Application of Text Rank Algorithm for Context Based Indexing in Text Summarization

[1] Dipti D. Pawar
[1] M.Tech. Student, Bharati Vidyapeeth COEP,
Department of Computer Engineering,
Bharati Vidyapeeth College of Engineering,
Pune, India

M. S. Bewoor and [3]Dr. S. H. Patil
[2]Asst. Professor, Department of Computer Engineering,
[3]Professor, Computer Engineering Department,
Bharati Vidyapeeth College of Engineering
Pune, India

Abstract: **Today immeasurable information present on an internet. It often contains high quality information in the form of web pages. But it is not easy job to manually find out significant information and to pick the topmost collection of information for a particular information need. This problem can be resolved by Text Summarization. Huge amount of work has been done on similarity measure based text summarization task. Existing extraction based text summarization systems shows that the similarity values are calculated using indexing weights of the terms in the given input document. Due to this context independent indexing weights calculated similarity values remains independent on the context of a document. Very small work has been done for the problem of this type of indexing in text summarization task. This research work proposes the combine approach of Lexical association (semantic association) and context based document term indexing with NLP. In the particular we have also used novel concept of Lexical association to calculate the contextual similarity between sentences using computed indexing Weights. Then graph based ranking algorithms has been used with computed similarity values to obtain summary of document.**

Keywords-term indexing, text summarization, text rank, cosine similarity, sentence vector, lexical association.

I. INTRODUCTION

Now a day's internet is an immense source of electronic collections and high quality information. However, it provides more information than is required. User needs to choose top collection of data for particular information need in minimum possible time. Text summarization [1] is a task to covert the input text into a reduced form within a short duration of time, by protecting its general purpose and content. Hence text summarization is helpful for user to get relevant information. Also text summarization plays crucial role in the generation of snippets by search engine. Huge amount of work has been done on similarity measure based query dependent [2] text summarization task.

This project is primarily about the sentence extraction based text summarization task. There has been a great amount of work on use of graph based ranking algorithms [4] by extraction based text summarization task. In the particular different ranking methods has been used to determine score or significance the sentences in input text and top scored sentences are included in a summary of given text. In this extractive text summarization systems [1] some weights are assigned to the terms in a given document to calculate the sentence similarity values. These indexing weights are computed using type of characteristics like term frequency, text length etc. Hence problem of context independent term indexing occurs, in which indexing weight of document term remains independent on context in which document term appears. Little work has work has been done for the above problem. The main contribution of this thesis is to combine both approaches of Lexical association (semantic association) and context based document term indexing with Natural Language Processing (NLP) [8].

This proposed method aims at providing novel idea of context based term indexing to resolve problem of context independent term indexing. Every document contains content-specific and background terms .The indexing [12] used in existing models cannot differentiate between terms included in similarity measure. In this proposed system we are considering the above problem of indexing by using semantic association [9] between terms in given document. In the any given document, the contextual terms will be deeply linked with each other than that of the other terms. In this proposed system semantic association [12] between terms is nothing but the semantically related terms, which can be obtain through a corpus analysis. Context based document term indexing is implemented using Text rank [5] algorithm. It is also used to determine the informativeness of any sentence or document term.

Main aim to use the Lexical association [7] between terms is that the context in which term present gives important information about its sense. Sentence [12] similarity measure computed using this method provides the contextual sentence similarity.

II. SYSTEM DESIGN

There are several stages[14] while generating summary. As shown in Fig 1 initially input is given as a text file. Input text file undergoes through different NLP processing phases like Splitting, Tokenization, and Pos Tagging, Parsing etc., which results into meaningful document terms. These document terms appears as nodes in document graph. The total system analysis is divided into following phases.

Phase 1: Text preprocessing using OpenNLP tool.

i) Splitting and Tokenization

The input text is separated into disconnect sentences by the new line character and converted them into the array of paragraphs by using split method. This can be done by treating each of the characters '.', '!', '?' as separator. The text can be separated into tokens using word breaking characters like Punctuation marks, spaces and word terminators.

ii) Part of Speech Tagging

After tokenization POS tagging is applied to know grammatical semantics of tokens.

iii) *Parsing*

It creates parse tree for a given sentence.

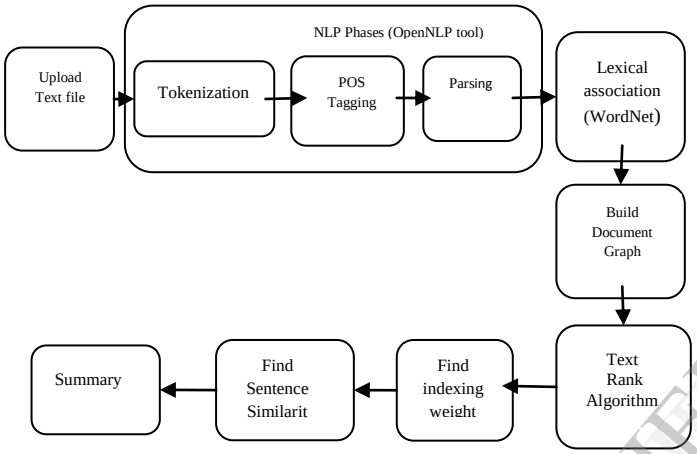

Figure 1: Architectural Diagram of proposed system

Phase 2: Lexical Association between document terms.

It [12] is basically used to determine the indexing weights of document terms using context based indexing. Once grammatical aspect of document term is clear we use our offline Word Net to find out different meaning of the word. The connection between different parts of the document will be recognizing using Word Net; and also it is used to extract the Lexical associations [9] between different terms which are most significant. It is the method to hold the text together by considering the semantic or identical relations between the terms of the text. The Associativity [12] of the document terms with each other can be stored in some table called as Lexical association table.

Phase 3: Indexing weights of document terms with both indexing methods.

i] Random indexing

In this method indexing of document terms is done only on the basis of term frequency rather than lexical association between terms.

ii] Context based indexing

After calculated lexical association measure from the table of semantically related terms, it is used to find indexing weight of document term in a text [12]. It can be calculated by using text rank algorithm [5].

Phase 4: Sentence similarity and generate summary for both indexing methods

Next step is to find Similarity between sentences using the function cosine similarity. Both indexing methods make the use same similarity measure to calculate sentence similarity. In random indexing methods indexing weights of document terms are calculated only on the basis of term frequency .Hence these weights are context independent and doesn't shows contextual similarity between sentences. But the similarity values calculated using context based indexing weights [12] of document terms reflects the contextual similarity between sentences with the help of lexical association. Then again graph based ranking algorithm [4] used by both indexing methods to find score of sentences and topmost scored sentences are included in summary. Sentence similarity values are stored in a table like structure called as similarity matrix given below

Sentence(Si)	Sentence(Sj)	Similarity Values
0	1	0.67
0	2	0.82
. .	. .	. .
0	17	0.82
15	17	0.76
16	17	0.63

Table 1: Sentence similarity matrix

III. SYSTEM IMPLEMENTATION

Our thesis includes different implemented functions in the form of following project modules

Module 1: Text preprocessing using NLP.

Uploading text file

The any standard text file can be uploaded through button '*Browse'*. The text file contains new line characters. After uploading different NLP processing like splitting, tokenization, tagging and parsing done on that file by calling OpenNLP tool. Implemented code for removing stop word from given text file is given as follow

Module 2: Lexical association (used for context based indexing only)

Once grammatical aspect of document term is clear we use our offline Word Net to find out different meaning of the word. The connection between different parts of the document will be recognizing using Word Net; and also it

is used to extract the Lexical associations [9] between different terms which are most significant. Lexical association will be display in table. In that table first column shows meaningful terms in input text file and next column lists semantically related terms from word net database with respective each term in first column. This lexical association will be used to calculate context based indexing weights only.

Module 3: Indexing weights of document terms.

In this module we are calculating indexing weights of terms using two methods as follows

1. Context based indexing method

From the table of semantically related terms, lexical association [9] measure between two terms is taken; the next job is to calculate weights of all document terms in a given document. It can be calculated by using Text rank [5] algorithm. For this graph for given document is built. We are considering undirected unweighted graph in which out-degree and in degree of a vertex are same. This method takes into account frequency of term plus frequencies of its semantically related words recursively in given text file recursively to calculate its indexing weight. Hence indexing weights of terms captures the context of the given text file. The score (indexing weight) of each vertex calculated using equation [5] "(1)" given below.

$$S(V_i) = (1-d) + d * \sum_{j \in In(V_i)} \frac{1}{|Out(V_j)|} S(V_j) \qquad \ldots\ldots\ldots\ldots (1)$$

S (V_i) : Score of the vertex
V_i : Vertex
In (V_i) : Number of predecessors
Out (V_j) : Number of successors
d: damping factor (The possibility of switching from a given vertex to another vertex set between 0 to 1)

2. Random Indexing Method

This method takes into account frequency of term in given text file to calculate its indexing weight. We are simply using a counter to calculate term occurrence. Hence these weights are remains independent on the context of input text file. Hence values of random indexing weights are less as compared to context based indexing weights.

Module 4: Sentence similarity and summary generation

In this phase similarities between sentences calculated for both indexing methods 1) context based indexing 2) Random indexing. Both indexing methods make the use of cosine similarity formula "(2)"given below to get similarity value between two sentences Si and Sj.

$$\mathbf{SIM}(S_i, S_j) = \frac{\overrightarrow{S_i} \cdot \overrightarrow{S_j}}{|\overrightarrow{S_i}| \, |\overrightarrow{S_j}|} \qquad (2)$$

For given sentences and calculated similarity values document graph is built. In this graph sentences appears as a vertex and similarity value between sentences appears as edges between vertices. Score of vertices calculated using equation (1).Finally topmost scored sentences are included in summary of input text.

Module 5: Performance analysis

In this module performance of summary generated by the both methods 1] Context based indexing 2] Random indexing is compared w.r.t time, space complexity and accuracy. Time complexity is nothing but the total computation cycles required for execution. The counter is assigned to obtain time complexity required for both the methods. A space complexity is nothing but the storage required for system variables, functions and data. Accuracy is measured using information retrieval measure Precision formula (3) as shown below.

$$\text{Precision} = \frac{\{\text{Total contextual terms}\} \cap \{\text{Total terms}\}}{\{\text{Total retrieved terms}\}} \qquad \ldots\ldots (3)$$

IV. RESULTS

This proposed system provides user with good Graphical user interface. In particular we have provided a different option for user to easily interact and use this system. Hence system mainly contains three forms. The first form provides user with basic NLP tasks, second form is about stop word removal and lexical association, and third form provides user a way to deal with options like indexing weight, sentence similarity and summary.

The figures shown below represents [15]snapshots for generated outputs like uploading input file, processing input file, building document graph, lexical association, indexing weights and summary. Initially text file is taken through '*browse*' option, and goes through different NLP tasks. Then it will use word net to get semantically related terms (lexical association).In Fig 2 we have shown that how the input text file is uploaded by our system. In particular user first clicks on a button *'Browse'* to upload any text file. Once a file is uploaded, we have provided user with different text preprocessing an options like *'tokenize the sentence'*, *'Pos tagging'*, and *'Chunk'*. Also we have provided an options for user like '*Next'*, '*Back*', and '*Exit*' to navigate through forms or quit.

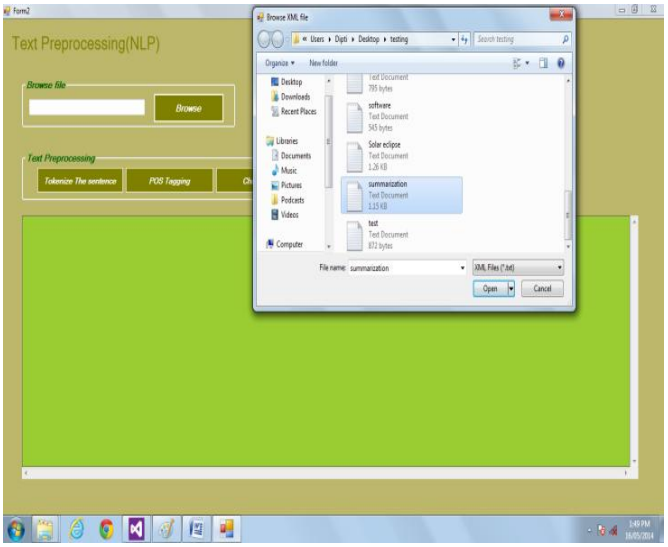

Figure 2: uploading of input text file

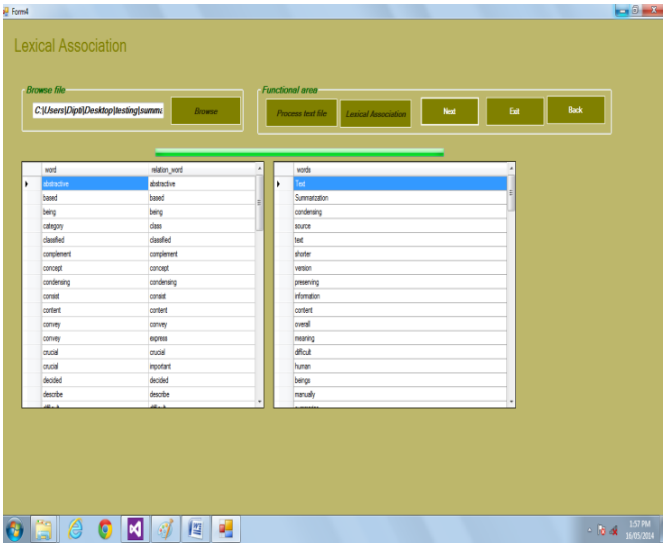

Figure 4: lexical association (semantic association)

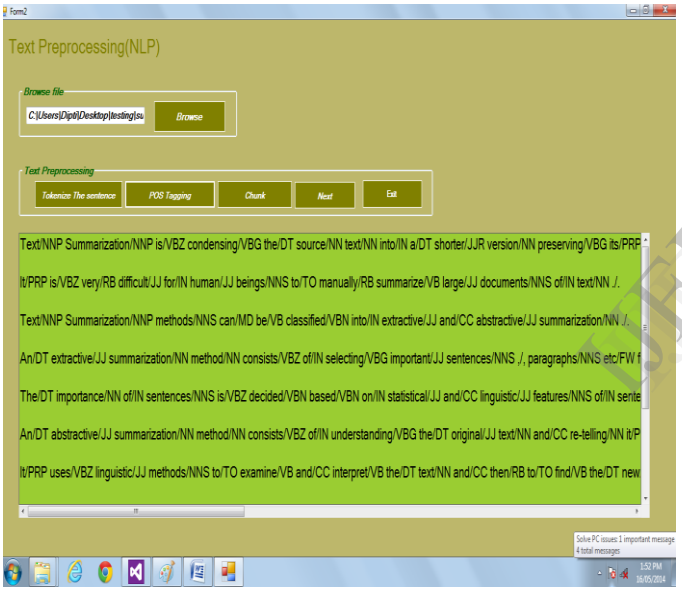

Figure 3: Text preprocessing on uploaded text file

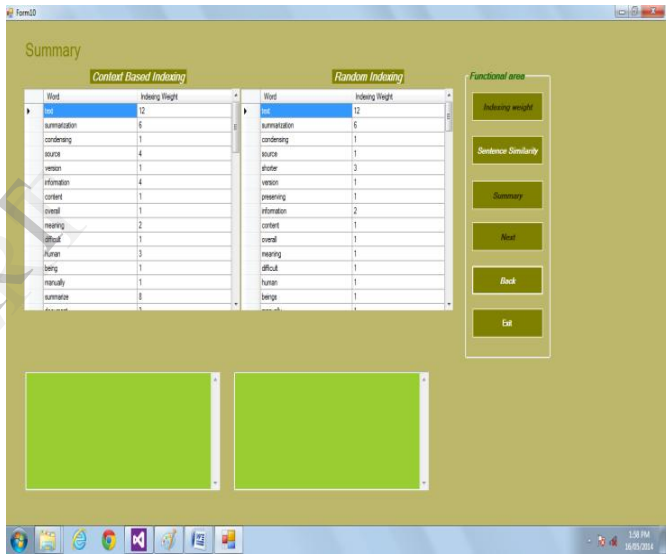

Figure 5: Context based indexing weights and Random indexing weights of document terms

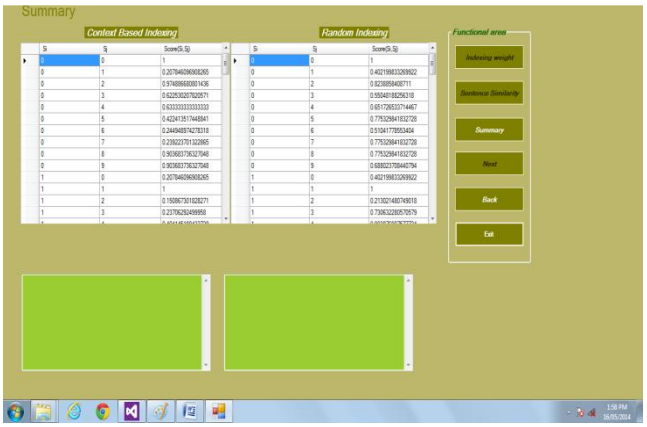

Figure 6: Sentence Similarity Matrix

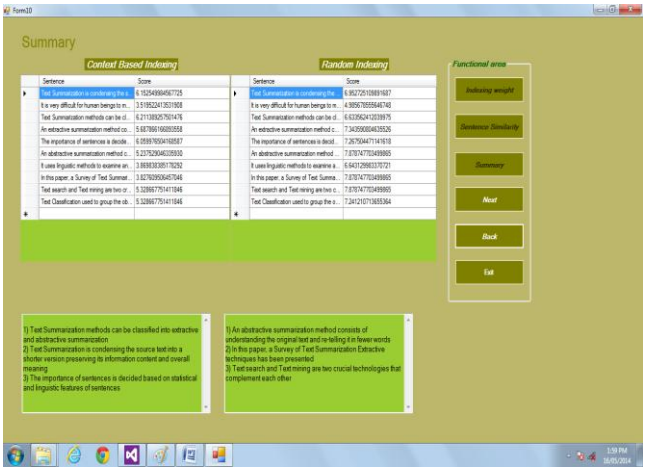

Figure 7: Summary of uploaded text files using both indexing methods

Figure 8: Performance analysis of Context based Indexing Algorithm and Random Indexing Algorithm with respect to accuracy.

V. CONCLUSION AND FUTURE WORK

In this thesis we have studied the random indexing and context based indexing and implementation of both indexing methods in text summarization. Also we have compared the performance of Text summarization using random indexing and context based indexing method. We have compared algorithms on various parameters like space complexity, time complexity, accuracy, redundancy and the performance of context based indexing is better than random indexing. Hence by comparing summaries obtained by both indexing methods, we have concluded that summary generated by our system is more contextual than random indexing technique.

In the future, we plan to widen our work to consider associations between documents of the dataset. Also we will try to implement same technique in different applications. Furthermore same technique can be applied on different file formats and best indexing method can be suggested for different file formats.

REFERENCES

[1] V.Gupta ,G. S. Lehal, "A Survey of Text Summarization Extractive techniques", Journal of Emerging Technologies in Web Intelligence, Vol. 2, No. 3, August 2010.
[2] D. Sureshrao, S. Subhash and P. Dashore, Analysis of Query Dependent Summarization Using Clustering Techniques, International Journal of Computer Technology and Electronics Engineering (IJCTEE) Volume 2, Issue 1.
[3] N.Chatterjee, S.Mohan, "Extraction-Based Single-Document Summarization Using Random Indexing".
[4] R. Mihalcea, "Graph-based Ranking Algorithms for Sentence Extraction, Applied to Text Summarization", Department of Computer Science University of North Tex asrada@cs.unt.edu.
[5] R. Mihalcea, P.Tarau, "Text Rank: Bringing Order into Texts".
[6] J.Leskovec, N. Milic-Frayling, M.Grobelnik, "Extracting Summary Sentences Based on the Document Semantic Graph", by Jurij Leskovec Natasa Milic-Frayling Marko Grobelnik.
[7] G. Ercan, I. Cicekli, Lexical Cohesion Based Topic Modeling for Summarization", Dept. of Computer Engineering Bilkent University, Ankara, Turkey.
[8] G. G. Chowdhury, "Natural Language Processing".
[9] P. Pecina, "Lexical Association Measures Collocation Extraction".
[10] Stephen, "Vector Space Models of Lexical Meaning".
[11] The complete Reference of .NET - by Matthew, Tata McGraw Hill Publication Edition 2003.
[12] P.Goyal, L. Behera, "A Context-Based Word Indexing Model for Document Summarization", Senior Member, IEEE, and Thomas Martin Mc Ginnity, Senior Member, IEEE.
[13] Abstract—//www.google.com.
[14] Dipti D.Pawar,"Context based indexing in text summarization using lexical association",IJERT.
[15] Dipti D.Pawar "Text Rank:A novel concept for extraction based text summarization",IJCSIT.