STUDY ON TEXT SUMMARIZATION USING EXTRACTIVE METHODS

S.Mohamed Saleem[1], R.Krithiga[2], S.K.Rani[3], S.Celin Sindhya[4]

Abstract- Text summarization is a challenging problem nowadays, due to large amount of information is developed and became more widespread. Text summarization is an inventing source text into a shorter version to preserve the information contents and overall meaning. Summarization is divided into extractive and abstractive. In this paper we focus on extractive methods in text summarization. It's a process of choosing the important sentence and the paragraph from original source text and concatenating them into a shorter form. The techniques involved here are text summarization with neural network, fuzzy logic, regression, Graph based method, machine learning and SR ranking.

***Index Terms*— Text summarization, Extractive method, Neural networks, fuzzy logics, regressions, Graph based, LDA based approach and SR ranking..**

I. INTRODUCTION

Text summarization has become an important and timely tool for assisting and interpreting text information in today's fast-growing information world. Text summarization is the process of automatically creating a compressed version of a given document preserving its information content. Automatic document summarization is an important research area in natural language processing (NLP)[2]. Natural language processing (NLP) is a field of computer science, artificial intelligence and machine learning with the interactions between computers and human language. The use of World Wide Web and many sources like Google, Yahoo! surfing also increases due to this the problem of overloading information also increases. The process of text summarization can be decomposed into three phases: analysis, transformation, and synthesis. The analysis phase analyzes the input text and selects a few salient attributes. The transformation phase transforms the results of the analysis into a summary representation. Finally, the synthesis phase takes the summary representation, and produces an appropriate summary corresponding to users' needs.

Text summarization can be categorized into two approaches: extractive and abstractive. Extractive summarization methods simplify the problem of summarization into the problem of selecting a representative subset of the sentences in the original documents. Abstractive summarization may compose novel sentences, unseen in the original sources. However, abstractive approaches require deep NLP [1] such as semantic representation, inference and natural language generation, which have yet to reach a mature stage now a day. The main aim of summarization is to create summary which provides minimum redundancy, maximum relevancy and co referent object of same topic of summary. Extractive summaries are formulated by extracting key text segments (sentences or passages) from the text, based on statistical analysis of individual or mixed surface level attributes such as word/phrase frequency, location or cue words to locate the sentences to be extracted. The "most important" content is treated as the "most frequent" or the "most favorably positioned" content. Such an approach thus avoids any efforts on deep text understanding[3]. They are conceptually simple, easy to implement.

II ATTRIBUTES FOR EXTRACTIVE TEXT SUMMARIZATION

Text summarization consists of following attributes to implement in final summary.

1. Sentence Position Attribute:

Usually first and last sentence of first and last paragraph of a text document are more important and are having greater chances to be included in summary.

2. Sentence Distance End To End Attribute:

End to end attribute with very large and very short sentences are not included.

3. Proper Noun Attribute:

Proper noun is name of a person, place and concept etc. Sentences containing proper nouns are having greater chances for including in summary.

4. Upper-Case Word Attribute:

Sentences containing acronyms or proper names are included.

5. Keyword Attribute:

Content words or Keywords are usually nouns and determined using tf × idf measure. Sentences having keywords are of greater chances to be included in summary.

ISSN: 2278 – 7798

Text summarization has become an important and timely tool for assisting and interpreting text information in today's fast-growing information world. Text summarization is the process of automatically creating a compressed version of a given document preserving its information content. Automatic document summarization is an important research area in natural language processing (NLP)[2]. Natural language processing (NLP) is a field of computer science, artificial intelligence and machine learning with the interactions between computers and human language. The use of World Wide Web and many sources like Google, Yahoo! surfing also increases due to this the problem of overloading information also increases. The process of text summarization can be decomposed into three phases: analysis, transformation, and synthesis. The analysis phase analyzes the input text and selects a few salient attributes. The transformation phase transforms the results of the analysis into a summary representation. Finally, the synthesis phase takes the summary representation, and produces an appropriate summary corresponding to users' needs.

Text summarization can be categorized into two approaches: extractive and abstractive. Extractive summarization methods simplify the problem of summarization into the problem of selecting a representative subset of the sentences in the original documents. Abstractive summarization may compose novel sentences, unseen in the original sources. However, abstractive approaches require deep NLP [1] such as semantic representation, inference and natural language generation, which have yet to reach a mature stage now a day. The main aim of summarization is to create summary which provides minimum redundancy, maximum relevancy and co referent object of same topic of summary. Extractive summaries are formulated by extracting key text segments (sentences or passages) from the text, based on statistical analysis of individual or mixed surface level attributes such as word/phrase frequency, location or cue words to locate the sentences to be extracted. The "most important" content is treated as the "most frequent" or the "most favorably positioned" content. Such an approach thus avoids any efforts on deep text understanding[3]. They are conceptually simple, easy to implement.

II ATTRIBUTES FOR EXTRACTIVE TEXT SUMMARIZATION

Text summarization consists of following attributes to implement in final summary.

1. Sentence Position Attribute:

Usually first and last sentence of first and last paragraph of a text document are more important and are having greater chances to be included in summary.

2. Sentence Distance End To End Attribute:

End to end attribute with very large and very short sentences are not included.

3. Proper Noun Attribute:

Proper noun is name of a person, place and concept etc. Sentences containing proper nouns are having greater chances for including in summary.

4. Upper-Case Word Attribute:

Sentences containing acronyms or proper names are included.

5. Keyword Attribute:

Content words or Keywords are usually nouns and determined using tf × idf measure. Sentences having keywords are of greater chances to be included in summary.

6. Typeset Based Attribute:

Sentences containing words appearing in upper case, bold, italics or Underlined fonts are usually more important.

7. Pronouns:

Pronouns such as "she, they, it" cannot be included in summary unless they are expanded into corresponding nouns.

8. Influenced Word Attribute:

If a word appearing in a sentence is from biased/influenced word list, then that sentence is important. Biased word list is previously defined and may contain domain specific words.

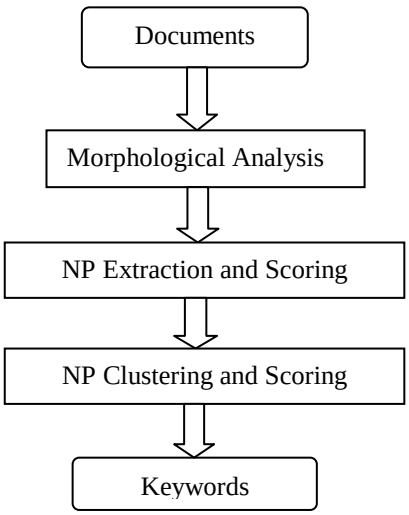

Figure1. Keyword extraction method

III EXTRACTIVE SUMMARIZATION TECHNIQUES

Extractive Summarization is the approach of concatenating extracts taken from a collection of return texts into a summary. There are three phases in our process: neural network training, feature fusion, and sentence selection. The first step involves training a neural network to recognize the type of sentences that should be included in the summary. The second step, feature fusion, prunes the neural network and collapses the hidden layer unit activations into discrete values with frequencies [4]. This step generalizes the important features that must exist in the summary sentences by fusing the features and finding trends in the summary sentences. The third step, sentence selection, uses the modified neural network to filter the text and to select only the highly ranked sentences. This step controls the selection of the summary

ISSN: 2278 – 7798

sentences in terms of their importance. These three steps are explained in detail in the next three sections.

a). Neural Network

Neural Network based approach - A neural network is trained on a corpus of documents. The neural network is then modified, through feature fusion, to produce a summary of highly ranked sentences in the document [5]. Through feature fusion, the network discovers the importance (and unimportance) of various features used to determine the summary-worthiness of each sentence. The input to the neural network can be either real or binary vectors.

The first phase of the process involves training the neural networks to learn the types of sentences that should be included in the summary. This is accomplished by training the network with sentences in several test paragraphs where each sentence is identified as to whether it should be included in the summary or not. This is done by a human reader. The neural network learns the patterns inherent in sentences that should be included in the summary and those that should not be included. We use a three-layered feed forward neural network, which has been proven to be a universal function approximate. It can discover the patterns and approximate the inherent function of any data to an accuracy of 100% as long as there are no contradictions in the data set. Our neural network consists of seven input-layer neurons, six hidden-layer neurons, and one output-layer neuron. We use a conjugate gradient method where the energy function is a combination of error function and a penalty function [7]. The goal of training is to search for the global minima of the energy function. The addition of the penalty function drives the associated weights of unnecessary connections to very small values while strengthening the rest of the connections. Therefore, the unnecessary connections and neurons can be pruned without affecting the performance of the network.

b). Feature Fusion

Once the network has learned the features that must exist in summary sentences, we need to discover the trends and relationships among the features that are inherent in the majority of sentences. This is accomplished by the feature fusion phase, which consists of two steps:

1) Eliminating uncommon features.

2) Collapsing the effects of common features.

Since the neural network is trained using a penalty function, the connections having very small weights after training can be pruned without affecting the performance of the network. As a result, any input or hidden layer neuron having no emanating connections can be safely removed from the network [7]. In addition, any hidden layer neuron having no abutting connections can be removed. This corresponds to eliminating uncommon features from the network. The hidden layer activation values for each hidden layer neuron are clustered utilizing an adaptive clustering technique. Each cluster is identified by its centroid and frequency. The activation value of each hidden layer neuron is replaced by the centroid of the cluster, which the activation value belongs to. This corresponds to collapsing the effects of common features. The combination of these two steps corresponds to generalizing the effects of features, as a whole, and providing control parameters for sentence ranking.

c). Cluster based method

Documents are usually written such that they address different topics one after the other in an organized manner. They are normally broken up explicitly or implicitly into sections. This organization applies even to summaries of documents. It is intuitive to think that summaries should address different "themes" appearing in the documents. Some summarizers incorporate this aspect through clustering. If the document collection for which summary is being produced is of totally different topics, document clustering becomes almost essential to generate a meaningful summary [9]. Documents are represented using term frequency inverse document frequency (TF-IDF) of scores of words. Term frequency used in this context is the average number of occurrences (per document) over the cluster. IDF value is computed based on the entire corpus. The summarizer takes already clustered documents as input. Each cluster is considered a theme. The theme is represented by words with top ranking term frequency, inverse document frequency (TF-IDF) scores in that cluster. Sentence selection is based on similarity of the sentences to the theme of the cluster Ci. The next factor that is considered for sentence selection is the location of the sentence in the document (Li) [13]. In the context of newswire articles, the closer to the beginning a sentence appears, the higher its weight age for inclusion in summary. The last factor that increases the score of a sentence is its similarity to the first sentence in the document to which it belongs (Fi).

The overall score (Si) of a sentence i is a weighted sum of the above three factors: Si =W1 *Ci + W2 *Fi+ W3 *Li where Si is the score of sentence Ci,, Fi are the scores of the sentence i based on the similarity to theme of cluster and first sentence of the document it belongs to respectively. Li is the score of the sentence based on its location in the document. w1, w2 andw3 are the weights for linear combination of the three scores.

d). Fuzzy Logic

This method considers each characteristic of a text such as sentence length, similarity to little, similarity to key word and etc. as the input of fuzzy system. Then, it enters all the rules needed for summarization, in the knowledge base of system. Afterward, a value from zero to one is obtained for each sentence in the output based on sentence characteristics and the available rules in the knowledge base [8]. The obtained value in the output determines the degree of the importance of the sentence in the final summary. The input membership function for each feature is divided into three membership functions which are composed of insignificant values (low L), very low (VL), medium (M), significant values (High h) and very high (VH). The important sentences are extracted using IF-THEN rules according to the feature criteria[14]. The obtained value in the output determines the degree of importance of the sentence in the final summary.

$$f(m,x,y,z) = high\left(low\left(\frac{m-x}{y-x}, \frac{z-m}{z-y}\right), 0\right)$$

We derived the new equations with very high accurate result in summarization.

ISSN: 2278 – 7798

f). Ranking Approach

Ranking

Based Approach usually provides the higher ranked sentences into the summary. Ranking algorithms extracts the rank sentences and merges the all rank sentences and generate the summary. Basically, it applies ranking algorithm, extracts rank sentences and generate a summary. SR-Rank algorithm is a type of graph based algorithm. Firstly, assign the sentences and get the semantic roles, and then apply a novel SR-Rank algorithm. SR-Rank algorithm simultaneously ranks the sentences and semantic roles; it extracts the most important sentences from a document [6]. A graph based SR-Rank algorithm rank all sentences nodes with the help of other types of nodes in the heterogeneous graph. Here three kinds of graphs are explained as graph-cluster, graph-scan and basic graph. So in this paper, three kinds of graphs are generated as SR-Rank, SR-Rank-span and SR-Rank-cluster. Experimental results are given on two DUC datasets which shows that SR-Rank algorithm surpasses few baselines and semantic role information is validated which is very helpful for multi-document summarization.

The proposed sentence ranking algorithm: **ispread Rank**

The proposed sentence ranking algorithm, ispread Rank, which is the major contribution of this work, borrows many concepts from the spreading activation theory, and is designed to rank the importance of sentences for extraction-based summarization Spreading activation was originally developed in psychology to explain the cognitive process of human comprehension through semantic memory. The theory states that human long-term memory is structured as an associative network in which similar memory units have strong connections and dissimilar units have none or weak connections [6]. Accordingly memory retrieval is viewed as searching across the network by activating a set of source nodes with stimuli (or energy), then iteratively propagating the energy in parallel along links through the network to other connected nodes to discover more related nodes with hidden information.

Example for ispread ranking

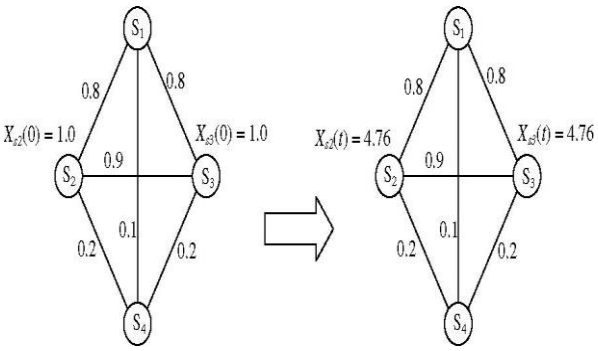

Fig 2 ispread ranking

g). Regression

1.The SVR model of sentence importance

A Support Vector Regression (SVR) model aims to learn a function *f* : R*n* 7! R, which will be used to predict the value of a variable *y* 2 R given a feature vector *X* 2 R*n*. In particular, given *l* training instances (*X*1, *y*1), . . (*Xl* , *yl*), an SVR model is learnt by solving the following optimization problem; *W* is a vector of feature weights; is a function that maps feature vectors to a new vector space of higher dimensionality to allow non-linear functions to be learnt in the original space; *C* > 0 and _ > 0 are given.

The goal is to learn a linear (in the new space) function, whose prediction (value) *WT* · _(*Xi*)+*w*0 for each training instance *Xi* will not to be farther than _ from the target (correct)value *yi* . Since this is not always feasible, two slack variables _*i* and __*i* are used to measure the prediction's error above or below the target *yi*. The objective (9) jointly minimizes the total prediction error and k*W*k, to avoid over fitting.

We now present the experiments that we performed, starting from the datasets.

Example Datasets

We used the datasets of TAC 2008, TAC 2009,TAC 2010,TAC2011. Each dataset contains document clusters. Each cluster contains documents relevant to a query (a question or topic description), which is also given. For each cluster, a summary not exceeding a maximum allowed length has to be produced, so that the summary will provide an answer to the corresponding query. Multiple reference (gold, human-authored) summaries are also provided per cluster[19]. Table 1 provides more information on the datasets we used. For our experiments, we extracted all the sentences from the documents of each cluster, discarding sentences shorter than or equal to 7 words.

$$\min_{w,b,\varepsilon,\varepsilon^*} \frac{1}{2}\|w\|^2 + c\sum_{l=1}^{1}\varepsilon_{\iota} + c\sum_{c=1}^{1}\varepsilon_{\iota}^*$$

We also applied a small set of cleanup rules to remove unnecessary formatting tags.

ISSN: 2278 – 7798

Data set	Documents per Cluster	Clusters	Reference Summaries	Word Limits
TAC2008	15	58	5	150
TAC2009	20	55	4	200
TAC2010	25	60	5	200
TAC2011	30	60	5	150

2.SVR-based sentence scoring

The features used in our SVR-based system are similar to those in our TAC2008 system.

- Word-based Feature.
- Phrase-based Name Entity Feature.
- Semantic-based Word Net Feature.
- Centroid Feature.
- Named Entity Number Feature.
- Sentence Position Feature.

h). LDA Based Approach:

Latent Dirichlet Allocation (LDA) has been recently introduced for generating corpus topics and applied to sentence based multi-document summarization method [4]. It is not compulsion to estimate topics are of equal importance or relevance collection of sentence or significance themes. Some of the topics can contain different theme and irrelevancy so for this LDA is used for topic models .The paper Mixture of Topic Model for Multi-document Summarization based on Titled-LDA algorithm which models title and content of documents then mixes them by asymmetric method. Here mixture weights for topics to be determined. Topic model illustrate an idea how documents can be modelled in the form of probability distributions over words in a document. Titled-LDA divided into three tasks:

First, distribution of topic is done over the topic which is sampled from a Dirichlet distribution. Second, a single topic is selected according to this distribution for each word in the document. Finally, each word is sampled from a polynomial distribution over words which are defined in sampled topic. And get the title information and the content information in appropriate way which is helpful in performance of Summarization [18]. The experimental results shows good result by proposing a new algorithm compared to other algorithm on.

The paper Latent Dirichlet Allocation and Singular Value Decomposition based on Multi-Document Summarization proposed LDA-SVD (Latent Dirichlet Allocation and Singular Value Decomposition) Multi-Document Summarization algorithm. As multi-document summarization covers different events from the sentences in the documents and LDA break down that document into different topics or events. But here orthogonal vector is required to reduce common information content and it provides association of sentences. SVM is used to get the orthogonal representations of vectors and also can represents in the form of sentence orthogonal. LDA finds different topics in the documents whereas SVD finds the sentences which are best represent these topics. Finally, evaluate the algorithms on DUC 2002 CORPUS multi- document summarization tasks using the ROUGE evaluator to evaluate the summaries. This algorithm gives better results for ROUGE-1 recall measures in comparison of DUC 2002.In this. LDA-SVD Multi-Document summarization algorithm is better than GISTEXTER and WSRSE.
It gives better results than traditional method.

i). Query based extractive text summarization:

In query based text summarization system, the sentences in a given document are scored based on the frequency counts of terms (words or phrases). The sentences containing the query phrases are given higher scores than the ones containing single query words[11]. Then, the sentences with highest scores are incorporated into the output summary together with their structural context. Portions of text may be extracted from different sections or subsections. The resulting summary is the union of such extracts. The number of extracted sentences and the extent to which their context is displayed depends on the summary frame size which is fixed to the size of the screen that can be seen without scrolling. In the sentence extraction algorithm, whenever a sentence is selected for the inclusion in the summary, some of the headings in that context are also selected [9]. The query based sentence extraction algorithm is as follows:

Algorithm:

1: Rank all the sentences according to their score.
2: Add the main title of the document to the summary.
3: Add the first level-1 heading to the summary.
4: While (summary size limit not exceeded)
5: Add the next highest scored sentence.
6: Add the structural context of the sentence (if any and not already included in the summary)
7: Add the highest level heading above the extracted text (call this heading h).
8: Add the heading before h in the same level.
9: Add the heading after h in the same level.
10: Repeat steps 7, 8 and 9 for the next highest level headings.
11: End while An another query-specific summarization method views a document as a set of interconnected text fragments (passages) and focuses on keyword queries, since

ISSN: 2278 – 7798

keyword search is the most popular information discovery method on documents, because of its power and ease of use.

Firstly, at the preprocessing stage, it adds structure to every document, which can then be viewed as a labeled, weighted graph, called the document graph. Then, at query time, given a set of keywords, it performs keyword proximity search on the document graphs to discover how the keywords are associated in the document graphs. For each document its summary is the minimum spanning tree on the corresponding document graph that contains all the keywords. In query-specific opinion summarization system (QOS)[12].When input an opinion question, the system returns a summary with relevance to the opinion and target described by the question. The system has several modules to be able to do this: a question analysis and query reformulation module, a latent semantic indexing based sentence scoring module, a sentence polarity detection module, and a redundancy removal module.

Bayesian summarization (BAYESUM) is a model for sentence extraction in query-focused summarization. BAYESUM leverages the common case in which multiple documents are relevant to a single query. Using these documents as reinforcement for query terms, BAYESUM is not afflicted by the paucity of information in short queries[16]. For a collection of D documents and Q queries, assume a D × Q binary matrix r, where rdq = 1 if an only if document d is relevant to query q. In multi document summarization, rdq will be 1 exactly when d is in the document set corresponding to query q.

j). Graph based approach:

Our enhancements rest on the foundations of graph based approaches already developed by Erkan and Radev [3] namely LexRank and Continuous LexRank. We have introduced two enhancements to the above schemes namely discounting technique and incorporation of position weight factor.

1.Discounting

Discounting technique envisages that once a sentence is selected by any one of the methods then the corresponding row and column values of the matrix are set to zero. The next sentence is selected based on the contributions made by the remaining 'n-1' sentences only[5]. Thus when we use discounting technique to any of the methods proposed, the sentences were picked up desired on the target ratio, provided the adjacency matrix is modified as stipulated. The idea behind discounting technique is that once the sentence is selected, the chance for repetition of information in the succeeding sentences is minimized [21]. The information will not be duplicated and the summary will be cohesive and meaningful in nature.

2.Position Weight

The location of a sentence in a document plays a significant part in determining the importance of a sentence. In the graph based approach, for multi document summarization, importance to position of the sentence can be given in a way by giving preference to sentences that occurs earlier out of the two documents considered[22]. Consider an example to illustrate the situation clearly. For instance if document1 has 10 sentences and document 2 has 5 sentences and if there is tie in selecting the first sentence, then we select sentence1 from document 1(since it gets a weight of 1/10=0.1) rather than sentence1 from document 2 (which gets a weight of 1/5=0.20).

3.Position weight factor is given by:

$$P_{ft} = \text{gama} + \text{beta}^{i-\alpha-I}$$

Where gama and beta are design parameters. $\alpha=0$ for the sentences of the first document, $\alpha=n1$ for the sentences of the second document and $\alpha=n1+n2$ for the sentences of the third document etc., ni being the number of sentences in the document. Thus position weight of any sentence is allocated based on its relative position in the document in which it is present. In order to clearly distinguish between various methods, we call LexRank methods with the incorporation of discounting and position weight as Sentence Rank (SR) methods. Combines position weight and discounting technique together with the basic schemes proposed by Erkan and Radev [3].

- **Method 1**: LexRank (threshold).
- **Method 2**: Continuous LexRank.
- **Method 3**: Discounted LexRank (threshold).
- **Method 4**: Discounted Continuous LexRank.
- **Method 5**: Sentence Rank (threshold).
- **Method 6**: Sentence Rank (continuous).

For methods 1 to 6 several investigations were made relating to threshold, damping factor, direction of graph and impact of self weight. While it is recommended to adopt a damping factor in the interval 0.1 to 0.2, we have adopted an optimal damping factor of 0.10 . We have adopted undirected graphs and threshold of 0.10 for threshold methods.

IV. CONCLUSIONS

This survey paper is concentrating on extractive summarization methods. An extractive summary is selection of important sentences from the original text. The importance of sentences is decided based on statistical and linguistic features of sentences. Many variations of the extractive approach have been tried in the last 12 years. However, it is hard to say how much greater interpretive sophistication, at sentence or text level, contributes to performance. Without the use of NLP, the generated summary may suffer from lack of cohesion and semantics. Deciding proper weights of individual features is very important as quality of final summary is depending on it. We should devote more time in deciding feature weights. The biggest challenge for text summarization is to summarize content from a number of textual and semi structured sources, including databases and web pages, in the right way (language, format, size, time) for a specific user. The text summarization software should produce the effective summary in less time and with least redundancy. Summaries can be evaluated using intrinsic or extrinsic measures. While intrinsic methods attempt to measure summary quality using human evaluation and extrinsic methods measure the same through a task based

ISSN: 2278 – 7798

performance measure such information retrieval- oriented task.

References

[1] Cretu B., Chen Z., Uchimoto T., and Miya K., "Automatic Summarizing Based on Sentence Extraction: A Statistical Approach," International Journal of Applied Electromagnetics and Mechanics , vol. 13, no. 1- 4, pp. 19-23, 2002.

[2] Edmundson H., "New Methods in Automatic Extracting," Journal of the ACM , vol. 16, no. 2, pp. 264-285, 1969.

[3] Erkan G. and Radev D., "LexRank: Graph-Based Lexical Centrality as Salience in Text Summarization," Journal of Artificial Intelligence Research , vol. 22, pp. 457-479, 2004.

[4] Lin C. and Hovy E., "Automatic Evaluation of Summaries Using N-gram Co-Occurrence Statistics," in Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics on Human Language , USA, pp. 71-78, 2003.

[5] Litvak M. and Last M., "Graph-Based Keyword Extraction for Single-Document Summarization," in Proceedings of the Workshop on Multi-Source Multilingual Information Extraction and Summarization Coling , USA, pp. 17-24, 2008.

[6] Liu Y., Wang X., Zhang J., and Xu H., "Personalized PageRank Based Multi-document Summarization," in Proceedings of IEEE International Workshop on Semantic Computing and Systems , Huangshan, pp. 169-173, 2008.

[7] Karel Jezek and Josef Steinberger, "Automatic Text summarization", Vaclav Snasel (Ed.): Znalosti 2008, pp.1- 12, ISBN 978-80-227-2827-0, FIIT STU Brarislava, Ustav Informatiky a softveroveho inzinierstva, 2008.

[8] Farshad Kyoomarsi, Hamid Khosravi, Esfandiar Eslami and Pooya Khosravyan Dehkordy, "Optimizing Text Summarization Based on Fuzzy Logic", In proceedings of Seventh IEEE/ACIS International Conference on Computer and Information Science, IEEE, University of Shahid Bahonar Kerman, UK, 347-352, 2008.

[9] Joel larocca Neto, Alex A. Freitas and Celso A.A.Kaestner, "Automatic Text Summarization using a Machine Learning Approach", Book: Advances in Artificial Intelligence: Lecture Notes in computer science, Springer Berlin / Heidelberg, Vol 2507/2002, 205-215, 2002.

[10] Weiguo Fan, Linda Wallace, Stephanie Rich, and Zhongju Zhang, "Tapping into the Power of Text Mining", Journal of ACM, Blacksburg, 2005.

[11] S. Jones, M. Staveley, Phrasier: A system for interactive document retrieval using Key phrases, In: proceedings of SIGIR, 1999, Berkeley, CA

[12] C. Gutwin, G. Paynter, I. Witten, C. Nevill-Manning, E. Frank, Improving browsing in digital libraries with keyphrase indexes, Journal of Decision Support Systems, 2003, 27(1-2), 81-104

[13] B. Kosovac, D. J. Vanier, T. M. Froese, Use of key phrase extraction software for creation of an AEC/FM thesaurus, Journal of Information Technology in Construction, 2000, 25-36

[14] S.Jonse, M. Mahoui, Hierarchical document clustering using automatically extracted keyphrase, In proceedings of the third international Asian conference on digital libraries, 2000, Seoul, Korea. pp. 113-20 .

[15] Ha Nguyen Thi Thu, "An Optimization Text Summarization Method Based on Naïve Bayes and Topic Word for Single Syllable Language", Applied Mathematical Sciences, Vol. 8, 2014no. 3, 99 – 115, HIKARI Ltd, www.m-ikari.com http://dx.doi.org/10.12988/ams.2014.36319

[16] Otterbacher J., Erkan G. and Radev D. 2005. Using Random Walks for Question-focused Sentence Retrieval. In Proceedings of HLT/EMNLP 2005.

[17] Qazvinian V. and Radev D. R. 2008. Scientific paper summarization using citation summary networks. In Proceedings of COLING2008.

[18] Sun P., Lee J.H., Kim D.H., and Ahn C.M. 2007. Multi-Document Using Weighted Similarity Between Topic and Clustering-Based Non- negative Semantic Feature. APWeb/WAIM 2007.

[19] M. G. H. Omran, A. P. Engelbrecht, and A. Salman, "An overview of clustering methods," Intelligent Data A sis, Vol. 11, No. 6, pp. 583–605, 2007.

[20] K. M. Hammouda and M. S. Kamel, "Efficient phrase- based document indexing for web document clustering," IEEE Transaction MarchVol. 16, No. 10, pp. 1279–1296, 2004.

[21] R. L. Cilibrasi and P. M. B. Vitanyi, "The google similarity distance," IEEE Transaction on Knowledge and Data Engineering, Vol. 19, No. 3, pp. 370–383, 2007.

[22] Lin Y., "ROUGE: Recall-Oriented Understudy for Gisting Evaluation," available at: http://www.isi.edu/_cyl/ROUGE/, last visited 2003.

ISSN: 2278 – 7798