International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2015): 6.391

Article Summarizer

Aditya Gaikwad[1], Akshay Bhange[2], Atul Dalvi[3], Amit Nerurkar[4]

Department of Computer Engineering, Vidyalankar Institute of Technology, Mumbai, India

Professor, Department of Computer Engineering, Vidyalankar Institute of Technology, Mumbai, India

Abstract: ***Summarizing a text file is the process of constricting source file into a shorter version of the same file conserving its information content and generic meaning of the input source file. Human beings are the most effective mechanism used to generate summary, but if we have to generate a summary for a way too large document in few minutes, in such case automatic text summarizer comes into picture. "Automatic text summarization:" process' source file and generate summary by inputting the source file to the machine (computer) and it returns a summary of the original text file. Text summarization is classified in two methods: extract based summarization & abstract based summarization. In extract based summarization method the input source file is processing in such a way that the important sentences from the document are selected and integrated to form a summarized output file. An abstract based method on the other hand selects the most important sentences & para-phrase those keeping the meaning overall meaning of the source file intact.***

Keywords: Text Summarization, Extract based summarization, Abstract based summarization

1. Introduction

Our ultimate goal is to create a robust summarization system that can handle different types of documents in a uniform way. To achieve our goal, we have developed a summarization application based abstract summarization technique explained earlier. Sentence Extraction is required for summarization system to reduce size of the document. Features such as position of sentence, frequency of word & term frequency-Inverse document frequency etc. in the article should be integrated, in order to extract sentences.

We seek answers for the following questions.

- What are frailty & vigour of the existing article summarizer??
- How much of extra vocabulary from a document can be reduced without losing the useful information??
- Too what extent the existing applications can be improved??
- To what extent we can improve the existing applications??

2. Proposed System

Following are the major steps of our summarizer:-

1) Pre-processing- Pre-processing step is necessary for shortening the time required to process summarization of the document. We have included some sub modules in the pre-processing module:-

a) Stop word removal: - For removing stop-words we have created a stop word database so as to match it to the content in the input source file (if match found remove the word else don"t).

b) Sentence cut-off: - It keeps the count of the number of words in the sentence & checks whether they do not fall under the minimum value or above the maximum value, if they do then the whole sentence is eliminated from the summary.

c) Paragraph marker: - It is used to mark the first & the last sentence from each of the paragraph for using in the summary.

2) Title matching & Stemming- Stemming is done first then title match comes into picture for fast processing. The process to remove the suffixes & prefixes of the words in order to obtain the root word for further processing is called stemming. In title matching we do as the name says match the title with each and every sentences to check whether the match is found for obtaining value to give importance to particular sentence. For Ex-("from the word anthropology if we suffix „logy-science" we get the root „Anthropos-mankind" which is to be used further"). We have used Apache tool from stemming as it is open source library easily available & dependable.

3) Term Frequency-Inverse Document Frequency- It is the core of our article summarizer. The TF-IDF value is obtained by multiplication of the two terms (TF*IDF). Where TF= count of each word occurred in the document & the IDF obtained by the formula ("(IDF= log (N/DF))"). Where N = count of paragraphs in the article & DF = number of paragraph in which the particular term occurs.

4) Post-processing- After getting values from all the above features, we generate a weighted graph to generate rank of the sentence.

5) Summary generation- We have discussed about Extract based Summarization. These summaries are produced by ranking the sentences based on their scores which they get from the features discussed above. Now in our application we have given control to the user about the number of sentences required by him for summary; of course there would be some threshold to prevent it from misuse. To increase the readability, the sentences in the summary are reordered based on their appearances in the original text, for example, the sentence which occurs first in the original text will appear first in the summary.

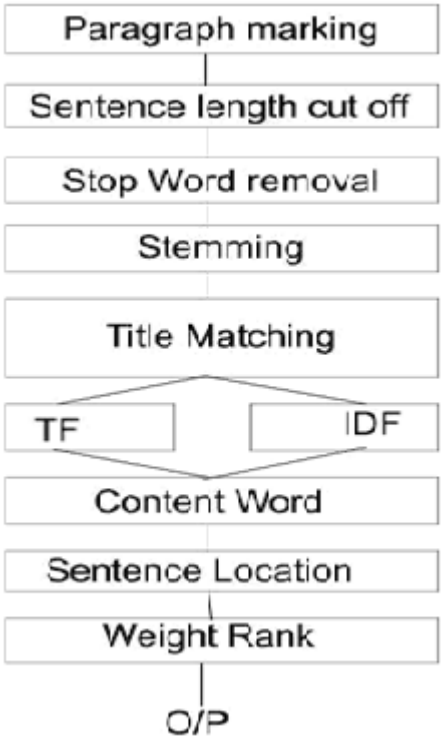

Figure 1: Overview of processing

3. Application Design

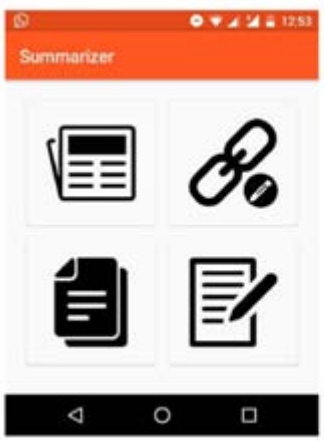

Figure 2: Home Page

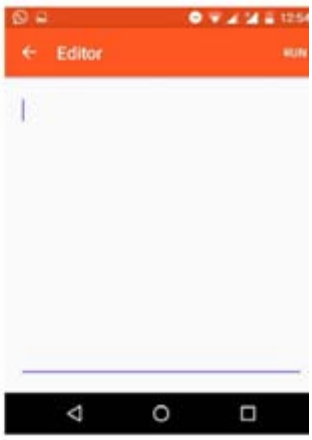

Figure 3: Edit Page

4. Conclusion

English is the primary language used by us for single document text summarization. This paper concentrates on Extractive text summarization methods which is based on the selection of the most important sentence from the document. The importance of the sentences are based on statistical & linguistic features of the sentence. The overall performance system may be further improved by improving the stemming process, exploring more number of feature & integrating them with the existing features as well as applying learning algorithm for effective feature combination.

To summarize content from various sources is the major challenge we have faced during the development process. The text summarization application should generate effective summary in less time & less redundancy. There are several features which are not included in our application, but in future, we are planning to use these features to improve the performance of the sentence extraction.

References

[1] Kamal Sarkar, "Bengali text summarization by sentence extraction".
[2] Vishal Gupta & Gurpreet Singh Lehlal, "A survey of text summarization extractive techniques", Journal of emerging technologies in Web Intelligence, August-2010.
[3] Chikasi Nobata, Satoshi Sekine & Hitoshi Isahara, "Evaluation of features sentence extraction on different types of corpa".
[4] Yiming Yang & Jan O. Pedersen, "A comparative study of feature selection in text categorization".