Webology, Volume 18, Number 5, 2021
ISSN: 1735-188X
DOI: 10.29121/WEB/V18I5/52

Text Summarization Employing Sentence Ranking Approach

Bhasker Pant[1], Vrince Vimal[2]

[1]Department of Computer Science & Engineering,Graphic Era Deemed to be University, Dehradun, Uttarakhand India, 248002
[2]Department of Computer Science & Engineering,Graphic Era Hill University, Dehradun, Uttarakhand India, 248002

ABSTRACT

Automatic Text Summarization (ATS) is the technique of selecting the significant and applicable information from a given text and is referred to as a summary of the text. There are two ways to approach it. 1) Text that is summarised abstractly, and 2) Text that is summarised in an extractive manner. The term "extractive text summarization" refers to taking key information or sentences from a given text file or the original document and summarising them. This article presents a unique statistical approach for doing an extractive text summarization on a single document. A technique for extracting sentences is described here; this approach summarises the meaning of the input text in a condensed format. Weights are given to each sentence, and then those weights are used to determine where each sentence falls in the overall ranking. It does this by selecting highly rated sentences from the input text, allowing it to choose essential phrases that tends to produce a high-quality summary of the source file, which is then stored as an audio file.

Keywords: Text Summarization, Extractive Approach, Term Weighted frequency, Sentence extraction, Summary.

INTRODUCTION

In this day and age, when a large amount of new material is being generated on the internet on a daily basis, it is essential to keep up. Therefore, it is essential to develop a better system for extracting important information most quickly and efficiently as possible. The text summary may be used to discover the significant essential information included in a document or collection of linked documents and then condense that information into a shorter version while maintaining the content's overall meanings. It solves the space issue that arises when storing a significant quantity of data and minimizes the amount of time needed to read the whole page. The challenge of automatically summarising text contains two sub-problems: one involves a single document and numerous files. In the case of the single document, the document in question serves as the input, and the information to be summarised is obtained from the single document in question. In many documents, multiple documents pertaining to a single subject are accepted as source file, and the result that is created must have some connection to that content.

Webology, Volume 18, Number 5, 2021
ISSN: 1735-188X
DOI: 10.29121/WEB/V18I5/52

A number of websites and services often create news feeds and article summaries using text summarising. Because of the packed schedules we have, it is now more important than ever before for us. Instead of reading a whole report and then summarising it on our own, we prefer to read condensed versions that highlight all the key elements. Therefore, a number of different efforts to mechanize the process of summarising have been attempted.

Automatic text summarization may be done in two different ways. 1) Text that is summarised abstractly, and 2) text that is summarised in an extractive manner. The term "extractive text summarization" refers to the process of taking key information or sentences from a given text file or the original document and summarising them. In a method known as extractive text summarization, analytical criteria are used to pick valuable, informative sentences. An abstractive text summary attempts to comprehend the source file, and re-produces the result in a small number of words by determining the most important idea presented in the input file. The process of extracting text summarization has been referred to as sentence ranking in a number of different research articles. ATS is an active field of study that can be described as the procedure of accessing valuable phrases or tidbits from a text corpus and incorporating them into a brief version of the file. It is possible to save both time and money by summarising a lengthy piece of writing. When it comes to time management, the individual reader might spend fewer minutes reviewing a document if the content is read in a form that has been summarised and captures the important elements of the document. A document summary tool may be used on several papers via newsgroups to collect the main information discussed in each document and present it in a condensed form. This allows the documents to be compared and contrasted more easily.

The quantity of textual data being transported from one device to another may be compressed through summarization, which can positively impact the overall cost efficiency of the process. It would be helpful for the user to view a summary form of a paper before deciding to acquire and go through the entire version of the manuscript or material. The present pace of data increase indicates that it will soon be necessary to have a tool that can produce shorter versions of transcripts as a utility to human readers. It will be necessary as soon as possible. The extractive method entails selecting the words and lines from the sources that are the most significant. After that, it brings together all of the pertinent lines to provide the Summary. In this particular instance, this means each and every line and phrase in the synopsis comes directly from the original material that is being summarised. The extractive text summarising process may be broken down into two stages: pre-processing and processing text. This article will discuss extractive text summarization based on a single document.

Background Studies

This section provides information on the procedures that were employed in the process of summarising the material. One of the subfields that fall under the umbrella of natural language processing is text summarization. The Summary of the input may be broken down into two categories: 1) Extractive summarization and 2) An abstract synthesis of the information. An extractive text summary involves selecting just the most important sentences from the source material. Utilizing paragraphs' linguistic and statistical characteristics enables one to single out this pivotal paragraph for selection. Abstract summarization [1][2] is the process of comprehending the

Webology, Volume 18, Number 5, 2021
ISSN: 1735-188X
DOI: 10.29121/WEB/V18I5/52

primary idea as well as the significance of the material that has been provided. It does this by using a linguistic approach and reading the text in order to identify the new notion that the document contains. The output that it creates will include the most recent and abridged version of the text, which will include the vital information that the document requires. In prior studies, summarization was performed on scientific articles based on the suggested parameters such as phrase frequency [3], word frequency [4], important phrase frequency [5], and location in the textual frequency. According to Reference [6], most of the early efforts were completed on a single document, with the primary focus being on the technical paper. Research on extractive summarization has been carried out by the author of Reference [7]. During his investigation, he isolated significant passages by computing the frequency of individual words and whole phrases; this provided a meaningful measurement of the relevance of the passages. The writers in [8] have conducted their study on extractive summarization while working with IBM. By analyzing the text's location, he could isolate the crucial sentences. The author has evaluated two hundred paragraphs to achieve his objective, and he has discovered that, in 85 percent of those sentences, the author has chosen the first subject, the primary theme phrase, and that the last sentence came in at 7 percent. The one deemed to be the most correct of these two sentences would be chosen.

A study on extracted summarization is carried out in [13] to extract essential sentences by leveraging two factors, position, and term frequency significance, which were collected from the works that came before it. These features were taken from the previous research. The existence of cue words and the text's structure are new elements that the author has included in the content. Extractive summaries have been generated using a variety of methods, like Luhn's recurrence count-based method and TextRank's graph-based method, where a text is modeled as a graph of phrases, with edges connecting sentences linked depending on a similarity assessment. LexRank is yet another graph-based method that utilizes the idea of eigenvectors as its foundation. Latent Semantic Analysis, also known as LSA, is a statistically-based method that applies Singular Value Decomposition to a text matrix D of size m n, where m corresponds to the number of phrases and n represents the total number of words. The goal of this method is to locate sentences within the document. The SumBasic algorithm is an example of a greedy search estimation strategy. It employs a frequency-based phrase to define word likelihood values, which helps reduce repetition.

The process of summarising a text may be broken down into two categories: extractive and abstractive summaries. Reusing parts of the original content, such as sentences, and integrating them into a summary is how an extractive summary is created. To produce an extractive summary, each phrase is assessed according to the importance of the information it contains, and then those sentences are reordered into a summary while the grammatical norms are maintained. One study that carries out extractive summarization is SumaRuNNer [14]. To produce an abstract summary, one must first understand the primary material and then generate meaningful phrases in a condensed version of the text. In this method, the model must have a deeper comprehension of the semantics of the text before it can compose a summary understandable to humans. Summarist paper, which contains modules that are able to conduct that sort of summarization, illustrated such a method by proposing it as an option [14].

Webology, Volume 18, Number 5, 2021
ISSN: 1735-188X
DOI: 10.29121/WEB/V18I5/52

Proposed Approach

In this strategy that we have provided, we will acquire a summary of the supplied input by utilizing an extraction method. We are accepting input in the form of text files (.txt). The steps of the proposed work are as follows,

- The file that is used as input is tokenized so that tokens of the phrases may be obtained.
- Once the tokenization process is complete, the stop words will be eliminated from the text. The terms that were left after elimination are those that are looked at as potential keywords.
- We are using the specific phrases as a source, so we are associating a portion of the tag with every key phrase.
- When we have finished with this phase of pre-processing, the next step is to determine the frequency of each keyword by determining the maximum frequency of the keyword and then calculating the frequency of each keyword based on that maximum frequency.
- Once the total number of times a set of keywords appears is divided by their maximum appearances, the result is the weighted frequency of the term.
- Finally, it computes the total number of weighted occurrences of each frequency. In the end, the summarizer will pull out the sentences that have a high weighted frequency, and the phrases that are pulled out will be transformed into digital file.

In the process of extractive summarising, the system receives source in the form of a text file. Tokenization of the text provided as input is then performed to locate the words included within the text. After that, the stop words are taken out of the text in order to filter it. And lastly, a tag denoting the token's part of speech is appended to each one. Once the parts-of-speech tag has been added to tokens or phrases, each unique weight will then be allocated to the tokens. The formula for computing the term weight (Wt) is as follows:

$$W_t = \frac{Term\ Frequency}{Total\ Term\ in\ Input} \quad (1)$$

After determining the maximum weight, the next thing to look at is the maximum weight of the token. The following is the formula that will be used to compute the document's weighted frequency (W_{tf}):

$$W_{tf} = \frac{Term\ Frequency}{Maximum\ frequency\ of\ the\ term} \quad (2)$$

After that, the frequencies are connected in position of related terms in the phrase, and the total of them is determined.

The weighted frequency is used as the basis for determining the rankings. The sentences have been arranged in a certain order according to their weighted frequency rankings, which are ordered from highest to lowest. The sentences are organized, with the most severe ones coming first. In conclusion, the summarizer will choose the phrases from the text that have the greatest priority. Then those selected sentences will be turned into an audio file (in digital audio). Figure 1 depicts the proposed architecture.

Webology, Volume 18, Number 5, 2021
ISSN: 1735-188X
DOI: 10.29121/WEB/V18I5/52

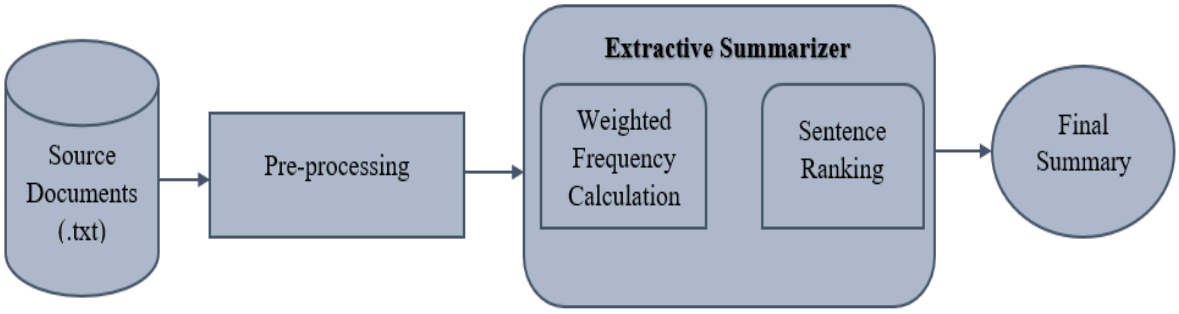

Figure 1: Proposed system architecture

1.1 Algorithm

Input: A text-based data representation is used as the input here.

Output: The final product is an appropriately summarised output text that is created and far shorter than the original. An audio version of this extracted and summarised output is created. The algorithm steps are given below,

Step 1: Analyzing the provided text, which also includes tokenizing the provided text

Step 2: The phrases are cleaned up by getting rid of the stop words.

Step 3: Every token has a part-of-speech category associated with it.

Step 4: The individual tokens are each given a weight, and then the weighted frequency of the tokens is computed by applying equations (1) and (2) in the appropriate Order.

Step 5: The rankings of the individual words are computed.

Step 6: In the end, the summarizer will pull out the phrases with the highest weighted Frequency to obtain an overview of the text and the summaries that have been Pulled out will be transformed into audio form.

Experimental Evaluation

The study evaluated a system with 20 phrases using five papers. The summarizer will only provide an output consisting of phrases whose ranks are higher than 8 when it is run. The text that was summarised is then produced in audio form. To construct extractive summarization, we relied on Python version 3.6 and NLTK. The proposed system is evaluated by contrasting the produced Summary with the standard gold summary of MS Word by employing the ROUGE metric, which then calculates the precision, recall, and F-score. This is done to determine how effective the suggested system is.

ROUGE, which stands for "Recall-Oriented Understudy for Gisting Evaluation," is a software program and a collection of metrics that have been developed primarily to analyze automated summarization, although they may also be used for machine translation. The metrics evaluate the quality of an autonomously generated summary or translation compared to reference summaries or translations, which are of a high standard and are made by humans. The analysis findings are presented in tabular form in Table 1, and a graphical representation may be found in Figure 2.

Webology, Volume 18, Number 5, 2021
ISSN: 1735-188X
DOI: 10.29121/WEB/V18I5/52

Table 1: Average performance measure comparison of MS word summarizer against proposed summarizer

Average performance measure

ROUGE Metric	MS word	Proposed system
Precision	0.82	0.85
Recall	0.75	0.80
F-score	0.77	0.81

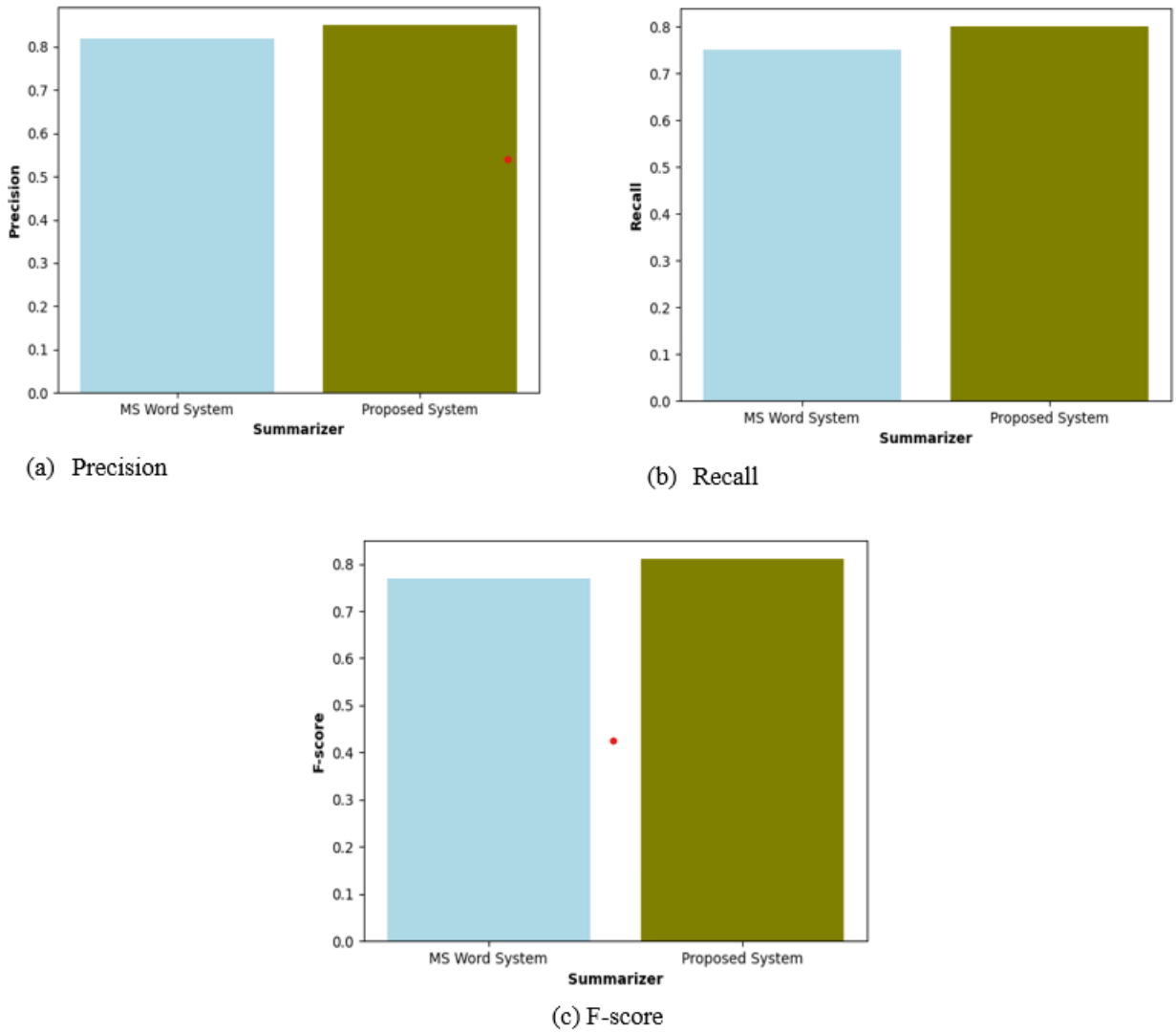

Figure 2: Average performance measure comparison of MS word summarizer against proposed summarizer

Webology, Volume 18, Number 5, 2021
ISSN: 1735-188X
DOI: 10.29121/WEB/V18I5/52

Conclusion

The process of automatically summarising text is a difficult one that consists of several smaller tasks within it. Each subtask can obtain summaries of high quality. Determining which paragraphs from the provided material contain relevant information is an essential aspect of extractive text summarization. In this study, we suggested an extractive-based text summary method by using a statistically new methodology depending on the ranking of the phrases. The proposed summarizer was responsible for selecting the sentences. After the sentences have been extracted, a summary of the text is created, and then that Summary is transformed into an audio file. The proposed work showed higher performance measures when compared to the conventional Ms word summarizer technique.

References

1. Nenkova, A.(2011). "Automatic summarization, Foundations andnTrends in Information Retrieval",5(2),103-233
2. Gupta,V and Lehal,G.s (2010). "A survey of text summarization extractive techniques." Journal of Emerging Technologies in Web Intelligence,2(3),258-268
3. Andrew Mackey and Israel Cuevas "AUTOMATIC TEXT SUMMARIZATION WITHIN BIG DATA FRAMEWORKS", ACM 2018
4. Yong Zhang, Jinzhi Liao, Jiyuyang Tang "Extractive Document Summarization based on hierarchical GRU", International Conference on Robots & Intelligent System IEEE 2018
5. Lili Wan "Extractive Algorithm of English Text Summarization for English Teaching" IEEE 2018
6. Anurag Shandilya, Kripabandhu Ghosh, Saptarshi Ghosh "Fairness of Extractive Text Summarization", ACM 2018
7. P.Krishnaveni, Dr. S. R. Balasundaram "Automatic Text Summarization by Local Scoring and Ranking for Improving Coherence", Proceedings of the IEEE 2017 International Conference on Computing Methodologies and Communication
8. Bagalkotkar, A., Kandelwal, A., Pandey, S., &Kamath, S. S. (2013, August). "A Novel Technique for Efficient Text Document Summarization as a Service", In Advances in Computing and Communications (ICACC), 2013 Third International Conference on (pp. 50-53). IEEE.
9. StanfordCoreNLP, From http://nlp.stanford.edu/software/corenlp.shtml
10. Apache Open NLP. From http://opennlp.apache.org/
11. Natural Language Toolkit. From http://nltk.org/
12. Rapid Miner. From Available: http://rapidminer.com/
13. Gambhir, M.; Gupta, V. Recent automatic text summarization techniques: A survey. Artif. Intell. Rev. 2016, 47, 1–66. [CrossRef]
14. Nallapati, R.; Zhai, F.; Zhou, B. Summarunner: A Recurrent Neural Network-Based Sequence Model for Extractive Summarization of Documents. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; pp. 3075–3081