SUMMARIZATION USING NTC APPROACH BASED ON KEYWORD EXTRACTION FOR DISCUSSION FORUMS

N.Lalithamani[1], K.Alagammai[2], Kolluru Kamala Sowmya[3], L.Radhika[4], Raga Supriya DarisiS[5], Shanmuga Priya[6]

[1]Assistant Professor(SG),Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham (University),Coimbatore - 641 112.
[2]4th year BTech, Computer Science and Engineering Student Amrita School of Engineering, Amrita Vishwa Vidyapeetham (University), Coimbatore - 641 112
[3]4th year BTech, Computer Science and Engineering Student Amrita School of Engineering, Amrita Vishwa Vidyapeetham (University), Coimbatore - 641 112
[4]4th year BTech, Computer Science and Engineering Student Amrita School of Engineering, Amrita Vishwa Vidyapeetham (University), Coimbatore - 641 112
[5]4th year BTech, Computer Science and Engineering Student Amrita School of Engineering, Amrita Vishwa Vidyapeetham (University), Coimbatore - 641 112
[6]4th year BTech, Computer Science and Engineering Student Amrita School of Engineering, Amrita Vishwa Vidyapeetham (University), Coimbatore - 641 112

Abstract

Internet has become a ubiquitous medium of communication, be it through any social networking websites like Facebook, Twitter or any discussion forums like Yahoo Answers, Quora, Stack Overflow. One can participate in any kind of discussion ranging from politics, education, spirituality, philosophy, science and geography to medicine and many more. Often, most of the discussion forums are loaded up with data. Hence, when a new user wants to know the public opinion, it is impossible for him/her to go through all the tens or hundreds of threads or comments under a particular thematic discussion. The problem here is - we are buried in data but we starve for information. So, to solve this problem, we are proposing a novel approach called Discussion Summarization which is aimed at presenting the user with the most relevant summary containing all the important points of the discussion. This allows the user to easily and quickly grasp and catch up on the on-going conversation in a discussion thread. The summary generated follows CRS approach (Clustering and Ranking and Score calculation for each sentence).The Cluster based Summarization technique is coupled with Nested Thematic Clustering (NTC) and Corpus Based Semantic Similarity (CBSS) approaches. The summary produced is the set of top-ranked sentences (of high scores). Results have shown that a completely unbiased summary with the multidimensionality of comments is generated.

Keywords *- Clustering based Summarization, Corpus Based Semantic Similarity, Discussion Summarization, Nested Thematic Clustering, Ranking.*

----------***----------

1. INTRODUCTION

There is always a buzz in the world to know the current news, the need to be updated on the happenings at various places to various people. Now, the trend has changed from the 'need to know' to the 'need to participate'. In either case, there is a necessity to get our hands dirty with the information, but not just with the data. One way to catch up to what's happening is through internet – blogs, discussion forums, social networking sites etc. With the ever-evolving data everywhere, it gets very difficult for an individual to actually spend some time and effort to read all the comments under a post(on a theme).

Hence, inorder to solve this problem, we have come up with a bright solution, which is Discussion Summarization. A general summarization can be defined as the process presentation of distilled or filtered information containing the key points of the discussion, whereas aDiscussion summarization aims at presenting an extractive summary of a thematic discussion by clustering and rankingthe discussion threads based on their similarity [4].According to Maniand M. T. Maybury[14], discussion summarization can be defined as the process of extracting the most important information from a source (or sources) to produce an abridged version for a particular user (or users) and task (or tasks).We aim at presenting the user with a relevant and non-redundant extractive summary (of the discussion threads) based on the credible approach - CRS, which is a combination of clustering, ranking and score calculation. Each of them is described below in detail.

A summarization can be of two types generally – abstractive or extractive [15]. An abstractive summarization involves

identification of keywords and later framing sentences based on the meaning of the sentences. Extractive methods work by selecting a subset of existing words, phrases, or sentences in the original text to form the summary. The former approach involves implementation of machine learning algorithms and is generally very difficult whereas as the later is comparatively easier as it requires only a set of formulae for identification of the most relevant sentences (threads of discussion). Lot of present research is still on the abstractive approach (as it deals with semantics).

Clustering can be defined as the grouping or bundling up of data based on their similarities. It is mainly used in data mining and statistical data analysis like information retrieval, machine learning etc. Clustering is of two types – hard and soft clustering. In hard clustering, data is divided into distinct clusters even though each data element belongs to exactly one cluster (or document or in this case, a thread of discussion). In contrast, soft clustering, which is also known as fuzzy clustering; the data elements can belong to more than one cluster. Here, we follow the soft clustering approach.

Ranking is defined as the retrieval of most relevant(top scorer(s)) sentences based on the scores. Here, ranking is done for each sentence with the help of a Bi-Type graph model while considering the users rating the comments (if available). The score calculation of each sentence is the summation of value of importance of each term or word in the sentence. Thus, a score for each sentence is entirely based on the scores of its constituent words.

The data retrieval from the web can be done by using pattern module of Natural Language Processing Toolkit (NLTK). The text analysis is carried using the Natural Language Processing (NLP) along with the help of python. For data retrieval in this case, we have identified our domain to be Yahoo Answers (because it provides an easy access to its data with its API). We have used YQL(Yahoo Query Language) to parse the data to an XML format. Then, we retrieve the data from the XML code using a small snippet of code and paste it in a text editor. Before proceeding any further, we make sure the data is immediately brought into the proper format. That is, if the data is in the chat style (informal language),then the necessary corrections are made. This is done with the help of Chat corpus from the NLTK.

Hence, the discussion summarization is aimed at providing a relevant (high precision and recall) and non-redundant summary to the user.

2. RELATED WORK

General text summarization can be done using many methods – extractive, abstractive, aided and entropy based [4], [15].

Sawkat Ali et al. [24] describe bagging which is a popular method that improves the classification accuracy for any learning algorithm. A trial and error classifier feeding with the Bagging algorithm is a regular practice for classification tasks in the machine learning community. In this work a rule based method using statistical information for unique classifier selection is proposed.

Xiaoyan Cai, and Wenjie Li[27], Address the problem of "context independent document indexing" using the lexical association between document terms. QCS system (query, cluster, and summarize), retrieves relevant documents in response to a query, clusters these documents by topic and produces a summary for each cluster.

Fei liu et al. [26] handles the problem of automatic keyword extraction in the meeting domain, a genre significantly different from written text. For the supervised framework, a rich set of features beyond the typical TF-IDF measures, such as sentence salience weight, lexical features, summary sentences, and speaker information are proposed to be taken into account.

There exists numerous ways to carry out the clustering, ranking and extraction mechanisms. Few of them are discussed here.For the keyword Extraction purposes, we have

KEA (**Keyword Extraction Algorithm**) - This is anopen-source, backed by solid research, comes with some annotated training data, and it can extract key phrases over unrestricted text, without the need of vocabulary for possible key phrases.

Maui - Maui automatically identifies main topics in text documents. Depending on the task, topics are tags, keywords, key phrases, vocabulary terms, descriptors, index terms or titles of Wikipedia articles.

The Yahoo term extraction API - It is only available through YQL. It results in low recall but high precision.

Also, we have many more software like alchemy API, Ter-mine by NacTem etc., but we will be using Yahoo Term extraction API. Likewise, for finding similarity, Cosine (calculates the distance between a sentence and the entire document.), Centroid score and CBSS (Corpus Based Semantic Similarity) are few algorithms that are used to find the similarity between any two sentences.

Cosine Similarity

$$\cos(\vec{q},\vec{d}) = \frac{\vec{q}\bullet\vec{d}}{|\vec{q}||\vec{d}|} = \frac{\vec{q}}{|\vec{q}|}\bullet\frac{\vec{d}}{|\vec{d}|} = \frac{\sum_{i=1}^{|V|} q_i d_i}{\sqrt{\sum_{i=1}^{|V|} q_i^2}\sqrt{\sum_{i=1}^{|V|} d_i^2}}$$

Also, different forms of the tf–idf weighting scheme are often used by search engines as a central tool in scoring and ranking a document's relevance given a user query. This tf–idf, term frequency – inverse document frequency lists out the union of more frequent and lessfrequent but most important words. The same can be successfully used for stop-words (which adds less meaning to the context)

filtering in various subject fields including text summarization and classification.

Corpus-Based Semantic Similarity

In case of CBSS algorithm, all the terms are given weightage as per their respective tf(term frequency) and idf(inverse document frequency) values[16]. Hence, we use this model in this paper.

3. PROPOSED MODEL

In this paper, we put forward a SIX step process to generate a summary. They are:

[1] Retrieval of data from Yahoo Answers.
[2] Formatting the data.
[3] Keyword extraction.
[4] Clustering the sentences.
[5] Ranking the sentences.
[6] Selecting and Re-ordering the sentences.

After the execution of these steps, the summary of the discussion is presented to the user. The above model is modified version (of [4]) wherein the steps B and C are introduced. The diagram given below is a better representation of the CRS technique.

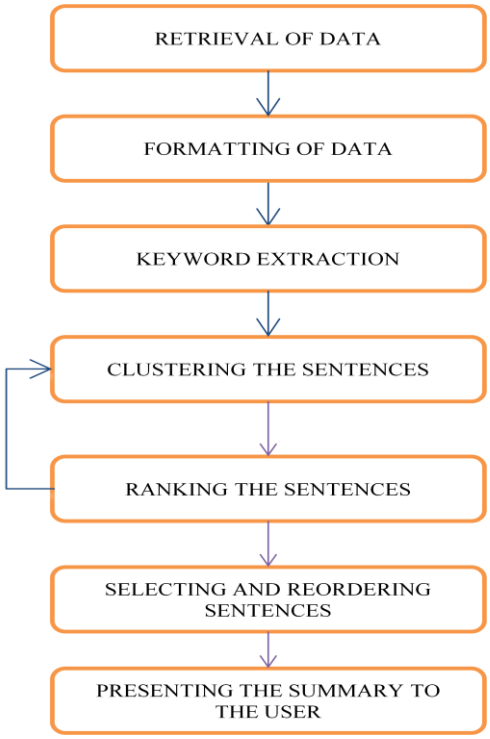

Fig1. Proposed Model

In the above proposed model, step 1 – retrieval of data can be done using pattern module (of NLTK) or any API(For example, Yahoo Answers here).

Then, we move on to the step 2 – formatting the data. We must have read or commented many posts. We would have observed that most of the information inthe discussion forums is informally written – using short-forms, similar phonetics, spelling mistakes etc. but this information may be useful. So, we try not to ignore any information of this type. To do this, we format the whole text to a formal (English in this case) using the Chat corpus in the NLTK.

Next, is most important step in the process of summarization, which is, Keyword identification. How do we identify what is the central idea of the discussion? This is done with the help of any Keyword extraction algorithm. In this case, we follow Yahoo Keyword Extraction algorithm as it is easier to integrate as the entire process involves YQL.

Then, we perform the clustering usingNTC algorithm. It is a two level nested clustering[4]. Initial clustering (Theme Clustering) is based on the keywords and then second level (Topic clusters) is based on the similarity of sentences. This similarity between words or phrases is calculated either using Synsets(or WordNet) in maxSim() or using PCC(Pearson Correlation Coefficient).The main Similarity function is given as[16]:

$$sim(T1,T2) = 0.5 * (X + Y)$$

$$X=\frac{\sum_{w\in\{T1\}}(maxSim\ (w,T2)*idf\ (w))}{\sum_{w\in\{T1\}}(idf\ (w))}$$

$$Y=\frac{\sum_{w\in\{T2\}}(maxSim\ (w,T1)*idf\ (w))}{\sum_{w\in\{T2\}}(idf\ (w))}$$

For each word *w* in segment (sentence) *T1*, we find a word in segment (sentence) *T2* that has the highest semantic similarity to *w* (*maxSim*(*w*, *T2*)). Similarly, for the words in *T2*, we identify the corresponding words in segment *T1*[4].The similarity score of the two text segments is then calculated by combining the similarity of the words in each segment, weighted by their *idf* values (word specificity). The value of $sim(w,w_i)$ is 1, if the word *w* and w_i is present in the sentence T_i. In the above, the maxSim() is modified as the PCC [20] in contrast to the one [4] calculated using synsets. Here, $r_{X,Y}$ is,

$$\frac{\sum_{i=1}^{n}(x_i y_i) - n\bar{x}\bar{y}}{(n-1)s_x s_y}$$

where n is the number of sentences/phrases in each domain(Topic Cluster) and are the respective means of X and Y, $\sigma_X$and σ_Y(S_X and S_Y)are the respective standard deviation of X and Y, and Σ(XY) is the sum of the XY cross-product.If $r_{X,Y} > 0$, X and Y arepositively correlated (X's values increase as Y's). Thus, the maxsim() =1 else, zero.This correlation finds the similarity of data.

NTC – Two Tiered Clustering (TTC) approach [4]. The blue rectangles in the *Fig. 2* depicts the theme clusters, the ones in red are for the similarity clusters and the green rectangles represent the sentences.

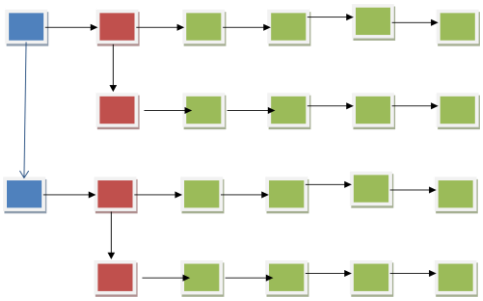

Fig 2 NTC of sentence

Here, Xiaoyan Cai and Wenjie Li's proposed algorithm [1] is modified such that the rankings of the comments given by the different users are done in effective way. The modified version of the original Bi – Type graph model states that:

$$r(s_i)=L+M$$

$$L = \sum_{i=1}^{m} W_{ST}(i,j)$$

$$M = \sum_{i=1}^{n} W_{SS}(i,j)$$

$$W = \begin{pmatrix} W_{SS} & W_{ST} \\ W_{TS} & W_{TT} \end{pmatrix}$$

$$r(t_i) = P+Q$$

$$P = \sum_{i=1}^{n} W_{TS}(j,i)$$

$$Q = \sum_{i=1}^{m} W_{TT}(i,j)$$

Here, E is the set of edges that connects the vertices. An edge can be defined as a connection between any combination of sentences and words. The element represents the weight of the edge connecting two vertices (vertices are terms and sentences) for which $W$is the adjacency matrix. Note that,W can be decomposed into four blocks - W_{SS}, W_{ST}, $W_{TS}$and W_{TT}, each representing a sub-graph of the textual objects indicated by the subscripts [1].

$W_{ST}(i,j)$is the cosine similarity between the sentence s_i and the term t_j. Thus, the value of $W_{ST}(i,j)$ oscillates between 0and 1. If $W_{ST}(i,j)$ is close to 1, it means the sentence s_i andthe term t_j are semantically similar. Else if, $W_{ST}(i,j)$ is close to 0,it means the sentence s_i and the term t_j are semantically different.$W_{SS}(i,j)$ is the cosine similarity between the sentences s_i and the term t_j is equal to the relationships between termsand sentences which are symmetric. The $W_{TT}(i,j)$ value is the measure of cosine similarity between the terms t_i and t_j[1].

To present a coherent summary, it is necessary that we re-order the sentences as per its original order. To achieve this, we compare each of the selected sentences from above with the sentences in the TC and get their relative positions in the summary [4]. Hence, the ordering of the sentences in the final summary is preserved.

4. ANALYSIS OF RESULTS

Initially, we get the information from the Yahoo Answers (using YOL) or any other website using patterns module.

```
from xml.dom.minidom import parse
doc = parse('yql2.xml')
As = doc.getElementsByTagName("Answers")
for Answers in As:
Aa = Answers.getElementsByTagName("Answer")
for Answer in Aa:
subsub = childNodes[0].data
sub= getElementsByTagName("Content")[0]. sub
ANSWER=Answer. sub.subsub
print ANSWER
```

Fig3.Code for the extraction of content from the XML file

From the above, we can successfully, retrieve data from various forums. Now, we use Chat corpus (NLTK) for the formatting of data.

I'm gonna go out now. Il see you in a bit. Byeee!

Fig 4.Input file for data formatting

I am going to go out now. I will see you in a bit. Bye.

Fig 5.Formatted version for the input in the Fig. 4

The Fig. 5 shows the modification of the text from informal language(Fig. 4.) to formal language. Then, we move on to the Keyword Extraction phase.

Italian sculptors and painters of the renaissance favored the Virgin Mary for inspiration.

Fig 6 Input for Keyword Extraction

Usage of Yahoo Keyword Extraction on the above text would give us the keywords and their respective entity scores:

1. Italian sculptors(score -0.803)
2. the Virgin Mary(score – 0.696)

Hence, now these two important key phrases are extracted from the XML format using python.Consider that the input for processing now as the one given below. If you want to include the number of likes for each post in the text file, then you might as well retrieve it and paste in next to each sentence for easy identification as suggested in [4].

If a black cat crosses your way, it's bad luck for you. Turn around, take another route.
Sneezing before doing something good/big is a bad omen. I have not heard of such a belief, but let us think of what it might signify.
In early Egyptian times, Black cats were iconic

character in Animal world.
Take another path and turn around if a cat crosses.
Cats are evil.

Fig. 7.Input file for text analysis

Now, the text analysis is done so as to make sure that the test gets normalised before further processing. This text analysis includes a series of activities like – tokenization, stop words elimination, case folding, stemming, lemmatization etc. Hence, the keywords generated after this is process are given below:

['black', 'cat', 'crosses', 'way', ',', "'s", 'bad', 'luck', '.', 'turn', 'around', ',', 'take', 'another', 'route', '.', 'sneezing', 'something', 'good/big', 'bad', 'omen', '.', 'heard', 'belief', ',', 'let', 'us', 'think', 'might', 'signify', '.', 'early', 'egyptian', 'times', ',', 'black', 'cats', 'iconic', 'character', 'animal', 'world', '.', 'take', 'another', 'path', 'turn', 'around', 'cat', 'crosses', '.', 'cats', 'evil', '.']

Fig. 8 Normalisation of text in fig. 7

Next step is to cluster the contents as per the topics (Topic Clusters). The results of Topic cluster would be as follows:

If a black cat crosses your way, it's bad luck for you.
Turn around, take another route.
In early Egyptian times, Black cats were iconic character in Animal world.
Take another path and turn around if a cat crosses.
Cats are evil.

Fig. 9 Cat Cluster

Note that 'Turn around take another route' and 'Take another path and turn around if a cat crosses' are semantically same. They will be put under the same second level cluster of NTC. So, henceforth, only one of them is taken for further presentation.

Sneezing before doing something good/big is a bad omen.
I have not heard of such a belief, but let us think of what it might signify.

Fig. 10.Sneeze Cluster

After ranking and score calculation, the contents of the summary would be as shown below:

If a black cat crosses your way, it's bad luck for you.
Turn around, take another route.
Sneezing before doing something good/big is a bad omen.
Cats are evil.
I have not heard of such a belief, but let us think of what it might signify.
In early Egyptian times, Black cats were iconic character in Animal world.

Fig. 11.Sentences after ranking

Finally, we re-order sentences as per their occurrences in the original file and present the summary(as shown in fig. 12) to the user.(Based on the user specified length of the summary)

Cat:
If a black cat crosses your way, it's bad luck for you.
Turn around, take another route.
Sneeze:
Sneezing before doing something good/big is a bad omen.

Fig. 12.Final Summary presented to the user

Hence, the final summary obtained is non-redundant and relevant (with respect to recall and precision) and is of the user specified top relevant sentences.

5. DISCUSSIONS

We aim at presenting a non-redundant summary. To attain this, we identify the similar sentences and extract the most relevant sentences to be a part of the summary. Also, we take enough care to chop put the stop words which is in other words, detecting and cropping out the outliers. Stop words are defined as the words that are common and carry less important meaning than keywords. Consider the input file given below.

If a black cat crosses your way, it's bad luck for you. Turn around, take another route. Sneezing before doing something good or bad is a bad omen. If you see a cat, take another path and turn around.

Fig 13 Sample Input file

The bar graph (Fig. 14.) depicts the occurrence of stop words in the input file given in Fig. 13.

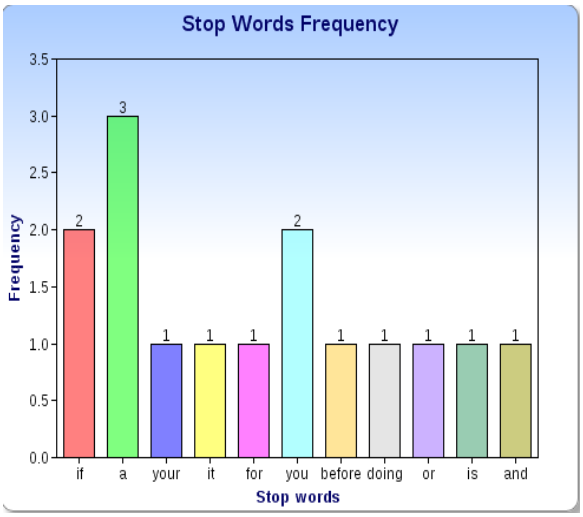

Fig 14.Stop Words Frequency for the file Fig. 13

From the above, it is clear that there is need to eliminate these stop words from the further steps of processing (in identifying the keywords for the generation of the summary).

6. CONCLUSIONS AND FUTURE ENHANCEMENTS

This paper discusses how the summary is generated for a discussion using the concepts of Clustering- Corpus based Semantic Similarity coupled with NTC approach, Ranking – Bi Type Graph model and Score calculation of every sentence.

This project can be extended to an interactive level with the user. In other words, the user can be given the summary after which he/she can be tested by giving some questions (multiple choice questions or subjective questions). Note that the questions are to be from the original text whereas the answers which will be provided by the users are from the summary (assuming that the user has read the summary alone). So, this can be a way to test the completeness and user understanding ability of the summary that is presented to the user although, it has few glitches like – user's level of attention in understanding and answering the questions etc. It also requires an abstractive approach of analysis of text, which is requires a set of machine learning algorithms to be implemented.

REFERENCES

[1] Xiaoyan Cai and Wenjie Li, "Ranking Through Clustering: An integrated approach to multi-document summarization", IEEE Transactions on Audio, Speech and Language Processing, Vol. 21, No. 7, July 2013.

[2] S. Fisher and B. Roark, "Query-focused summarization by supervised sentence ranking and skewed word distributions," in Proceedings of Document Understanding Conference, 2006.

[3] Xiaoyan Cai and Wenjie Li,"Mutually reinforced manifold-ranking based relevance propagation model for query-focused multi-document summarization", IEEE Transactions on Knowledge And Data Engineering, Vol. 25, No. 8, August 2013.

[4] N. Lalithamani, K. Alagammai et al.,"Discussion Summarization", International Journal of Recent Development in Engineering and Technology, Vol. 2, Issue 1, January 2014.

[5] Hien Nguyen, Eugene Santos, and Jacob Russell, "Evaluation of the impact of user-cognitive styles on the assessment of text summarization", IEEE Transactions on Systems, Man and Cybernetics, Vol. 41, No. 6, November 2011.

[6] Chien Chin Chen and Meng Chang Chen, "A content anatomy approach to temporal topic summarization", IEEE Transactions on Knowledge And Data Engineering, Vol.24, No. 1, January 2012

[7] Elias Iosif and Alexandros Potamianos, "Unsupervised semantic similarity computation between terms using web documents", IEEE Transactions on Knowledge And Data Engineering, Vol. 22, No. 11, November 2010.

[8] Davide Falessi, Giovanni Cantone, and Gerardo Canfora, "Empirical principles and an industrial case study in retrieving equivalent requirements via natural language processing Techniques", IEEE Transactions on Software Engineering, Vol. 39, No. 1, November 2013.

[9] Manning, C.D., Raghavan, P. and Schütze, H. (2009), "An Introduction to Information Retrieval. ",England: Cambridge University Press.

[10] X. Wan, "Towards a Unified Approach to Simultaneous Single-Document and Multi-Document Summarizations," Proceedings of 23rd International Conference on Computational Linguistics, pp. 1137-1145,http://portal.acm.org/ citation.cfm?id=1873781.1873909, 2010.

[11] X. Wan, "An Exploration of Document Impact on Graph-Based Multi-Document Summarization," Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 755-762, http://portal.acm.org/ citation.cfm?id=1613715.1613811, 2008.

[12] Celikyilmaz and D. Hakkani-Tur, "Discovery of topically coherent sentences for extractive summarization," in Proceedings of 49th Association for Computational Linguistics Conference, 2011, pp. 491–499.

[13] H. Lin and J. Bilmes, "A class of sub modular functions for document summarization," in Proceedings 49th Association for Computational Linguistics Conference, 2011.

[14] Mani and M. T. Maybury, "Advances in Automatic Text Summarization", Cambridge, MA: MIT Press, 1999.

[15] Automatic Summarization online: http://www.en.wikipedia.org/wiki/Automatic_summarization.

[16] Shasha Xie, Yang Liu, " Using Corpus And Knowledge-Based Similarity Measure In Maximum Marginal Relevance For Meeting Summarization, " The University of Texas at Dallas, Richardson, TX, USA.

[17] Jaime Carbonell and Jade Goldstein, "The use of MMR, diversity-based re-ranking for re-ordering documents and producing summaries," in Proceedings of Special Interest Group in Information Retrieval, 1998.

[18] A. Nenkova, "Automatic Text Summarization of Newswire:Lessons Learned from the Document Understanding Conference,"Proceedings 20th National Conference on Artificial Intelligence (AAAI), pp. 1436-1441, 2005.

[19] J. G. Corbonell and J. Goldstein, "The use of MMR, diversity-basedre-ranking for re-ordering documents and producing summaries," inProceedings 21st Special Interest Group in Information Retrieval Conerenc., 1998, pp. 335–336

[20] PCC (Pearson Correlation Coefficient) http://en.wikipedia.org/wiki/Pearson_product-moment_correlation_coefficient

[21] L. Antiqueris, O. N. Oliveira, L. F. Costa and M. G. Nunes, "A complex network approach to text

summarization," Information Sciences, vol. 175, no.5, pp. 297–327, February 2009.

[22] V. Qazvinian and D. R. Radev, "Scientific paper summarization using citation summary networks," in Proceedings 17th Computational LinguisticsConference, 2008, pp. 689–696.

[23] D.R. Radev and K.R. McKeown, "Generating Natural Language Summaries from Multiple Online Sources," Computational Linguistics, Vol. 24, No. 3, 1998, pp. 469-500.

[24] ABM Sawkat Ali, Ben Pang and Kevin Tackle, "Rule Based Base Classifier for bagging Algorithm", Proceedings of the 2008 International Conference on Data Mining (DMIN2008), 14-17 July, 2008 Las Vegas, USA : CSREA Press, Pages 26-29.

[25] Endres-Niggemeyer and Hobbs, "Summarizing Text for Intelligent Communication Symposium",Dagstuhl Seminar,Dagstuhl, Germany, 1993.

[26] Fei Liu, Feifan Liu, and Yang Liu, "A Supervised Framework for Keyword Extraction from Meeting Transcripts", IEEE Transactions on Audio, Speech and Language Processing, Vol. 19, No. 3, pp. 538-548, March 2011 (IEEE TASL).

[27] Xiaoyan Cai, and Wenjie Li, "A Context-Based Word Indexing Model for Document Summarization", IEEE Transactions on Knowledge And Data Engineering, Vol. 25, No.8, August2013.