ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

IJARCCE

International Journal of Advanced Research in Computer and Communication Engineering

ICRITCSA

M S Ramaiah Institute of Technology, Bangalore

Vol. 5, Special Issue 2, October 2016

Graph Based Approach for Automatic Text Summarization

Akash Ajampura Natesh[1], **Somaiah Thimmaiah Balekuttira**[2], **Annapurna P Patil**[3]

Department of Computer Science and Engineering, M S Ramaiah Institute of Technology, Bangalore, India [1, 2, 3]

Abstract: Research in Automatic text summarization systems has gained momentum in recent times mostly due to the advances in natural language processing libraries and techniques. In this work, we have proposed a graph based approach for automatic text summarization. This approach uses the concept of computing how closely, significant words in a sentence are related to each other. This metric further weighs the significance of the sentences in the text document. NLTK library for python is used to build the automatic text summarization system based on this approach. The results obtained show that this technique is effective in producing high quality summaries.

Keywords: Automatic Text Summarization, Extractive Summarization, Natural Language Processing, NLTK Library.

I. INTRODUCTION

Automatic text summarization is the process by which the condensed information of a text is retrieved from the original text. It produces relevant and specific information from a large amount of data in the text format. It has numerous applications from analyzing audits to producing quick understandable notes. Text summarization condenses documents to less than half of their original size without significantly compromising on the semantics of the documents. An efficient text summarizer must generate concise summaries of documents laden with redundancies. This field of research is in existence since 1950. Even though there are problems pending in this field of research yet to be solved, there has been significant progress, especially after the advances in natural language processing libraries and techniques.

There are two types of text summarizers. A text summarizer can be either extractive or abstractive. An extractive summarizer picks sentences of the highest scores directly from the source document without modification. It involves concatenating extracts taken from the original text and adding it to the final summary. In this type of summarization, portions such as words or sentences of the text are reused in the summary. In contrast, the abstractive summarizer rebuilds sentences chosen to constitute the final summary. It translates the original document into a more concise text where the final words are more compact; yet represent a thorough representation of the idea of the initial text. Abstractive summarizers generate novel sentences from information gathered from the original document. For instance, the sentence “They visited New South Wales, Queensland and Victoria during summer” could be summarized as “They visited Australia during summer”.

The primary objective of a text summarizer is to produce a summary of the original text with least redundancy. It must reflect all the key ideas of the original document with minimum duplication but with maximum coherence amongst the summarized sentences.

In every document written in the English language, the nouns of the text play the most vital role in helping us understand the meaning of the text basis the context it was written in. The proposed approach constructs a graph of all the nouns of the text to determine how closely related the nouns of the text are to each other which ultimately helps in weighing the sentences. The approach that we present in this paper is an extractive based text summarization method. The sentences are scored based on how significant the nouns present in the sentence are to the entire document. The high scoring sentences are considered the most important sentences in the text and these sentences are chosen for the summary.

The rest of the paper is organized as: Section 2 is about related work in the field of text summarization; Section 3 describes the Proposed Approach; Section 4 describes the modules used; Section 5 describes the Evaluation method used; Section 6 presents the Test Cases and Results; Section 7 represents Conclusion of the paper.

II. RELATED WORK

In recent years, many methods and algorithms for text summarization have been proposed. One of the most looked into method of text summarization is the Graph-based method which builds graph models for the text and applies ranking algorithms on the models for summary generation. All the Graph Based methods mainly include the tasks of pre-processing, building graph models, applying ranking algorithms and finally generating summaries. Many approaches for graph based text summarization have been proposed.

DOI 10.17148/IJARCCE

In [3] S. S. Ge, Z. Zhang and H. He in 2010 proposed a weighted graph model using a hybrid approach which involves both sentence clustering and ranking for document summarization.

This approach for text summarization is Graph based and clustering. Steps followed in this approach are

1. It uses both sentence ranking which is used in graph model and clustering for combining similar sentences.
2. Sentence clustering is done for the text based on singular non matrix factorization.
3. Finally, weighted graph model approach used in this approach considers discourse relationship between sentences for clustering and ranking sentences in a document.

In [4] S. Hariharan and R.Srinivasan in 2009 have investigated a method for summarization of news articles using Graph Based method. In this method, measure of similarity between sentences of the article is represented by an adjacency matrix, which forms the foundation of Graph Based techniques. There are two techniques that are investigated in this paper. Cumulative sum was the first technique which was proposed by the authors. The degree of centrality was the second technique investigated, which was an already existing method. Further in this paper, with the help of the above two techniques a new method for evaluating the adjacency matrix was proposed which introduces two metrics: Effectiveness 1 and Effectiveness 2. These help in evaluating the system summaries with the human summaries. Comprehensive Investigations have showed that this method is better than basic methods and provides further scope for improvement in this area of text summarization.

In [5] K. S. Thakkar, R. V. Dharaskar and M. Chandak in 2010 proposed a method which uses an approach similar to Text Rank. Shortest path algorithm is used for summary generation.

The following steps are used for summarization

1. A graph model is built for representing the text which connects text entities in the graph to form meaningful relations. After which a graph based ranking algorithm is used to score each vertex of the graph that was generated in the previous step.
2. Finally, shortest path algorithm is applied on the graph to generate the text summary.

In [6] Xiaojun Wan in 2008 proposed a method for multi-document summarization which uses graph-based ranking algorithm. It assumes that all the sentences are indistinguishable. A concept of Document impact on summarization performance is talked about here along with document-based graph model. This is to incorporate the document-level information and the sentence-to-document relationship into the graph-based model for ranking the sentences for extraction. The graph is a two-link graph which includes both sentences and documents. It works on the assumption that sentences that belong to an important document, highly correlated with the document, will have higher chances of being chosen for the summary.

III.PROPOSED METHODOLOGY

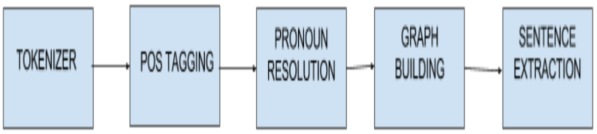

Figure 1 Flow graph depicting the proposed system

The components of the flow graph shown above in Figure 1 are discussed below

A. Tokenizing

Tokenizer performs tokenization on the paragraph as a whole and also on individual sentences. Hence the two functions performed are,

Sentence Tokenizing: The given text is decomposed into sentences.

Word Tokenizing: Each sentence is decomposed into a stream of individual tokens (words).

B. Parts of Speech Tagging

Part of speech tagging is the process of marking the tokens (words) with the part of speech to which they belong. Each token (word) is assigned the most appropriate part of speech tag depending on the form of the word and the tags of its neighboring words. This is a very crucial step for the semantic analysis of the sentence with respect to the context in which it is written. The parts of speech of each of the words finally help in understanding the meaning of each of the sentences.

C. Pronoun Resolution

Pronouns are mapped to the corresponding nouns. The nouns are classified as ‘objects’ or ‘person’ and the most significant person in the previous sentence is mapped to pronouns that refer to a person and the most significant object in the previous sentence is mapped to pronouns that refer to an object.

D. Graph Building

A graph is built with nouns as vertices and the weights of the edges connecting them represent the relevance between the nouns.

E. Sentence Extraction

Each sentence is scored based on the collective weights of the nouns in the sentence. The sentence with the highest score is chosen for the final summary.

IV. MODULES

A. Tokenizer

It takes as input a text document and returns a list of sentences and a list of words.

First the document is decomposed into sentences. Then each sentence is split into words.

Tokenize (Text Document): returns a list of sentences and a list of words

Input: “The English were not the first Europeans to land their ships on American soil.”

Output: ['The', 'English', 'were', 'not', 'the', 'first', 'Europeans', 'to', 'land', 'their', 'ships', 'on', 'American', 'soil']

B. Part of Speech Tagging

It takes as input the list of tokens and assigns the most appropriate part of speech tag to each token.

It also creates a separate list of nouns corresponding to each sentence.

Input: [‘The’, 'English', 'were', 'not', 'the', 'first', 'Europeans', 'to', 'land', 'their', 'ships', 'on', 'American', ‘soil’]

Output: [('The', 'DT'), ('English', 'NNP'), ('were', 'VBD'), ('not', 'RB'), ('the', 'DT'), ('first', 'JJ'), ('Europeans', 'NNPS'), ('to', 'TO'), ('land', 'VB'), ('their', 'PRP'), ('ships', 'NNS'), ('on', 'IN'), ('American', 'JJ'), ('soil', 'NN')]

C. Pronoun Resolution

For each pronoun found in the text, it is substituted with the noun that is referred.

Each noun is classified as a person or an object. The noun that appears as the subject in the preceding sentences is considered to be the most significant noun. It keeps track of the most significant person and the most significant object. Depending on the pronoun, either it is substituted by the most significant noun or the most significant person. At the end of this step all the pronouns are mapped to their corresponding nouns.

Input: “Grenville, the English captain, was furious. He believed that the Aquascogoc had stolen the silver cup.”

Output: “Grenville, the English captain, was furious. Grenville believed that the Aquascogoc had stolen the silver cup.”

D. Graph Building

For any two nouns in the same sentences, an edge is added between them. First the distance between the nouns in the sentence is calculated as the number of words appearing in between the two noun phrases.

distance(n1,n2) = | position(n1) - position(n2) |

where, n1 and n2 are nouns.

The weights of the edges signify the relation between them. Nouns that appear in the same sentence have some degree of correlation. A pair of nouns that appear together in multiple sentences have higher correlation. The edge weight is inversely proportional to the distance between the two nouns.

edge_weight(n1,n2) = 1/(1+(distance(n1,n2)))

where, n1 and n2 are nouns.

E. Sentence Extraction

The relevance score of each noun is calculated as the summation of weights of all the edges associated with the noun.

relevance(n) = ΣNi=0 edgeWeight(n,i)

where, n is a noun , N is the total number of nouns.

The sentence score is calculated as the summation of relevance of all the nouns present in it

sentenceScore(s) = ∀n ∈ s Σ relevance(n)

where, n is a noun and s is a sentence.

The sentence with the highest score is chosen and added to the summary.

To avoid choosing the same sentence, the relevance score of all the nouns that appear in the chosen sentence is reduced by a predetermined factor, γ.

∀n ∈ s relevance(n) = (1- γ) * relevance(n)

where, n is a noun, s is a sentence and γ is the reduction factor.

For the development of the system NLTK library was used in python language. It uses PunktSentenceTokenizer for decomposing the document into sentences and TreebankWordTokenizer to tokenize words . It uses PerceptronTagger for part of speech tagging

V. EVALUATION METHOD AND RESULTS

Evaluating automatic text summarization systems is not a straight-forward process. There are many measures that can calculate the topical similarities between two summaries. For evaluating the result for the proposed method we use a manual corpus with text for which a summary is written by an experienced human editor. The corpus and the generated summary on which evaluation is performed on can be found at https://github.com/Akash-an/TextSummarization. The main advantage of this method is the good quality short summary of the sample that is available to compare the generated summary with. For evaluation 33% of the original text is produced as summary. The reducing factor (γ) is chosen to be 0.1.

ROUGE-1 [7] was used for the assessment of the summaries.

The results are expressed in terms of three parameters:

Precision(p) correct : (correct | wrong)

Recall(r) correct : (correct | missed)

F measure (FM) 2 * p * r : (p | r)

where,

correct = Those words common to both reference summary and the system summary.

wrong = Those words present in system summary but not in the reference summary.
missed = Those words present in reference summary but not in the system summary.

Table1. ROGUE-1 Evaluation Results

ROUGE-Type	System Name	Avg_ Recall	Avg_ Precision	Avg_ F-Score	Number of Reference Summaries
ROUGE-1	DOC1.txt	0.3270	0.32000	0.32350	2
ROUGE-1	DOC2.txt	0.3350	0.33000	0.33370	2
ROUGE-1	DOC3.txt	0.3293	0.32000	0.32270	2

Table 1. shows the evaluation results of performing Rogue-1 analysis on three sample text documents and the generated summaries are evaluated against summaries written by experts. This shows that this method can be used for generation of good quality summaries of text documents. This approach reduces the number of sentences in the original document by a significantly high percentage to generate the summary.

VI. CONCLUSION

In this paper, we have presented a graph based approach for automatic text summarization. We propose an approach which consists of five steps. First the document is decomposed into sentences. Each of these sentences are further split into words. Part of Speech Tagging is performed which takes as input the list of tokens and assigns the most appropriate part of speech tag. It also creates a separate list of nouns corresponding to each sentence. After this, Pronoun Resolution is done. For each pronoun found in the text, it is substituted with the noun that it refers to. Finally, the words in the sentences are connected by a link and a graph is built for the entire text. A weight is computed for each of the links in the graph depending on factors such as distance of the most significant nouns in sentence, after which the most significant sentences of the text is extracted into the summary. This method of text summarization works well with news articles, Wikipedia searches and technical documents. ROUGE-1 was used for the assessment of the summaries. The results above show that this method performs reasonably well for automatic text summarization. The main contributions of this study are as follows:

1. It proposes a sentence similarity computing method based on parts of speech tags of the words in the sentences and on how closely they are related to each other.
2. It gives a method to assign link weights in the graph which represents the entire text.
3. It gives a graph based approach for automatic text summarization.

ACKNOWLEDGMENT

The authors would like to thank the Department of Computer Science and Engineering, M S Ramaiah Institute of Technology, Bangalore, for their constant support and encouragement during the course of this project.

REFERENCES

[1] Pal, Alok Ranjan, and Diganta Saha. "An approach to automatic text summarization using WordNet", 2014 IEEE International Advance Computing Conference (IACC),2014.

[2] Meena, Yogesh Kumar, Ashish Jain, and Dinesh Gopalani. "Survey on Graph and Cluster Based approaches in Multi-document Text Summarization", International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014), 2014.

[3] S. S. Ge, Z. Zhang, and H. He, "Weighted graph model based sentence clustering and ranking for document summarization, " in Interaction Sciences (ICIS), 2011 4th International Conference on, pp. 90-95, IEEE,2011.

[4] S. Hariharan and R. Srinivasan, "Studies on graph based approaches for single and multi document summarizations," Int. 1. Comput. Theory Eng, vol. 1, pp. 1793-8201, 2009.

[5] K. S. Thakkar, R. V. Dharaskar, and M. Chandak, "Graph-based algorithms for text summarization, " in Emerging Trends in Engineering and Technology (lCETET), 2010 3rd International Conference on, pp.516- 519, IEEE, 2010.

[6] Xiaojun Wan, “An Exploration of Document Impact on Graph-Based Multi-Document Summarization,” Proceedings of the Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Honolulu, Hawaii, pp. 755-762. 2008.

[7] C.-Y. Lin, “Rouge: A package for automatic evaluation of summaries,” in Text Summarization Branches Out: Proceedings of the ACL-04 Workshop, M.-F. Moens and S. Szpakowicz, Eds. Barcelona, Spain: Association for Computational Linguistics, July 2004, pp. 74–81.

[8] F. Wei, W. Li, Q. Lu, and Y. He, "A document-sensitive graph model for multi-document summarization, " Knowledge and information systems, vol. 2, no. 2, pp. 245-259, 2010.

[9] T.-A. Nguyen-Hoang, K. Nguyen, and Q.-V. Tran, "Tsgvi: a graph-based summarization system for vietnamese documents, " Journal of Ambient Intelligence 313, 2012.

DOI 10.17148/IJARCCE