Scientific Journal Impact Factor (SJIF) : 1.711

International Journal of Modern Trends in Engineering and Research

ISSN (ONLINE) : 2349 - 9745 ISSN (PRINT) : 2393 - 8161

A CONCEPT BASED TEXT MINING AND CLUSTERING FOR SUMMARIZATION

Dr. G. Rasitha Banu MCA., M.Phil., Ph.D.[1], VK Chitra* MCA.,B.ED., Mphil Scholar[2]

[1] Assistant Professor, Department of Health Information Management and Technology,
*Faculty of Public Health and Tropical Medicine,*Jazan University,*Kingdom of Saudi Arabia.*
[2]Department of Computer Science, Mother Teresa Women's University,Chennai, India

ABSTRACT - Most of the common techniques in text mining are based on the statistical analysis of a term either word or phrase. Statistical analysis of a term frequency captures the importance of the term within a document only. However, two terms can have the same frequency in their documents, but one term contributes more to the meaning of its sentences than the other term. Thus, the underlying text mining model should indicate terms that capture the semantics of text. In this case, the mining model can capture terms that present the concept of the sentence, which leads to discover the topic of the document. A new concept-based mining model that analyzes terms on the sentence, document, and corpus levels is introduced. The concept-based mining model can effectively discriminate between non-important terms with respect to sentence semantics and terms which hold the concepts that represent the sentence meaning. The similarity between documents is calculated based on a new concept-based similarity measure. The proposed similarity measure takes full advantage of using the concept analysis measures on the sentence, document, and corpus level in calculating the similarity between documents.

Key Words – Similarity Measures, Clustering Algorithms, Methodology, Implementations and Results

I. INTRODUCTION

NATURAL Language Processing (NLP) is both a modern computational technology and a method of investigating and evaluating claims about human language itself. NLP is a term that links back into the history of Artificial Intelligence (AI), the general study of cognitive function by computational processes, with an emphasis on the role of knowledge representations. Text mining attempts to discover new, previously unknown information by applying techniques from natural language processing and data mining. Clustering, one of the traditional data mining techniques is an unsupervised learning paradigm where clustering methods try to identify inherent groupings of the text documents, so that a set of clusters is produced in which clusters exhibit high intracluster similarity and low intercluster similarity. The similarity between the documents is measured by one of several similarity measures that are based on such a feature vector.

In this paper, a novel concept-based mining model is proposed. The proposed model captures the semantic structure of each term within a sentence and document rather than the frequency of the term within a document only. In the proposed model, three measures for analyzing concepts on the sentence, document, and corpus levels are computed. Each sentence is labeled by a semantic role labeler that determines the terms which contribute to the sentence semantics associated with their semantic roles in a sentence. Each term that has a semantic role in the sentence, is called a concept. Concepts can be either words or phrases and are totally dependent on the semantic structure of the sentence. When a new document is introduced to the system, the proposed mining model can detect a concept match from this document to all the previously processed documents in the data set by scanning the new document and extracting the matching concepts.

II. SIMILARITY MEASURES TECHNIQUES

A. TEXT PREPROSSING

Text pre-processing is the task of converting a raw text file, essentially a sequence of digital bits, into a well-defined sequence of linguistically-meaningful units: at the lowest level characters representing the individual graphemes in a language's written system, words consisting of one or more characters, and sentences consisting of one or more words. Text pre-processing is an essential part of any NLP system, since the characters, words, and sentences identified at this stage are the fundamental units passed to all further processing stages, from analysis and tagging components, such as morphological analyzers and part-of-speech taggers, through applications, such as information retrieval and machine translation systems. Figure 1 on next page, depicts a generic process model for a text mining system activities.

It is done by five techniques,

- Identifying the Paragraph,
- Separate Sentences
- Label terms
- Remove stop words
- Stem Words

B. CONCEPT BASED ANALYSIS

concept based analysis is a principled way of deriving a *concept hierarchy* or formal ontology from a collection of objects and their properties. Each concept in the hierarchy represents the set of objects sharing the same values for a certain set of properties; and each sub-concept in the hierarchy contains a subset of the objects in the concepts above it.

It is analyzed by three techniques,

- Conceptual term frequency
- Term frequency
- Document Frequency

C. CONCEPT BASED DOCUMENT SIMILARITY

Document similarity measures are crucial components of many text analysis tasks, including information retrieval, document classification, and document clustering. Conventional measures are brittle: they estimate the surface overlap between documents based on the words they mention and ignore deeper semantic connections. We propose a new measure that assesses similarity at both the lexical and semantic levels, and learns from human judgments how to combine them by using machine learning techniques. Experiments show that the new measure produces values for documents that are more consistent with people's judgments than people are with each other. We also use it to classify and cluster large document sets covering different genres and topics, and find that it improves both classification and clustering performance.

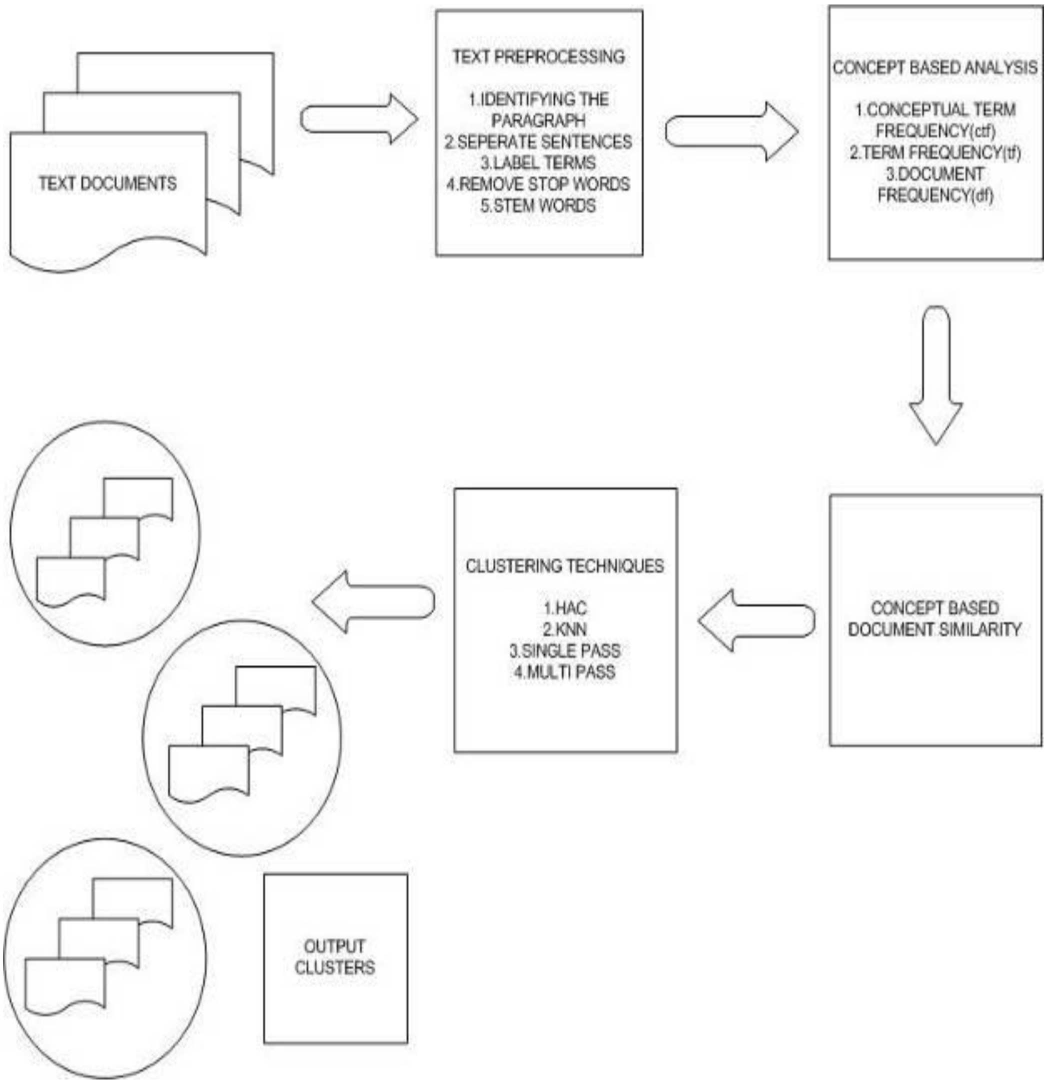

Figure 1. Activity diagram of the system

III. CLUSTERING TECHNIQUES

1. HAC

The **Hierarchical Agglomerative Clustering** methods are most commonly used. The construction of an hierarchical agglomerative classification can be achieved by the following general algorithm.

Find the 2 closest objects and merge them into a cluster

- Find and merge the next two closest points, where a point is either an individual object or a cluster of objects.
- If more than one cluster remains , return to step 2

Individual methods are characterized by the definition used for identification of the closest pair of points, and by the means used to describe the new cluster when two clusters are merged. There are some general approaches to implementation of this algorithm, these being stored matrix and stored data, are discussed below

- In the second approach , an N*N containing all pairwise distance values is first created, and updated as new clusters are formed. This approach has at least an O(n*n) time requirement, rising to O(n^3) if a simple serial scan of dissimilarity matrix is used to identify the points which need to be fused in each agglomeration, a serious limitation for large N.
- The stored data approach required the recalculation of pairwise dissimilarity values for each of the N-1 agglomerations, and the O(N) space requirement is therefore achieved at the expense of an O(N^3) time requirement

2. KNN

In pattern recognition, the *k*-Nearest Neighbors algorithm (or *k*-NN for short) is a non-parametric method used for classification and regression. In both cases, the input consists of the *k* closest training examples in the feature space. The output depends on whether *k*-NN is used for classification or regression:

- In *k-NN classification*, the output is a class membership. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its *k* nearest neighbors (*k* is a positive integer, typically small). If *k* = 1, then the object is simply assigned to the class of that single nearest neighbor.
- In *k-NN regression*, the output is the property value for the object. This value is the average of the values of its *k* nearest neighbors.

3. SINGLE PASS

Single pass is defined as the clustering of data that arrive continuously such as telephone records, multimedia data, financial transactions etc. Data stream clustering is usually studied as a streaming algorithm and the objective is, given a sequence of points, to construct a good clustering of the stream, using a small amount of memory and time.

We have the following set of documents and terms, and that we are interested in clustering the terms using the single pass method (note that the same method can beused to cluster the documents, but in that case, we would be using the document vectors (rows) rather than the term vector (columns).

	T1	T2	T3	T4	T5
Doc1	1	2	0	0	1
Doc2	3	1	2	3	0
Doc3	3	0	0	0	1
Doc4	2	1	0	3	0
Doc5	2	2	1	5	1

Start with T1 in a cluster by itself, say C1. At this point, C1 contains only one item, T1, so the centroid of C1 is simply the vector for T1:
C1 = <1, 3, 3, 2, 2>.

Now compare (i.e., measure similarities) of the next item (T2) to centroids of all existing clusters. At this point we have only one cluster, C1 (we will use dot product for simplicity):
SIM(T2, C1) = 1*2 + 1*3 + 0*3 + 1*2 + 2*2 = 11

Now we need a pre-specified similarity threshold. Let's say that our threshold is 10. This means that if the similarity of T2 to the cluster centroid is >= 10, then we add T2 to the cluster, otherwise we use T2 to start a new cluster.

In this case. SIM(T2, C1) = 11 > 10. Therefore we add T2 to cluster C1. We now need to compute the new centroid for C1 (which now contains T1 and T2). The centroid (which is the average vector for T1 and T2 is:
C1 = <3/2, 4/2, 3/2, 3/2, 4/2>

Now, we move to the next item, T3. Again, there is only one cluster, C1, so we only need to compare T3 with C1 centroid. The dot product of T3 and the above centroid is:
SIM(T3, C1) = 0 + 8/2 + 0 + 0 + 4/2 = 6

This time, T3 does not pass the threshold test (the similarity is less than 10). Therefore, we use T3 to start a new cluster, C2. Now we have two clusters
C1 = {T1, T2}
C2 = {T3}

We move to the next unclustered item, T4. Since we now have two clusters, we need to compute the MAX similarity of T4 to the 2 cluster centroids (note that the centroid of cluster C2 right now is just the vector for T3):
SIM(T4, C1) = <0, 3, 0, 3, 5> . <3/2, 4/2, 3/2, 3/2, 4/2>
= 0 + 12/2 + 0 + 9/2 + 20/2 = 20.5
SIM(T4, C2) = <0, 3, 0, 3, 5> . <0, 2, 0, 0, 1>
= 0 + 6 + 0 + 0 + 5 = 11
Note that both similarity scores pass the threshold (10), however, we pick the MAX, and therefore, T4 will be added to cluster C1. Now we have the following:
C1 = {T1, T2, T4}
C2 = {T3}

The centroid for C2 is still just the vector for T3:
C2 = <0, 2, 0, 0, 1>

and the new centroid for C1 is now:
C1 = <3/3, 7/3, 3/3, 6/3, 9/3>

The only item left unclustered is T5. We compute its similarity to the centroids of existing clusters:
SIM(T5, C1) = <1, 0, 1, 0, 1> . <3/3, 7/3, 3/3, 6/3, 9/3>
= 3/3 + 0 + 3/3 + 0 + 9/3 = 5

SIM(T5, C2) = <1, 0, 1, 0, 1> . <0, 2, 0, 0, 1>
= 0 + 0 + 0 + 0 +1 = 1

Neither of these similarity values pass the threshold. Therefore, T5 will have to go into a new cluster C3. There are no more unclustered items, so we are done (after making a single pass through the items). The final
clusters are:
C1 = {T1, T2, T4}
C2 = {T3}
C3 = {T5}

Note: Obviously, the results for this method are highly dependent on the similarity threshold that is used. You should use your judgment in setting this threshold so that you are left with a reasonable number of clusters.

4. MULTI PASS

Most co reference resolution models determine if two mentions are co referent using a single function over a set of constraints or features. This approach can lead to incorrect decisions as lower precision features often overwhelm the smaller number of high precision ones. To overcome this problem, we propose a simple co reference architecture based on a sieve that applies tiers of deterministic co reference models one at a time from highest to lowest precision. Each tier builds on the previous tier' entity cluster output. Further, our model propagates global information by sharing attributes (e.g., gender and number) across mentions in the same cluster. This cautious sieve guarantees that stronger features are given precedence over weaker ones and that each decision is made using all of the information available at the time. The framework is highly modular: new co reference modules can be plugged in without any change to the other modules. In spite of its simplicity, our approach outperforms many state-of-the-art supervised and unsupervised models on several standard corpora. This suggests that sieve based approaches could be applied to other NLP tasks.

IV. METHODOLOGY

A new concept-based similarity measure which makes use of the concept analysis on the sentence, document, and corpus levels is proposed. These are,

- Text Mining
- Concept Based Mining Model
- Text Clustering
- Concept-based Statistical Analyzer
- Conceptual Ontological Graph (COG)
- Concept-based Extractor Algorithm

(I) TEXT MINING

Text mining usually involves the process of structuring the input text (usually parsing, along with the addition of some derived linguistic features and the removal of others, and subsequent insertion into a database), deriving patterns within the structured data, and finally evaluation and interpretation of the output.

(2) CONCEPT BASED MINING MODEL

The proposed model can efficiently find significant matching concepts between documents, according to the semantics of their sentences. The similarity between documents is calculated based on a new concept-based similarity measure. A raw text document is the input to the proposed model. Each document has well defined sentence boundaries. Each sentence in the document is labeled automatically based on the Prop Bank notations. The sentence that has many labeled verb argument structures includes many verbs associated with their arguments. The labeled verb argument structures, the output of the role labeling task, are captured and analyzed by the concept-based model on the sentence and document levels. In this model, both the verb and the argument are considered as terms. One term can be an argument to more than one verb in the same sentence. This means that this term can have more than one semantic role in the same sentence.

(3) TEXT CLUSTERING

Clustering is an unsupervised classification process; differently from supervised classification no a priori information about classes is required. Document clustering is an optimization process which attempts to determine a partition of the document collection so that documents within the same cluster are as similar as possible (cluster compactness) and the discovered clusters as separate as possible (cluster distinctness). Document clustering algorithms are used in a variety of tasks and applications for facilitating automatic organization, browsing, summarization, and retrieval of structured and unstructured documents.

(4) CONCEPT-BASED STATISTICAL ANALYZER

The objective of this task is to achieve a concept-based statistical term analysis (word or phrase) on the sentence and document levels rather than a single-term analysis in the document set only. The ctfis the number of occurrences of concept c in verb argument structures of sentence s. The concept c, which frequently appears in different verb argument structures of the same sentence s, has the principal role of contributing to the meaning of s. The major challenge of clustering is to efficiently identify meaningful groups that are concisely annotated.

(5) CONCEPTUAL ONTOLOGICAL GRAPH (COG):

The COG representation is a conceptual graph G = (C,R) where the concepts of the sentence, are represented as vertices (C). The relations among the concepts such as agents, objects, and actions are represented as (R). C is a set of nodes (c1, c2,…,cn), where each node c represents a concept in the sentence or a nested conceptual graph G; and R is a set of edges (r1; r2,….,rm), such that each edge r is the relation between an ordered pair of nodes (ci,.., cj). The output of the role labeling task, hich are verbs and their arguments are presented as concepts with relations in the COG representation. This allows the use of more informative concept matching at the sentence-level and the document-level rather than individual word matching. The concept-based model proposes new weight to each position in the COG representation to achieve more accurate analysis with respect to the sentence semantics. Thus, each concept in the COG representation is assigned a proposed weight, which is weight COG, based on its position in the representation.

(6) CONCEPT-BASED EXTRACTOR ALGORITHM

The concept extractor algorithm describes the process of combining the weightstat(computed by the concept-based statistical analyzer) and the weightCOG(computed by the COG representation) into one new combined weight called weightcomb. The concept extractor selects the top concepts that have the maximum weightcombvalue. The proposed weightcombis calculated by:

weightcombi= weightstati * weightCOGi(5)

The procedure begins with processing a new document which has well defined sentence boundaries. Each sentence is semantically labeled . For each labeled sentence, concepts of the verb argument structures which represent the semantic structures of the sentence are extracted to construct the COG representation. The concepts list L is sorted descendingly based on the weightcombvalues. The maximum weighted concepts are chosen as top concepts from the concepts list L. The concept extractor algorithm is capable of extracting the top concepts in a document (d) in O(m) time, where m is the number of concepts.

V. IMPLEMENTAION

SYSTEM IMPLEMENTATION

The objective behind the concept-based analysis task is to achieve an accurate analysis of concepts on the sentence, document, and corpus levels rather than a single-term analysis on the document only.

*** Sentence-Based Concept Analysis:**

To analyze each concept at the sentence level, a new concept-based frequency measure, called the conceptual term frequency ðctfÞ is proposed. The ctf calculations of concept c in sentence s and document d are as follows:

*** Calculating ctf of Concept c in Sentence s:**

The ctf is the number of occurrences of concept c in verb argument structures of sentence s. The concept c, which frequently appears in different verb argument structures of the same sentence s, has the principal role of contributing to the meaning of s. In this case, the ctf is a local measure on the sentence level.

*** Calculating ctf of Concept c in Document d:**

A concept c can have many ctf values in different sentences in the same document d. Thus, the ctf value of concept c in document d is calculated by:

$$ct\ \mathbf{f} = \frac{\sum_{n=1}^{\mathbf{sn}} \mathbf{ctf}}{sn}$$

where sn is the total number of sentences that contain concept c in document d. Taking the average of the ctf values of concept c in its sentences of document d measures the overall importance of concept c to the meaning of its sentences in document d. A concept, which has ctf values in most of the sentences in a document, has a major contribution to the meaning of its sentences that leads to discover the topic of the document. Thus, calculating the average of the ctf values measures the overall importance of each concept to the semantics of a document through the sentences. To illustrate the calculation of ctf in a document, consider a concept c which appears twice in document d in the first and the second sentences. The concept c appears five times in the verb argument structures of the first sentence s1, and three times in the verb argument structures of the second sentence s2. In this case, the ctf value of concept c is equal to $\frac{5+3}{2}$ =4.

***Document-Based Concept Analysis:**

To analyze each concept at the document level, the concept based term frequency tf , the number of occurrences of a concept (word or phrase) c in the original document, is calculated. The tf is a local measure on the document level.

***Corpus-Based Concept Analysis:**

To extract concepts that can discriminate between documents, the concept-based document frequency df, the number of documents containing concept c, is calculated. The df is a global measure on the corpus level. This measure is used to reward the concepts that only appear in a small number of documents as these concepts can discriminate their documents among others. The process of calculating ctf, tf , and df measures in a corpus is attained by the proposed algorithm which is called Concept-based Analysis Algorithm.

***Concept-Based Analysis Algorithm**

```
1. ddoci is a new Document
2. L is an empty List (L is a matched concept list)
3. sdoci is a new sentence in ddoci
4. Build concepts list Cdoci from sdoci
5. for each concept ci 2 Ci do
6. compute ctfi of ci in ddoci
7. compute tfi of ci in ddoci
8. compute dfi of ci in ddoci
9. dk is seen document, where k= { 0,1 . . ; doci _ 1}
10. sk is a sentence in dk
11. Build concepts list Ck from sk
12. for each concept cj 2 Ck do
13. if (ci == cj) then
14. update dfi of ci
15. compute ctfweight ¼ avgðctfi; ctfjÞ
16. add new concept matches to L
17. end if
18. end for
19. end for
20. output the matched concepts list L
```

The concept-based analysis algorithm describes the process of calculating the ctf, tf , and df of the matched concepts in the documents. The procedure begins with processing a new document (at line 1) which has well defined sentence boundaries. Each sentence is semantically labeled according to [23]. The lengths of the matched concepts and their verb argument structures are stored for the concept-based similarity calculations in Section 3.6. Each concept (in the for loop, at line 5) in the verb argument structures, which represents the semantic structures of the sentence, is processed sequentially. Each concept in the current document is matched with the other concepts in the previously processed documents. To match the concepts in previous documents is accomplished by keeping a concept list L, which holds the entry for each of the previous documents that shares a concept with the current document.

After the document is processed, L contains all the matching concepts between the current document and any previous document that shares at least one concept with the new document. Finally, L is output as the list of documents with the matching concepts and the necessary information about them. The concept-based analysis algorithm is capable of matching each concept in a new document (d) with all the previously processed documents in O(m) time, where m is the number of concepts in d.

VI. RESULTS

To test the effectiveness of concept matching in determining an accurate measure of the similarity between documents, extensive sets of experiments using the concept-based term analysis and similarity measure are conducted. The experimental setup consisted of four data sets. The first data set contains 23,115 ACM abstract articles collected from the ACM digital library. The ACM articles are classified according to the ACM computing classification system into five main categories: general literature, hardware, computer systems organization, software, and data. The second data set has 12,902 documents from the Reuters 21,578 data set. There are 9,603 documents in the training set, 3,299 documents in the test set, and 8,676 documents are unused. Out of the five category sets, the topic category set contains 135 categories, but only 90 categories have at least one document in the training set. These 90 categories were used in the experiment. The third data set consisted of 361 samples from the Brown corpus. Each sample has 2;000+ words.

The Brown corpus main categories used in the experiment were press: reportage; press: reviews, religion, skills and hobbies, popular lore, belles-letters, and learned; fiction: science; fiction: romance and humor. The fourth data set consists of 20,000 messages collected from 20 Usenet newsgroups. In the data sets, the text directly is analyzed, rather than, using metadata associated with the text documents. This clearly demonstrates the effect of using concepts on the text mining process. The similarities which are calculated by using the sentence-based, document-based, corpus-based and the combined approach concept analysis are used to compute four similarity matrices among documents. Three standard document clustering techniques are chosen for testing the effect of the concept-based similarity on clustering [28]: 1) Hierarchical Agglomerative Clustering (HAC), 2) Single-Pass Clustering, and 3) k-Nearest Neighbor (k-NN).2.

VII. CONCLUSION AND FUTURE WORK

This work bridges the gap between natural language processing and text mining disciplines. A new concept based mining model composed of four components, is proposed to improve the text clustering quality. By exploiting the semantic structure of the sentences in documents, a better text clustering result is achieved. The first component is the sentence-based concept analysis which analyzes the semantic structure of each sentence to capture the sentence concepts using the proposed conceptual term frequency ctf measure. Then, the second component, document-based concept analysis, analyzes each concept at the document level using the concept-based term frequency tf. The third component analyzes concepts on the corpus level using the document frequency df global measure. The fourth component is the concept-based similarity measure which allows measuring the importance of each concept with respect to the semantics of the sentence, the topic of the document, and the discrimination among documents in a corpus. By combining the factors affecting the weights of concepts on the sentence, document, and corpus levels, a concept-based similarity measure that is capable of the accurate calculation of pair wise documents is devised. This allows performing concept matching and concept-based similarity calculations among documents in a very robust and accurate way. The quality of text clustering achieved by this model significantly surpasses the traditional single term-based approaches. There are a number of possibilities for extending this paper. One direction is to link this work to Web document clustering. Another direction is to apply the same model to text classification. The intention is to investigate the usage of such model on other corpora and its effect on classification, compared to that of traditional methods.

REFERENCES

[1] Ali Shah, N. & M. ElBahesh, E. “Topic-Based Clustering of News Articles”, University of Alabama at Birmingham. Retrieved September 23, 2013 from: http://pdf.aminer.org/000/006/766/topic_based_clustering_of_news_articles.pdf

[2] Bouras, C. & Tsogkas, V. 2010. "W-kmeans: Clustering News Articles Using WordNet". Retrieved September 5, 2013 from: http://ru6.cti.gr/ru6/publications/116262780379.pdf

[3] Buscaldi, D., Rosso, P. & Arnal Sanchis, E. 2005. “A WordNet-based Query Expansion method for Geographical Information Retrieval”. Retrieved September 2, 2013 from:
http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.99.8031&rep=rep1&type=pdf

[4] Manning, C., Raghavan, P. & Schütze, H. 2008. “Introduction to Information Retrieval”, Cambridge, England: Cambridge University Press. Retrieved September 4, 2013 from:
http://nlp.stanford.edu/IR-book/pdf/irbookonlinereading.pdf

[5] A Lew and H. Mauch ― Intoduction to Data Mining Principles‖ ,SCI, springer, 2006.

[6] V. Gupta, G.S. Lehal ― A Survey of Text Mining Techniques and applications ―, Journal of Emerging Technologies in Web Intelligence,2009.

[7] Patnaik, Sovan Kumar, Soumya Sahoo, and Dillip Kumar Swain, “Clustering of Categorical Data by Assigning Rank through Statistical Approach,” *International Journal of Computer Applications* 43.2: 1-3, 2012.

[8] Arockiam, L., S. S. Baskar, and L. Jeyasimman. 2012. Clustering Techniques in Data Mining.

[9] Han, J., Kamber, M. 2012. Data Mining: Concepts and Techniques, 3rd ed, 443-491.

[10] [Berry Michael W., (2004), “Automatic Discovery of Similar Words”, in “Survey of Text Mining: Clustering, Classification and Retrieval”, Springer Verlag, New York, LLC, 24-43.

[11] Navathe, Shamkant B., and Elmasri Ramez, (2000), “*Data Warehousing and Data Mining*”, in “*Fundamentals of Database Systems*”, Pearson Education pvt Inc, Singapore, 841-872.

[12] Weiguo Fan, Linda Wallace, Stephanie Rich, and Zhongju Zhang, (2005), “*Tapping into the Power of Text Mining*”, Journal of ACM, Blacksburg.

[13] Sergio Bolasco, Alessio Canzonetti, Francesca DellaRatta-Rinald and Bhupesh K. Singh, (2002), “Understanding Text Mining: a Pragmatic Approach”, Roam, Italy.

[14] Liu Lizhen, and Chen Junjie, China (2002), “ Research of Web Mining”, Proceedings of the 4th World Congress on Intelligent Control and Automation, IEEE, 2333-2337.

[15] Haralampos Karanikas and Babis Theodoulidis Manchester, (2001), “Knowledge Discovery in Text and Text Mining Software”, Centre for research in Information Management, UK Barzilay, M. Kan, and K.R. McKeown, “SIMFINDER: A Flexible Clustering Tool for Summarization,” Proc. NAACL Workshop Automatic Summarization, pp. 41-49, 2001.

[16] H. Zha, “Generic Summarization and Keyphrase Extraction Using Mutual Reinforcement Principle and Sentence Clustering,” Proc. 25th Ann. Int’l ACM SIGIR Conf. Research and Development in Information Retrieval, pp. 113-120, 2002.

[17] D.R. Radev, H. Jing, M. Stys, and D. Tam, “Centroid-Based Summarization of Multiple Documents,” Information Processing and Management: An Int’l J., vol. 40, pp. 919-938, 2004.

[18] R.M. Aliguyev, “A New Sentence Similarity Measure and Sentence Based Extractive Technique for Automatic Text Summarization,” Expert Systems with Applications, vol. 36, pp. 7764- 7772, 2009.

[19] R. Kosala and H. Blockeel, “Web Mining Research: A Survey,” ACM SIGKDD Explorations Newsletter, vol. 2, no. 1, pp. 1-15, 2000.

[20] G. Salton, Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer. Addison- Wesley, 1989.