-weighted graphs led to a better results and directed graph was used to maintain the order in which the words were used. When creating a graph, a moveable window was used to select adjacent words to have an edge connected to the respective vertices. For instance, a document d1 contains a sentence "Karachi is the biggest city of Pakistan. Karachi has a very high population". Here, the original document is reduced to contain only distinct words and repetitive words are removed. Each distinct word is a vertex and all adjacent words that are within the sliding window have directed edge between them. The edges that are created from the vertex leading from "Karachi" for clarity purpose are shown only in figure 1.

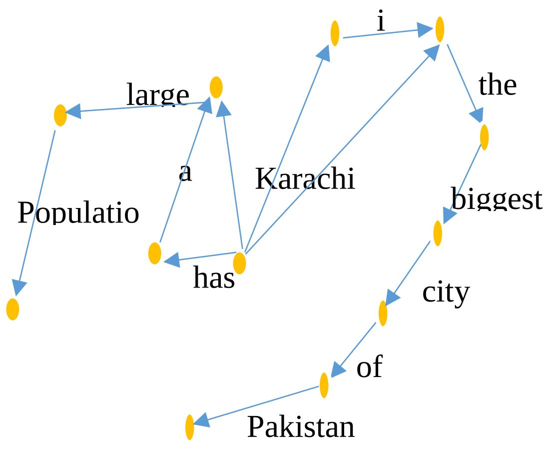

Fig. (1). Word Sequence from Word of Graph

To form clusters, the word-of-graphs of different documents are compared using the TF-IDF (term frequency–inverse document frequency) weighting model and documents with higher similar graph-of-word are merged into a cluster.

There have been tremendous amount of work on IR and different approaches have been proposed. Examples are vector space model (TF-IDF) [21], probabilistic (BM25) [22] and language modeling (Dirichlet prior) [23] approaches and the divergence from randomness framework (PL2) [24]. These methods represent the document in terms of bag-of-word or frequency based term weighting. A retrieval model could be defined as a function of a term weight (TW) and a document weight (DW) [15]. In this context, a new approach has been proposed using the graph-of-word concept but it utilizes the word-sequence document representation to produce effective clustering results.

The approach that has been proposed in this paper, extracts word-sequences from word-of-graph. A word-of-graph of a document is created using a window size of three, such that the following two words have an edge between them. Using this technique, a word-of-graph of an entire document is created similar to what was proposed in [15]. This document representation retains the context of text in which it was written. Using this graph, word-sequences dissimilar to what was proposed in [12] is extracted as it was used for generating word-sequence from the document itself, this resultant word-sequences generated from word-of-graph is the final representation of the document. To clear the working, consider the overly simplified example discussed earlier, if the document d1, is only represented by the word-sequence is generate from the document after stop-words removal and lemmatization, a document representation d containing sequences s1 = "Karachi,biggest" , s2="Karachi,city", s3 = "city, Pakistan" and so on would be obtained where d = {s1 + s2 + s3, …}. These word sequences are the features present in the document which are then compared with other documents to form clusters. This approach is different from the word-of-graph as in that approach the word-of-graphs of different documents were compared whereas in the proposed approach the word-sequences are compared. Furthermore, this approach is different from the one proposed in [12] as this approach does not consider the frequency of words. Each word is used uniquely but it may be connected to other words through edges between them. Hence it is assumed that this approach would be closer to the true representation of the document semantically which would generate better word sequences and result in better document clustering.

III. EXPERIMENTAL SETUP

In this section, the paper evaluates the performance of the new suggested approach against the other approaches that have been discussed in the previous section. The algorithm was implemented on C# 4.0 and experiments were executed on Windows 8.1 based standard PC. For the creation of graph, Quick Graph library was used.

A. Datasets

The dataset used is from the TREC 9-10 Web collections of documents. The Text Retrieval Conference (TREC), co-supported by the National Institute of Standards and Technology (NIST) and U.S. Branch of Defence, was begun in 1992 as a feature of the TIPSTER Text program. Its objective was to facilitate research within informational retrieval community by providing base to extensive scale assessment of text retrieval methodologies. Four datasets has been created from the TREC collection by selecting random documents of sizes 50, 100, 200 and 400 respectively. All the documents present in the selected datasets are pre-processed before use. Stop words are removed and each word is stemmed using Porter's Suffix Stripping algorithm and words are lemmatized using Morpha-Stemmer.

B. Algorithm

In proposed approach, the documents are firstly parsed to convert them in a generalized format which is understandable by application. This approach helps to cater different types of document formats. The documents are then passed to a pre-processing module which initially removes all the stop words in the document. For stop words removal, the Onix Textual Retrieval Toolkit stop-word list 1 and 2 were used. The next step is to convert the words to the root-form such as the word "better " and "good" meaning same but are in different forms and converting the words to the first form will help to identify the context behind the document. This refined document still contains derived words like "moved" and be replaced with "move" therefore the document is processed through word stemming module which further refines the document.

1- Creating Word-Of-Graph: The document is represented as a graph-of-word that related to an un-weighted directed graph in which vertices represent unique words and whose edges represent the relation between the words within the moveable window size. The direction of edges represents the sequence in which the words were used. The fundamental supposition is that all the words present in the document have associations with the others within the window size, outside of which the relationship is not taken into consideration. This approach connects all co-occurring terms together without taking into account their meanings [15].

For displaying the resultant graph which will be created in this step, definition of IR has been taken from the Wikipedia which is, “Information retrieval is the activity of obtaining information resources relevant to an information need from a collection of information resources”. The text is broken into words and converted into lowercase and parsed to the graph library. The graph will contain only unique words present in the content and in case there are repeating words then the same vertex will have edges to the subsequent two words for each instance (with window size set to 3). Figure 2 shows the resulting un-weighted directed graph.

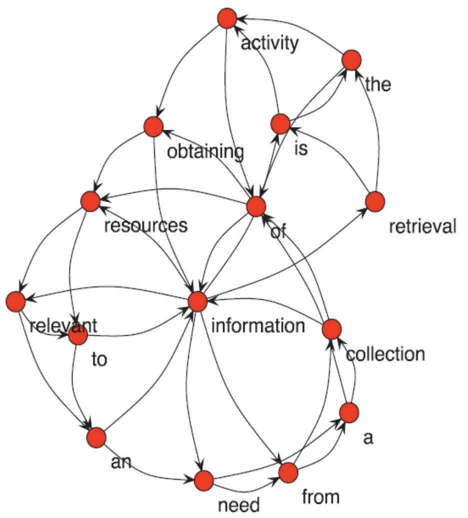

Fig. (2). Graph of Word [15]

2- Extracting Sequences**:** An ordered sequence of two or more words is called a word sequence [25]. A word sequence S is represented as (w1, w2...). There could be words between them which are removed in the pre-processing phase. A text document d supports this word sequence if these four words (w1, w2, w3, and w4) appear in d in the specified order. Multiple occurrences of a sequence in the same document are counted as one [12]. Hence, in this case the document treated as input to the word-sequence is the graph-of-word. Two word pairs have been used as a sequence in this approach. For clarity purpose, the previous example will be discussed. The algorithm starts with the first word presented in the graph and traverse the entire graph word by word and extract the two adjacent words that have an edge between them. So the obtained sequences would be w1 = {information, retrieval}, w2 = {information, resources}, w3 = {information, relevant}, w4 = {collection, information}, etc. The list of entire sequences present in the document is the final representation of the document and they maintain the correlation between the terms along with the context of the text.

To cluster the word-sequences extracted from word-of-graph, hierarchical clustering on the candidate clusters to obtain the final result was performed. Documents that have similar word-sequences extracted from graph-of-word display similarity and are merged together into one cluster. The same process is repeated until all the documents belonging to at least one cluster have been received. Figure 3 shows all the steps involved in the proposed document clustering approach.

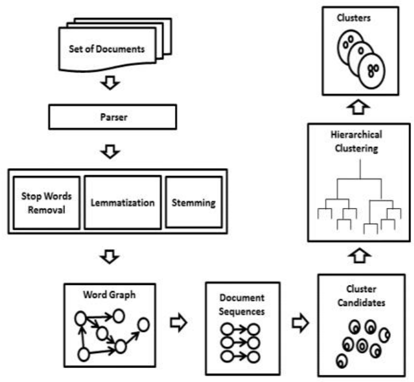

Fig. (3). Process of WSOBW

C. Measure

1. **Document Similarity:** Documents that have similar contextual features should be placed inside same cluster and those having dissimilarity should be part of a different cluster. To find the similarity following formula is used:

$$\text{Similarity} = \frac{d1 \cup d2}{d1 \cap d2}$$

Where d1 and d2 are documents which belong to the dataset D. Documents that have more sequences in common would have higher similarity measure.

2. **F-Score:** The f-measure utilizes a mix of precision and recall values of clusters. The number of documents is denoted as η_x in class x and the number of documents in cluster y as c_y. Hence c_{xy} would represent the number of items of class x which belongs to cluster y. So, the precision can be defined of cluster y with respect to class x as prec(x, y). Therefore, the equation can be written as prec (x,y) = $\frac{c_{xy}}{c_y}$

and recall of cluster y with respect to class x as $c(x,y)=\frac{C_{xy}}{C_y}$.

Therefore. the f-measure can be written as:

$$F(x,y)\frac{2*prec(x,y)*rec(x,y)}{prec(x,y)+rec(x,y)}$$

And the f-measure for the entire cluster can be written as:

$$\sum_x \frac{\eta_x}{\eta}\max(F(x,y))$$

3. Purity: It is defined as the maximum precision value for every class of y. Purity is calculated as following:

$$Purity=\sum_y \frac{C_y}{N} purity(f)$$

Where, N represents the sum of the cardinalities of each cluster. Hence this is used as the quantity instead of size of document.

4. Entropy: It is the measure that how similar each cluster y is. It is written as:

$$Ex=-\sum_{y\varepsilon L} prec(x,y)*\log(prec(x,y))$$

And the total entropy for a collection of clusters is calculated as:

$$Entropy_c=\sum_{x\varepsilon C}((\frac{N_x}{N})*E_x)$$

Minimum entropy and maximum purity values results in better clustering.

IV. RESULTS

Following are the results obtained from different statistics applied on NSTC, Bag of Words (BOW), Clustering based on Frequent Word Meaning Sequences (CFWMS) and approach of Word-Sequences from Bag of Words (WSFBW). These results are based on 4 dataset d1, d2, d3, d4 samples which vary in limit 50,100,200 and 400 respectively.

A. Generated Clusters

Following table 1 shows the number of clusters generated by each of the algorithm against different datasets:

Table 1. Number of Clusters

Algorithm	Number of Documents	Clusters	Expected Clusters
NSTC	50	9	5
	100	12	9
	200	17	13
	400	24	21
GOW	50	7	5
	100	11	9
	200	12	13
	400	17	21
CFWMS	50	2	5
	100	6	9
	200	10	13
	400	15	21
WSFGW	50	4	5
	100	7	9
	200	15	13
	400	18	21

With the results obtained, it can be seen that the results obtained from NSTC degrade with the number of documents increased whereas the other algorithms are near to the expected cluster's range. WSFGW performs better than the rest of the algorithm on the selected dataset.

B. F-Score

Following is the F-Score result plotted on the graph in figure 4:

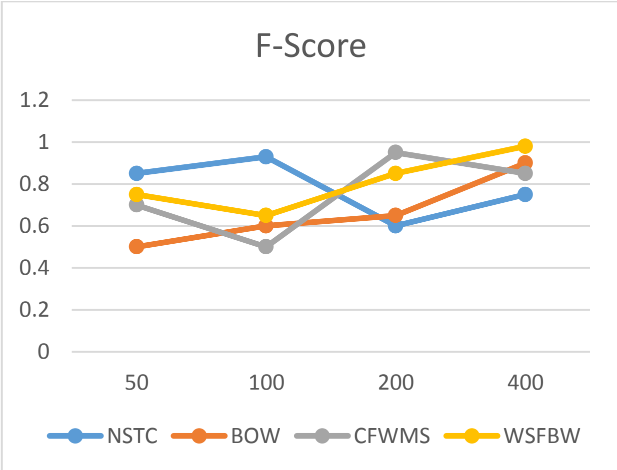

Fig. (4). F-Scores

The graph shows that the WSFBW performs better than the BOW approach when testing with 400 documents whereas CFWMS performs better with 100 documents under observation.

C. Purity

Following is the graph in figure 5 plotted for Purity of clusters and it can be seen that the results of BOW, CFWMS and WSFBW are almost leading to the same point but again it can be seen that the WSFBW is capturing the context behind the text and clustering the documents efficiently.

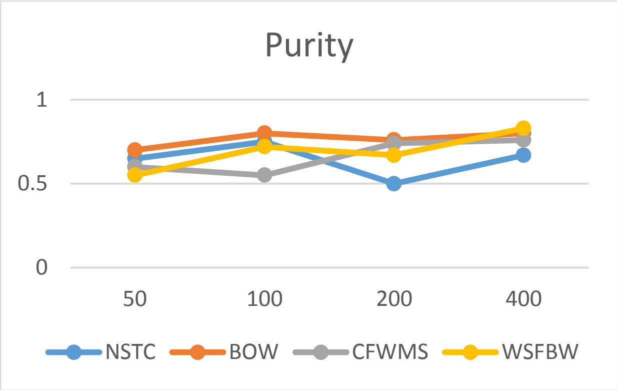

Fig. (5). Purity

D. Entropy

The value of entropy should be as less as possible for better and efficient results and it can be seen that the WSFBW slightly performs better than the BOW approach in figure 6.

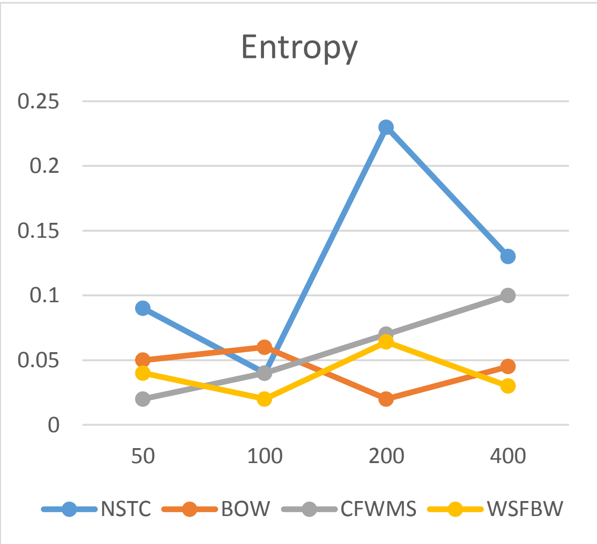

Fig. (6). Entropy

V. CONCLUSION

After analysing the results, it can be concluded that based on the datasets used, the proposed approach of Word Sequence from Word-of-Graph (WSFWS) outperforms the other approaches as it ensures the correlation between the words and captures the true meaning behind the textual document. It is producing better clustering results than other algorithms that are compared in the report. This approach reduces the size of document as the document is represented in terms of word sequences that are extracted from bag-of-words. This refined document still retains the semantics present in the actual document hence the results keep improving as the documents are increased.

REFERENCES

[1] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data Clustering: a review," *ACM Computing Survey*, vol. 31, no. 3 pp. 264-323, September 1999.

[2] A. Campi and S. Ronchi, "The Role of Clustering in Search Computing," In *Proceedings of 20th International Workshop on Databases and Expert Systems Application*, 2009, pp: 432- 436.

[3] D. R. Cutting, D. R. Karger, J. O. Pedersen, and J. W. Tukey, "Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections," In *Proceedings of Fifteenth Annual International ACM SIGIR Conference*, 1992, pp. 318-329.

[4] M. A. Hearst, and J.O. Pedersen, J. O., "Re-examining the cluster hypothesis: scatter/gather on retrieval results," In *Proceedings of 19th annual international ACM SIGIR conference on Research and development in information retrieval*, 1996, pp: 74-84.

[5] G. Amato, V. Dohnal, M. Batko and M. B. Pavel Zezula, Similarity Search-The Metric Space Approach: Springer, 2006.

[6] I. Kaufman and P. J. Rousseeuw, Finding Groups in Data: An Introduction to Cluster Analysis: John Wiley & Sons, 1990.

[7] B. Larsen and C. Aone, "Fast and effective text mining using linear time document clustering," *In Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1999, pp. 16-22.

[8] M. Steianbach, G. Karypis, and V. Kumar, "A comparison of document clustering techniques," *In Proceedings of KDD-Workshop on Text Mining*, 2000.

[9] K. M. Hammouda and M. S. Kamel,"Efficient Phrase-Based Document Indexing for Web Document Clustering," *IEEE Transaction on Knowledge and Data Engineering*, vol. 16, no. 10, pp: 1279-1296, 2004.

[10] C. Hung and D. Xiaotie, "Efficient Phrase-Based Document Similarity for Clustering," *IEEE Transaction on Knowledge and Data Engineering*, vol. 20, no. 9, pp. 1217-1229, September 2008.

[11] B.C.M. Fung, K. Wang and M. Ester, "Hierarchical document clustering using frequent Item sets," In *Proceedings of SIAM International Conference on Data Mining*, 2003.

[12] Y. Li, S. M. Chung, and J. D. Holt. "Text document clustering based on frequent word meaning sequences." *Data & Knowledge Engineering*, vol. 64, no.1, pp: 381-404, 2008.

[13] G. M. Landau and U. Vishkin, “Fast parallel and serial approximate string matching”, *Journal of Algorithms*, vol. 10, no. 2, pp: 157-169, 1989.

[14] C. Fellbaum, Ed., *WordNet: An Electronic Lexical Database*, MIT Press, 1998.

[15] R, François and M. Vazirgiannis. “Graph-of-word and TW-IDF: New Approach to Ad Hoc IR" In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, 2013, pp: 59-68.

[16] R. Mihalcea and P. Tarau. “TextRank: bringing order into texts”. In *Proceedings of Empirical Methods in Natural Language Processing* (*EMNLP ’04)*, 2004.

[17] G. Erkan and D. R. Radev. “LexRank: graph-based lexical centrality as salience in text summarization”. *Journal of Artificial Intelligence Research*, vol. 22, no. 1, pp: 457–479, July 2004.

[18] A. E. Motter, A. P. S. de Moura, Y.-C. Lai, and P. Dasgupta. “Topology of the conceptual network of language”. *Physical Review E*, vol. 65, no. 065102(R), 2002.

[19] R. Blanco and C. Lioma. “Graph-based term weighting for information retrieval”. *Information Retrieval*, vol. 15, no. (1), pp: 54–92, February 2012.

[20] O. Jespersen. The Philosophy of Grammar. Chicago: University of Chicago Press, 1992.

[21] A. Singhal, J. Choi, D. Hindle, D. Lewis, and F. Pereira. “AT&T at TREC-7”. In *Proceedings of the 7th Text Retrieval Conference* (TREC-7), 1999, pp: 239–252.

[22] S. E. Robertson, S. Walker, K. Spärck Jones, M. Hancock-Beaulieu, and M. Gatford. “Okapi at TREC-3”. In *Proceedings of the 3rd Text Retrieval Conference* (*TREC-3*), 1994, pp: 109–126.

[23] C. Zhai and J. Lafferty. “A study of smoothing methods for language models applied to ad hoc information retrieval”. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, (SIGIR ’01)*, 2001, pp: 334–342.

[24] G. Amati and C. J. Van Rijsbergen. “Probabilistic models of information retrieval based on measuring the divergence php from randomness”. *ACM Transactions on Information Systems*, vol. 20, no. 4, pp: 357–389, October 2002.

[25] C. D. Manning, P. Raghavan, and H. Schütze. Introduction to Information Retrieval: Cambridge University Press, 2008.