An Efficient Method based on Lexical Chains for Automatic Text Summarization

Shweta Saxena
Department of information Technology
Jaipur engineering college
and research center
Jaipur, Rajasthan, India

Akash Saxena, PhD
Department of Computer Engineering
CompuCom Institute of
technology and management
Jaipur, Rajasthan, India

ABSTRACT
Automatic Text Summarization is an interesting topic for research. Still it is growing on. Increment of the data is exponentially growing on and it becomes too much difficult to find out the correct or relevant data in huge amount of data. So it becomes important for researchers to use it for efficient retrieval of information. Hence Text Summarization plays an important role for this problem. Summarization gives the short version for the text document which contains the main context of the document. Summarization can be classified into two categories: Extractive and Abstractive. This paper presents the extractive summary using lexical chaining approach. Lexical chains are created by using Knowledge based database i.e. Wordnet. This paper compares results with the traditional methods and gives better results.

Keywords
Extractive Summarization, Lexical chains, Semantic relations, Text Summarization (TS), Wordnet.

1. INTRODUCTION
In the early age, the text document can be easily retrieved, but as the time grows, the data on the web also increased. Now the text documents also exponentially increased. It takes more time consumption for retrieving the useful information from the big data. Hence there is a requirement of a tool or system that automatically retrieves, summarizes and bifurcate the text documents as per user's requirement. Text Summarization is one resolution for this problem which summarizes the electronic data. Automatic Text summarization can be categorized into two categories: Extractive and Abstractive. In Extractive Summarization, the sentences in the generated summary are the same as in the original document. The proposed work is also concentrated on Extractive Summarization with single document. In this paper the approach is based on lexical chains. The lexical chains have been made by finding the relations among the words in the sentences. These relations are called semantic relations which provide the lexical cohesion among the words.

Lexical cohesion works as the adhesive substance which sticks the sentences and words. When lexical cohesion goes beyond the two words then it forms lexical chains. This paper presents generation of the extractive summary using lexical chains.

Lexical chains give a visual representation of the words that are related to each other in the text.

This paper presents a short overview of previous methods based on lexical chains. Literature Review is discussed in section 2. Text summarization is defined in section 2.1. The proposed algorithm is discussed in section 3. Evaluation of Summary is defined in section 4. Comparison of results with some previous methods is illustrated in section 5. Limitation of work is defined in section 6. Finally, section 7 concludes the paper.

2. LITERATURE REVIEW
2.1. Text Summarization
Text Summarization is the process to shorten the document with relevant sentences for generation of summary. In early of 1950's the summary had been generating by using the static feature of the text like: term frequency Luhn [1], sentence position, cue phrases Edmondson [2]. These are the static feature of the text in which no need of any other source for generating summary. Hence the research has grown. Researchers moved towards using other knowledge sources for generation of summary. The other sources like: thesaurus, Wordnet dictionary etc for finding semantic relationship among the words.

2.2. Lexical chains
The first concept in knowledge sources, Halliday and Hasan [3] gave the concept of the Lexical cohesion means there can be some relation between the two sentences by co-reference, ellipsis, conjunctions etc. Then concept of lexical cohesion is used by Morris and Hirst [4]. They gave the first concept of lexical chains by using Roget's Thesaurus. But they didn't implement it due to lack of machine readable form of Thesaurus. Hence Hirst and St-Onge [5] gave the concept of lexical chain and correction of malapropisms. They gave the concept of relations between the words basis of the distance between the words.

The next work is done Barzilay and Elhadad [6], he gave many interpretations of the words for making lexical chain. This method is very efficient which is followed by researchers for their research. The strong chains can be finding out by applying formulas. Lexical chains made by using Wordnet dictionary. But the drawback of [6] is more interpretations decrease the system efficiency. For removing the drawback of exponential time and space in [6], H.Gregory Silber, Kathleen F.McCoy [7] gave a concept of "meta chains". The "meta chains" contains many lexical chains row wise. They modified the noun database of the Wordnet and made an array of categories. Hence it reduces the exponential time and space. A graph based method is given by O. Medelyan [8], which gave the concept of the nodes and edges. Where nodes are the words and edge is the semantic relation between the words. And the weak graph is splitted into some strong graphs. For graph clustering, the Chinese whispers algorithm is used [9].

There are so many research has been done. Some research had done on WSD (Word sense disambiguation) [10], [11]. The word sense disambiguation means to disambiguate the words,

means in that text what is the exact sense of the word. Those researches give many contributions for making the lexical chains. Some methods have been used for user driven topic based extraction of sentences using lexical chains. They used other features of the text like: Term frequency, sentence of length, location of the sentences etc [12]. Some research depends upon the finding cohesion and coherence among the sentences based on the lexical chains [13]. Lexical chains have their applications like to detect the emotions [14], event tracking [15], WSD etc. We need to find out more efficient method for automatic text summarization which gives better results. For checking the performance of our method we used some sets of the document files.

3. PROBLEM FORMULATION

The proposed work is the method that is implemeted and described here. It is based on the lexical and semantic relation among the words of the text. It works on lexical chains and used wordnet dictionary for finding semantic relations. And finally sentences, that are highly scored through formulas, result as summary. The Flow chart of proposed system is given in the Fig.1.

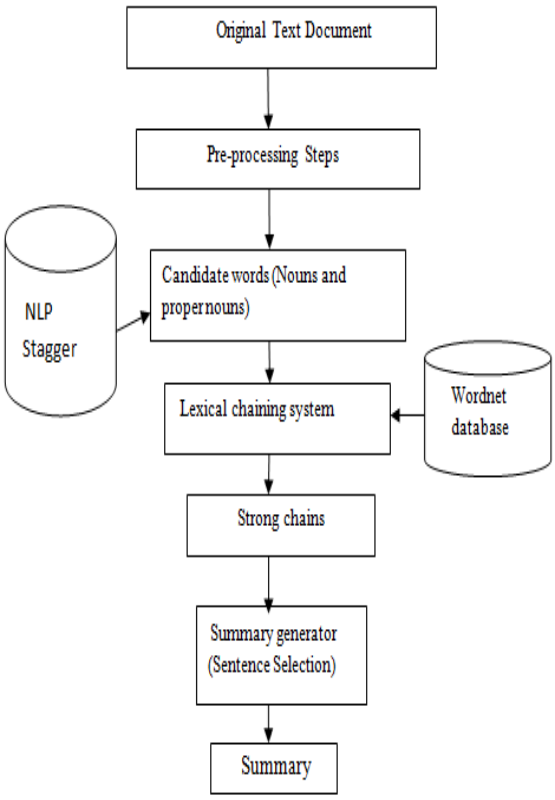

Fig 1: Architecture of proposed system

Details of the following steps:

3.1 Original Text Document

The text document should not contain images and tables. It only contains the text. The text document which needs to be summarized is called original text document. It works as the input in proposed text summarizer system. The file is in text file (.txt file).

3.2 Pre-Processing

The pre-processing step is called as the filtration of the valuable words in the text document. It includes some steps:

3.2.1 Text Segmentation

In text segmentation, the sentence has extracted through the period sign. Each and every sentence is separated in the form of many lines.

3.2.2 Stop words removal

The stop words are the words which are not much efficient in the sentences. If the stop words remove from the sentence then the meaning of sentence will not much effected. Stop words are like: is, am, the, about, different etc.

3.2.3 Tokenization

In tokenization the words which are remaining in each line will perform like tokens, which will lead to formation of lexical chains.

3.2.4 Stemming

The stemming of tokens is done through Wordnet dictionary by placing the original word. Like: If the token is "intelligence" then after stemming the word will be "intelligent".

3.2.5 POS tagging

In POS tagging the stemmed tokens will be tagged as Nouns and Pronouns. The tags are such as: NN (Nouns), NNP (Proper Nouns).

3.3 Candidate Words

This step has some set of the words which are nouns and proper nouns after tagging the words. After this, frequency of each candidate word is counted.

3.4 Lexical Chainer

The lexical chainer is the main module for our system. The chainer takes candidate words as input and then uses the Wordnet database for finding the relations among the words. The words have their senses and then the appropriate sense has added to the chain with any semantic relation (synonyms, merynyms, hypernyms, hyponyms etc.) with any word in the chain. If there is no chain is present, then the word will make a new chain. In this way lexical chainer has so many chains using candidate words.

In this module, Wordnet database is used, which gave the synsets of each word and if one synset is related to another then this relation is called hypernyms/hyponyms and merynyms.

3.5 Strong Chains

After making the lexical chains, it needs to score them. For Scoring, this algorithm applied some formulas. The formulas are:

Length (LC) = total of the number of the particular chain members (candidate words)

..(1)

The significance of the chain (LC) specifies that how randomly a lexical chain is present in the document:

$$\text{Sig (LC)} = \frac{\text{Length (LC)}}{\sum_{l \in D} length(l)} * \frac{\text{log2 Length (LC)}}{\sum_{l \in D} length(l)}$$

..(2)

Where Sig means significance of lexical chain in document, LC is lexical chain, D is the document; l is each chain in document D. After finding the significance of the chains, it has some numerical values for each chain. Now the Utility of

each chain will find. For Utility it needs to find relation, that the word "w" belongs to the chain "LC" or not.

Related (w, LC) = 1, if they are related

=0, if they are not related...(3)

Here the related means whether the word "w" has any semantic relation with the chain L (synonym, Hypernyms, hyponym, merynyms etc). So Utility specifies the contribution of the lexical chain in text document:

$$\text{Utility (LC, D)} = \text{Sig (LC)} * \sum_{\forall w \in D} related(w, LC) \quad ..(4)$$

After applying these three formulas on each chain, some numerical value is found which is used for finding the strong chains among them.

Score Chain (L) = AVG + 2 * STD. Dev .(5)

Where: AVG = average of scores of lexical chain (utility of each chain)

STD. Dev= Standard deviation of the utility scores of each lexical chain.

By applying (4) equation it has got a numerical value which would be comparing with the utility of each chain:

If Utility (LC) > Score Chain (L)

Then the lexical chain LC will consider as Strong chain.

Hence in this way the lexical chainer module generates many numbers of strong lexical chains.

3.6 Sentence Selection

The sentence selection is dependent upon the strong lexical chains. The members of the strong chains are the words which have their occurrences in the sentences. If a sentence contains any chain member of any strong chain then the summation of the number of the words (chain member) in the sentence will give a rank to the sentence. Hence the highly ranked sentences will be extracted as per the percentile of the text document. The sentences will be collected which are highly ranked. So these sentences are the sentences which are similar in the original document and in the same order.

4. SUMMARY EVALUATION

Evaluation is the most important work of any research. The summary has been evaluated by using the ROUGE tool. Rouge tool stands for Recall-Oriented Understudy for Gisting Evaluation [16]. This tool has some features by which human generated summary and the candidate (System generated) summary is evaluated. Precision, Recall and F- measure features of Rouge Tool are implemented.

4.1 Recall

The Recall can be defined as the efficiency of the approach of finding the relevant sentences form the document. Higher the Recall value, higher the efficient system in retrieving the accurate sentences.

$$\text{Recall} = \frac{\text{[Recovered Sentences]} - \text{[Applicable Sentences]}}{\text{[Applicable Sentences]}}$$

4.2 Precision

The Precision can be defined as the efficiency of the system in reducing the irrelevant sentences. Higher the precision value, higher the efficient system in reducing the irrelevant sentences.

$$\text{Precision} = \frac{\text{[Recovered Sentences]} - \text{[Applicable Sentences]}}{\text{[Recovered Sentences]}}$$

4.3 F- measure

F-measure is the weighted harmonic mean of the recall and precision.

$$\text{F-measure} = \frac{\text{[2 x Precision * recall]}}{\text{[Precision + recall]}}$$

5. EXPERIMENTAL RESULTS

The datasets has 10 text files which are related to news. The references (human generated) summaries of the text files have also given with that datasets. Some existing methods are implemented and then compare them with the proposed method, and got some better results.

On the basis of recall and precision value of the Rouge tool, the evaluation has done. The constant percentile of summary of the text files generated depends upon the length of the text file. Fewer contexts, more percentiles of text file and vice-versa. Summary generation is implemented on different percentile with different text files.

After evaluating on recall and precision, this paper also includes the graphical representation for evaluation.

Table no. 1 and table no. 2 shows the result of existing methods for text summarization. Table no. 3 shows the results of proposed method on the same text files and with the same compression ratio (% of summary).

Table 1: Recall and Precision of existing method 1 for text summarization

Text File Name	% of Summary	Recall of Existing Method 1	Precision of Existing Method 1
AP880911	28	0.43	0.69
AP880912	18	0.48	0.56
AP880915	13	0.55	0.43
AP880916	23	0.42	0.41
AP891018	21	0.58	0.51
LA102089	31	0.26	0.70
LA102489	23	0.50	0.30
WSJ880912	35	0.54	0.51
WSJ891019	40	0.31	0.30
WSJ8910190	22	0.36	0.59

Table 2: Recall and Precision of existing method 2 for text summarization.

Text File Name	% of Summary	Recall of Existing Method 1	Precision of Existing Method 2
AP880911	28	0.31	0.59
AP880912	18	0.31	0.50
AP880915	13	0.20	0.40
AP880916	23	0.45	0.45
AP891018	21	0.50	0.54
LA102089	31	0.32	0.47
LA102489	23	0.30	0.52
WSJ880912	35	0.30	0.70
WSJ891019	40	0.23	0.44
WSJ8910190	22	0.30	0.59

Table 3: Results of proposed methods for text summarization

Text File Name	% of Summary	Recall of Proposed Method	Precision of Proposed Method
AP880911	28	0.70	0.68
AP880912	18	0.67	0.43
AP880915	13	0.50	0.27
AP880916	23	0.48	0.45
AP891018	21	0.67	0.31
LA102089	31	0.35	0.82
LA102489	23	0.48	0.27
WSJ880912	35	0.67	0.54
WSJ891019	40	0.33	0.18
WSJ8910190	22	0.60	0.60

Figure 2 shows the graphical result of the method 1, in which the recall and precision are shown by the line graph. Similarly fig 3 and fig 4 shows the graphical result of the existing method 2 and the proposed method. The graphs are sketched on the constant percentile of summary.

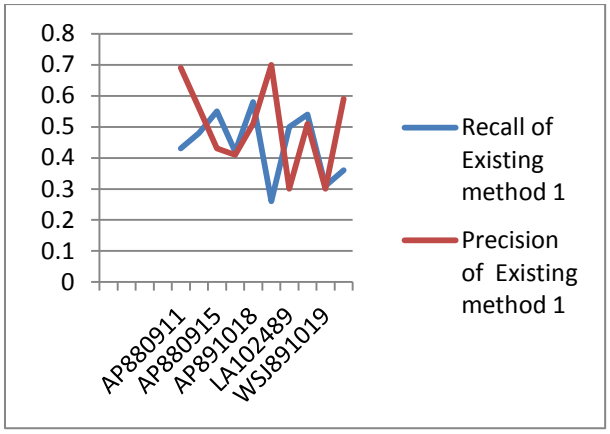

Fig 2: Graphical representation of recall and precision of existing method 1

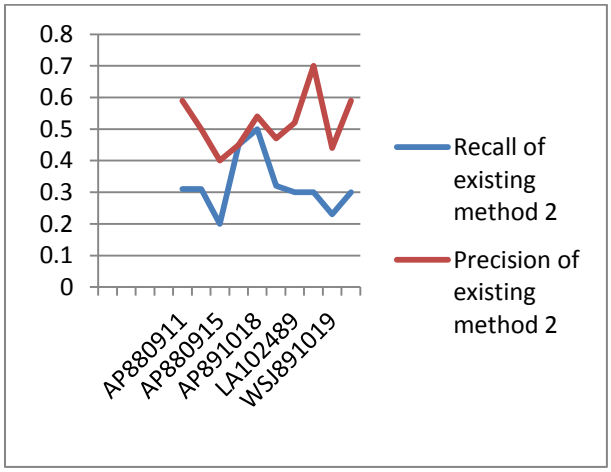

Fig 3: Graphical representation of recall and precision of existing method 2

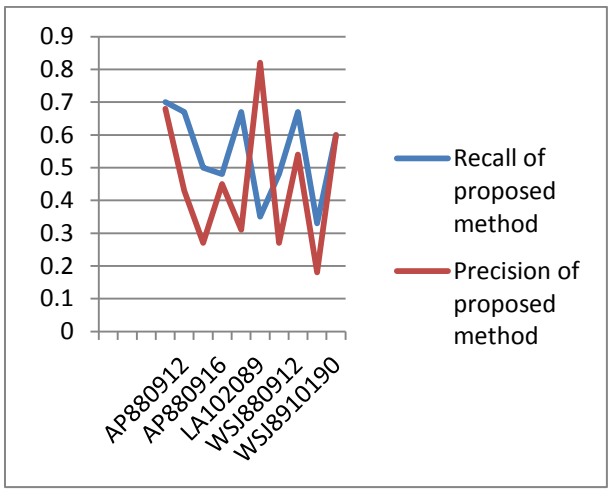

Fig 4: Graphical representation of recall and precision of proposed method

5.1 Comparison of Method's result

In this section, the graphical representation of the recall and precisions of existing methods and proposed method is shown. This would clearly prove that proposed method is efficient than existing methods. Figure 5 and 6 represents the pictorial view of recall and precision.

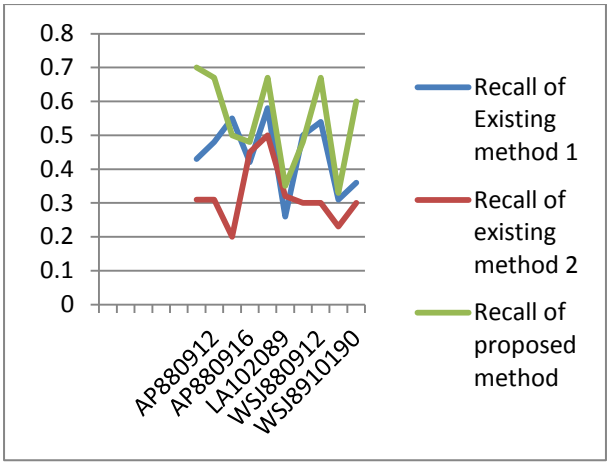

Fig 5: Comparison of recall of existing methods 1, 2 and proposed method.

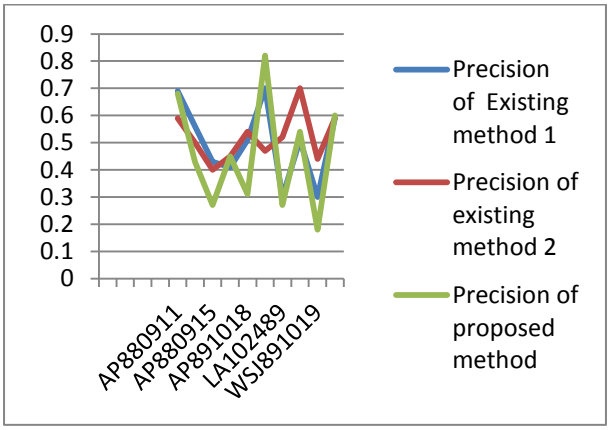

Fig 6: Comparison of precision of existing methods 1, 2 and proposed method.

From figure 5 and 6 this is clear that proposed method gave better result than existing method.

6. LIMITATION OF WORK

In proposed method there is still some limitations. First one is the sentence marker. The sentence marker is not up to the mark. It separates the sentence by the period (.) Sign, that is not much accurate. The second one is the processing time is little more. Proposed method takes some time for generating the summary after full procedure. It takes time due to scanning the Wordnet for lexical chain in pre- processing steps.

7. CONCLUSION

The above discussed method is compared with the two existing methods on the same files with same % of summary. These methods have been evaluated on recall and precision features. The recall factor is more important which tells that how much proposed method is significant in extracting the relevant or accurate sentences. Hence from the table and graphical representation, proposed method is significant. In some files proposed algorithm lacked. The results (summary) of proposed algorithm are evaluated with the human generated summary for each file. The future scope of Work is to generate more efficient tool or system which can extract the accurate coherent sentences and try to move towards abstractive text summarization.

8. ACKNOWLEDGEMENTS

Any work is not possible without any support, motivation, encouragement and without any proper guidance. I have no adequate words for showing my gratitude an indebtness to my guide, faculties and my friends, for proper guidance and worthy suggestions.

9. REFERENCES

[1] Luhn, H.P. 1958. The *automatic creation of literature abstracts.* IBM Journal of Research and Development, 2, pp.159–165.

[2] Edmondson, H.P. 1969. *New methods in automatic extracting*. Journal of the ACM, 16(2), pp. 264–285.

[3] Halliday and Hasan (1995). *Cohesion in English London:* Longman. pp. 591-595.

[4] Morris, J., and Hirst, G (1997) *Lexical cohesion computed by Thesaural relations as an Indicator of the structure of text*. Journal Computational Linguistics archive Volume 17 Issue1, March 1991 pp. 21-48.

[5] Hirst, G., and St-Onge, D. (1998) *Lexical chains as representation of context for the Detection and correction of malapropism.* In Fellbaum, C., ed., Wordnet: An Electronic *Lexical* Database and Some of Its Applications. Cambridge, MA: The MIT Press: 1998 pp. 305-332.

[6] Regina Barzilay and Michael Elhadad. (1997) *Using Lexical Chains for Text Summarization*, in Proceedings of the Intelligent Scalable Text Summarization Workshop (ISTS'97), ACL, Madrid, 1997. pp. 111-121

[7] Gregory Silber, Kathleen F. McCoy. (2002) *efficiently computed lexical chains as an Intermediate representation for automatic text Summarization* Journal Computational Linguistics - Summarization archive Volume 28 Issue 4, December 2002 pp. 487-496.

[8] Olena Medelyan, (2007) Computing Lexical Chains with Graph Clustering, Published in: Proceeding ACL '07 Proceedings of the 45th Annual Meeting of the ACL: Student Research Workshop. pp. 85-90.

[9] Chris Biemann (2011) Chinese Whispers - An Efficient Graph Clustering Algorithm and its Application to Natural Language Processing Problems. , Proceedings of the First Workshop on Graph Based Methods for Natural Language Processing, page 73-80. Stroudsburg, PA, USA, Association for Computational Linguistics.

[10] Michel Galley, Kathleen McKeown, (2003) Improving Word Sense Disambiguation in Lexical Chaining. Published in Proceedings of18th International Joint Conference on Artificial Intelligence (IJCAI03).

[11] Junpeng Chen, Juan Liu, Wei Yu, Peng Wu (2009), Combining Lexical Stability and Improved Lexical Chain for Unsupervised Word Sense Disambiguation Published in: Knowledge Acquisition and Modeling, 2009. KAM '09. Second International Symposium on (Volume: 1) pp.430-433

[12] Pankaj gupta, Vijay Shankar, Ishant Vats, (2011) Summarizing text by ranking text Units according to shallow linguistic features. Published in: Advanced Communication Technology (ICACT), 2011, 13th International Conference .pp. 1620-1625

[13] A.R.Kulkarni, S.S.Apte, (2014), an automatic text summarization using lexical cohesion and correlation of sentences. IJRET: International Journal of Research in Engineering and Technology Volume: 03 Issue: 06 pp. 285-292.

[14] M. Naveen Kumar, R.Suresh, 2012 Emotion Detectionusing Lexical Chains. International Journal of ComputerApplications 57(4): pp.1-4.

[15] Joe Carthy, Michael Sherwood- Smith (2002) , Lexical Chains for topic tracking, published in: Systems, Man andCybernetics, 2002 IEEE International Conference on Volume:7. pp.1-5.

[16] Lin, C.-Y., (2004), ROUGE: Recall-oriented understudy for gisting evaluation. In Proceedings of the ACL-04, 2004 (pp. 74-81)

[17] Jayarajan, D., Deodhare D., and Ravindran, B. "Lexical Chains as Document Feature", in 3rd International Joint Conference on Natural Language Processing, Hyderabad, January 2008. Volume-1, pp. 111-117.

[18] I. V. Mashechkin, M. I. Petrovskiy, D. S. Popov, and D. V. Tsarev, "Automatic text summarization using latent semantic analysis," In Programming and Computer Software, vol. 37, pp. 299–305, 2011

[19] Sulabh Katiyar, Samir Kr. Borgohain, " A novel approach towards automatic text summarization Using Lexical chain", in International Journal on Recent and Innovation Trends in Computing and Communication, ISSN-2321-8169, volume 3 issue 8 pp. 5115-5121, August 2015.