Context Sensitive Document Summarization using document term indexing with Lexical Association

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

diptipn24@gmail.com

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

msbewoor@bvucoep.edu.in

Abstract:

World Wide Web today is a largest source of online-information. Great amount of information is present on internet in the form of web pages. It is very tricky for human beings to manually find out valuable and significant information. This problem can be resolved by using text summarization. Text Summarization is the method of condensing the input text into shorter version by preserving its overall content and meaning. There has been a great amount of work on query specific summarization of documents using similarity measure. Indexing weights of the document terms are utilized to calculate the sentence similarity value which remains independent on context. Very little work has been done for the problem of context independent document indexing for the text summarization task. The main contribution of this research work is to combine both approaches of Natural Language Processing and context sensitive indexing. While doing so we have also used novel concept of Lexical association between document terms to measure the similarity between sentences using computed indexing Weights. The proposed concept of sentence similarity measure has been used with the graph-based ranking method to create document graph and obtain summary of document.

***Keywords**:* **NLP, Document Term Indexing, Document graph, Lexical Association, Text Summarization.**

I. INTRODUCTION

1. Overview

Text Summarization is information extraction technique, which generates shorter version or summary of original text such that generated summery is useful to give an overview of original text within a short duration of time.Text summary can be generic or query dependent.Text Summarization methods can be classified into extractive and abstractive summarization [1]. Former method consists of selecting significant words, sentences, paragraphs etc. from the input document and concatenating them into shorter form. The importance of sentences is decided based on statistical and linguistic features of sentences. Later method attempts to develop an understanding of the important concepts in a document and then express those concepts in natural language. Furthermore text Summarization can also be specific to the information needs of the user called as query specific document summarization [2].The task of producing summary from multiple documents is called multiple document summarizations [3].

Furthermore clustering based approach [4] can be use for text summarization that groups first, the similar documents into clusters & then sentences from every document cluster are clustered into sentence clusters. And top scoring sentences from sentence clusters are chosen in to the final summary. During the process of text summarization different condensation operations can be applied on entities such as words, phrases, clauses and sentences. These entities can be analyzed at various linguistic levels: morphological, syntactic, and semantic and discourse. Based on the level of linguistic analysis of the source, summarization methods [1] can be classified into approaches as follows

A) Shallow approach:

It considered features such as word count, presence of cue phrases, Position of the sentence to compute the important concepts of the document and saliency of the information.

B) Deeper approach :

It helps to find the theme or context of the content. Lexical association [5] comes under this approach. It is the method to hold the text together by considering the semantic or identical relations between the words of the text.

2.Proposed System

Text Summarization is the method of condensing the input text into a shorter version, preserving its information content and overall meaning [1].This paper focuses on sentence extraction based text summarization. Most of the previous methods on the sentence extraction-based text summarization task use the graph based algorithm [6] to compute importance of each sentence in document and most important sentences are extracted to generate summary of document. These extraction based text summarization methods [7] give an indexing weight to the document terms to calculate the similarity values between sentences .Document features like term frequency, text length are used to assign indexing weight to terms. Therefore document indexing weight remains independent on context in which document term appears. This proposed method aims at providing novel idea of context sensitive document term indexing to resolve problem of context independent document indexing. Every document contains content-specific and background terms .The indexing methods used in existing models cannot differentiate between Terms reflected in sentence similarity values.

In this proposed method we are considering the problem of context independent document indexing using Lexical association. In the document, the content specific words will be highly related with each other, while the background terms will have very low relationship with the other terms in the document. In this research work the relation between document terms is captured by the lexical association, computed through a corpus analysis.

Context based document indexing is implemented using Text Rank algorithm [6] to compute how informative is each of the document term .Main motivation behind using Lexical association is the main assumption that the context in which word appears gives valuable information about its meaning. Sentence similarity values calculated using the context sensitive indexing provides the contextual similarity between two sentences. This will allow two sentences to have distinct similarity values depending on the context.

II. SYSTEM ARCHITECTURE

There are several stages while generating summary. As shown in figure 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: NLP processing on the Input Text file.

i) Splitting and Tokenization

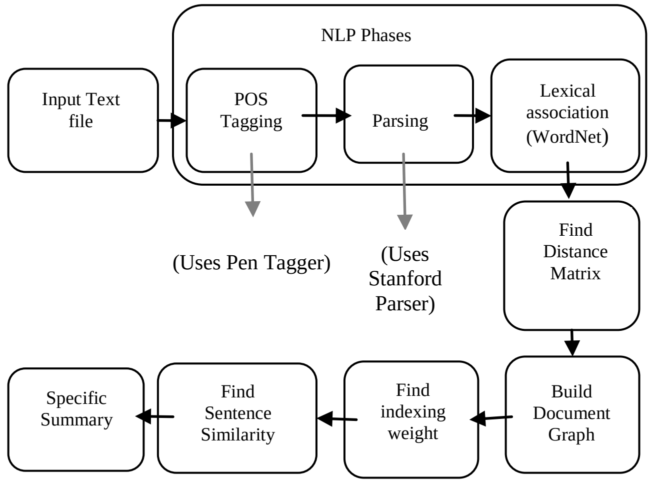

Figure 1 Architecture diagram of our system

The input text is divided into separate sentences by the new line character and transferred into the array of paragraphs by using split () method. This can be done by treating each of the characters '.', '!', '?' as separator rather than definite end-of-sentence markers.

The text can be separated into tokens using word breaking characters like Punctuation marks, spaces and word terminators.

ii) Part of Speech Tagging

Once the words are tokenized next POS tagger is applied to them for knowing their grammatical semantics. We are also using Penn tagger for Part of Speech tagging

iii) Parsing

Stanford parser is freely available on Internet It converts an input sentence into a hierarchical structure that corresponds to the units of meaning in the sentence.

Phase 2: Find Lexical Association between document terms and build distance matrix.

Once grammatical aspect of word is clear we use WordNet to find out different senses of the word. We use WordNet to understand the links between different parts of the document; subsequently extract the Lexical associations between two document terms which are most relevant. It is the method to hold the text together by considering the semantic or identical relations between the terms of the text. Associativity of the document terms with each other can be stored in some data structure called as distance matrix.

Phase 3: Building Document Graph and Finding context based indexing weight of document terms

Given the lexical association measure between two terms in a document from distance matrix, the next task is to calculate the context sensitive indexing weight of each term in a document. A graph-based iterative algorithm is used to find the context sensitive indexing weight of each term.In this graph for given document is built. Let G= (V, E) be an undirected weighted graph to reflect relationship between terms in document, where each vertex V= {vj |1≤ j≤ |V|} denotes set of vertices and each vertex is document term and E is a matrix of dimensions |V | ×|V |. Each edge ejk € E gives the lexical association value between the terms corresponding to the vertices vj and vk. The lexical association between the same terms is set to 0. Indexing weight of each term shows the importance of that term in document.

Phase 4: Finding Sentence similarity and generating summary

Next step is to find Similarity between sentences using the function sim (Si, Sj). Similarity values calculated using context based indexing weights of document terms reflects the contextual similarity between terms. In this for each sentence Sj in the document, the sentence vector $\overrightarrow{Sj}$ is built using calculated indexing weights of sentences. The sentence vector is calculated such that if a term vt present in sentence Sj, it is given a weight of term vt; else it is given a weight 0. The similarity between two sentences Si and Sj is computed using Equation (1).

$$Sim\ (Si, Sj) = \overrightarrow{Si}.\ \overrightarrow{Sj}\quad , \qquad (1)$$

At the end depending on the contextual similarity value summary will be generated specific to users query.

III. RELATED WORK

R.Varadarajan and V. Hristidis [4] developed a model to create Query Specific Summaries by identifying the most query-specific fragments and combining them using the semantic associations in the document. They focused on keyword queries since keyword search was the most popular information discovery method on documents. In particular, initially structure was added to the documents in the preprocessing stage and converted them to document graphs. Document graph was used to represent the hidden semantic structure of the document and then perform keyword proximity search on this graph. Then, the effective summaries were computed by calculating the top spanning trees on the document graphs.

R. Mihalcea [6] introduced a novel unsupervised Method for automatic sentence extraction using graph based ranking algorithms. The graph-based ranking algorithm is a method of deciding the importance of a vertex within a graph, by taking into account global information rather than only the local vertex-specific information. A similar method was applied to lexical or semantic graphs extracted from natural language documents using a graph-based ranking model called as Text Rank, which can be used for a variety of NLP applications where knowledge taken from a whole text was used in making local ranking decisions. Such text-based ranking methods can be applied to tasks ranging from automated extraction of key phrases, to extractive summarization and WSD. Text Rank finds the connections between various entities in a text, and executes the concept of recommendation.

R. Barzilay and M. Elhadad [8] introduced a new method to compute lexical chains in a text using knowledge sources like Word Net thesaurus, a part-of-speech tagger, and shallow parser. Summarization works in steps like Text segmentation, Lexical chain creation, scoring chains and sentence extraction. Furthermore they expand the set of candidate words to include noun compounds and evaluating importance of noun compounds by taking into account the noun compounds explicitly present in Word-Net. They generate chains in every segment using relatedness criteria, and next, they combine the chains from the different segments using much stronger criteria for connectedness only two chains are combined across a segment boundary only if they include a common word with the same meaning. For each text, they manually ranked chains in terms of relevance to the main topics. They then computed different parameters on the chains, including chain length, homogeneity index.

Erkan and Radev [10] introduced a stochastic graph-based method for computing relative importance of textual units for NLP. LexRank is use for calculating sentence significance based on the notion of eigenvector centrality in a graphical representation of sentences. In this model, a connectivity matrix based on intra-sentence cosine similarity was used as the adjacency matrix of the graph representation of sentences to assess the centrality of each sentence in a cluster and extract the most significant ones to include in the summary. They introduced different ways of defining the lexical centrality principle in multiple-document summarization, which computes centrality in terms of lexical characteristics of the sentences.

C. Zhou, W. Ding and Na Yang [9] authors introduced a dual indexing method for search engines based on campus Net. Indexing method, which means, it has document index as well as word index. Document index is depends on the documents to do the clustering, and arranged by the position in each document. In the information retrieval, the search engine first takes the document id of the word in the word index, and then goes to the position of respective word in the document index, because in the document index the word in the same text document is adjacent. The search engine compares the biggest word matching assembly with the sentence that user provides. The method proposed by them seems to be time consuming as the index exists at two levels.

IV. CONCLUSION AND FUTURE WORK

In this work we presented an indexing structure that can be constructed on the basis of the context of the document. The context of the document can be extracted by using thesaurus and ontology repository. So this paper uses Lexical association for context based index building. The context based indexing enables extraction from index on the basis of context rather than keywords. This aids in improving the quality of the retrieved results. The context based indexing plays an important role in result space and time consumption. We show with a user survey that our approach performs better than other state of the art approaches.

In the future, we plan to extend our work to account for links between documents of the dataset. Also we will try to implement same algorithm 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 and G. S. Lehal ,A Survey of Text Summarization Extractive techniques, Journal of Emerging Technologies in Web Intelligence, Vol. 2, No. 3, August 2010.

[2]R. Varadarajan and V. Hristidis, A System for Query-Specific Document Summarization, School of Computing and Information Sciences Florida International University Miami, FL 33199.

[3]D. Radev, H. Jing , M. Sty's ,and D. Tam, Centroid-based summarization of multiple documents, Information Processing and Management 40 (2004) 919–938, University of Michigan, Ann Arbor, MI 48109, USA IBM T.J. Watson Research Centre, Yorktown Heights, NY 10598, USA, 24 October 2003.

[4] 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.

[5]G. Ercan and I. Cicekli, Lexical Cohesion Based Topic Modeling for Summarization, Dept. of Computer Engineering Bilkent University, Ankara, Turkey.

[6]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.

[7]X. Wan and J. Xiao, Exploiting Neighbourhood Knowledge for Single Document Summarization and Keyphrase Extraction, ACM Trans. Information Systems,vol.28,pp.8:1-8:34,http://doi.acm.org/10.1145/1740592.1740596, June 2010.

[8]R. Barzilay and M. Elhadad, Using Lexical Chains for Text Summarization, Mathematics and Computer Science Dept. Ben Gurion University in the Negev Beer-Sheva, 84105 Israel .

[9]C. Zhou, W. Ding and Na Yang, Double Indexing Mechanism of Search Engine based on Campus Net, Proceedings of the 2006 IEEE Asia-Pacific Conference on Services Computing (APSCC'06) Quan, T. T., Hui, S. C., Fong, A.

[10]G. Erkan and D. Radev, LexRank: Graph-Based Lexical Centrality as Salience in Text Summarization, J. Artificial Intelligence Research,vol.22,pp.457-479http://portal.acm.org/citation.cfm?id=1622487.1622501, Dec. 2004.

[11] Z. Harris, Mathematical Structures of Language, Wiley, 1968.