Webology, Volume 18, Number 5, 2021
ISSN: 1735-188X
DOI: 10.29121/WEB/V18I5/44

Employing a Similarity-based Clustering approach for effective text summary

Vikas Tripathi[1], Dibyahash Bordoloi[2]

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

ABSTRACT

With so much new data being created every day, summarising documents is becoming more important. Document summarising facilitates comprehension of the text content without reading the whole document. Text summary offers a structure for intentionally condensing one or more text files. It is an important technique for discovering similar content in vast text libraries or on the Web. It is also vital to retrieving the data so that the material is interesting to the user. Extractive summarising and abstractive summarization are the two primary techniques for text summarization. Extractive summarization is a technique for generating summaries by selecting phrases from a text source and ranking them according to their importance. The abstractive summarization approach captures the document's key ideas and renders them in abstract form using natural language. Based on these two methodologies, several strategies for summarising have been created. Several approaches only work with certain languages. This article discusses numerous strategies depending on extractive and abstractive summary approaches and the limitations of these methodologies.

Keywords: Document Summarization, Extractive approach, Abstractive approach, Text summary, Information Extraction system.

INTRODUCTION

As more information becomes available, it becomes harder for individuals to discover what they're looking for, conduct targeted searches, and get a big-picture understanding of what's significant, influential, and relevant. Countless individuals in today's era of information are scouring the Internet for beneficial knowledge, yet seldom do they get what they need in a single document or web page. They might get several websites due to their search [5]. This topic has a novel solution related to data mining and machine learning that recovers query-oriented details from a vast group of offline articles and displays it to the reader as a single document. Therefore, automated summarization is an essential study topic in Natural Language Processing (NLP). Automatic summarising enables single-document and multi-document summarization tasks [3].

Multi-document merger refers to the combining of information from various materials. Data may be

Webology, Volume 18, Number 5, 2021
ISSN: 1735-188X
DOI: 10.29121/WEB/V18I5/44

available in an unstructured or structured form, and it is often necessary to construct a summary from multiple files in a short amount of time; hence, the multi-document merging approach is beneficial. Multi-document summary generates compact, accurate, and understandable information. The purpose of a concise summary is to facilitate a detailed search and minimize effort cum time by highlighting the pertinent facts.

There is a pressing need to create automated text summarising technologies so that insight may be readily extracted from the massive amounts of unstructured textual data now traversing the digital environment. Present-day humans have instantaneous access to vast stores of data. However, a large portion of this data is unnecessary, irrelevant, and may not even communicate the desired meaning. If you want to find a certain piece of information in an internet news story, you may have to spend a significant amount of time sifting through the article's substance and ignoring the fluff. As a result, it is becoming more important to utilize computerized text summarizers that can extract meaningful information from texts while excluding fluff. When used effectively, summarising may improve content readability, cut down on research time, and squeeze more content into a given space. Nowadays, Text summarization is of increasing significance. One explanation is that the need for unintentional text summaries has increased due to the increasing content proliferation. It is challenging for humans to summarise large text texts manually. Textual content is abundant on the Web.

Moreover, the Web offers more details than is often necessary. Hence, a problem of repetition arises: searching through a huge number of papers for similar types of documents is a tiresome operation [3]. The purpose of a text summary is to minimize the source content to an abbreviated form while preserving its meaning and substance. Making a summary would be ineffective if all of the phrases in a book were of equal importance. Each issue is seen and discussed from numerous viewpoints inside a single paper that combines diverse thoughts. While the primary objective of a concise summary is to facilitate data search and save the effort of individuals by linking to the pertinent input content, a multi-document summary includes the necessary content so that individuals do not need to visit the source content when refining is necessary. In this work, many strategies for sentence-based extractive summarization, as well as numerous similarity metrics and their comparisons, have been explored.

Extractive summarization focuses on obtaining the most significant and pertinent phrases from a text while minimizing repetition in summary. It is generated by the input text's word-for-word recycling portions (words, phrases, and so on). Typically, search engines provide extractive summaries of online sites. In contrast to picking the best representative existing snippets, abstractive text summary employs natural language processing methods to understand the text and produce fresh summarised content. This strategy involves rephrasing information from the original text. However, it is more difficult to employ due to complications such as semantic representations. For instance: Book reviews; using this strategy, it is possible to create a summary of the book The Lord of the Rings.

Related works

An enhanced technique of automated text summary for online material using lexical chains with

Webology, Volume 18, Number 5, 2021
ISSN: 1735-188X
DOI: 10.29121/WEB/V18I5/44

semantically related words presents an improved way of extractive text summarization for articles by upgrading the existing lexical chain approach to generate more pertinent content. Later, the authors analyzed the methods for extracting phrases from the input(s) depending on the lexical chain distributions and constructed a transition probability distribution generator (TPDG) for n-gram keywords that train the features of the given key terms from the training set. The system also includes a novel way of automated keyword extraction based on the Markov chains algorithm. Only unigrams are used from the retrieved n-gram keywords to form the lexical chain. Only unigrams are used from the retrieved n-gram keywords to form the lexical chain [1]. Sentences are ranked using the Top-K ensemble ranking algorithm. Term frequency and inverse document frequency (TF-IDF) are employed for counting words and extracting features at the word level. In article [2], the author initially extracted various candidate summaries by suggesting different approaches for enhancing the quality of the summary. Utilizing bilingual characteristics, they then developed a novel ensemble ranking strategy for scoring the candidate summary. On a benchmark dataset, numerous experiments have been carried out.

Automatic text summarising within the architecture for big data illustrates how to analyze huge data sets in parallel to overcome the volume issues associated with big data and provide a summary leveraging a sentence rating mechanism. TF-IDF is employed to retrieve document features. [3] MapReduce and Hadoop are employed to handle massive amounts of data. MapReduce and Hadoop are used to handle massive amounts of data. In order to extract meaningful information from large documents, a hierarchical gated recurrent unit (GRU)-based approach suggests a two-stage process: 1) Extraction of key phrases using the Levenshtein distance calculation. 2) Recurrent neural network for document summarization. To extract important sentences, the model first comes up with a hybrid sentence metric of similarity by merging sentence vector and Levenshtein distance and then incorporates this into a graph model. In phase two, GRU is constructed as a fundamental block, and the representation of the complete text based on Latent Dirichlet Allocation (LDA) is introduced as a summarization-supporting feature[4].

The extractive algorithm for summarising English text for English instruction is based on semantic association principles. Vectors of semantic association rules are used to summarise texts. In this work, we implement semantic relevance evaluation and feature extraction of terms in English abstracts by mining relative characteristics amongst English text words and phrases [5]. The integrity of extractive text summarization explores the concept of equity of text summarization algorithms for the first time. The author demonstrates that when summarising datasets with a sensitive characteristic, one must validate the summary's fairness. With the emergence of neural network-based summarization algorithms (which require super-wised learning), the issue of fairness has taken on an even greater degree of importance. According to the authors, this study will provide intriguing research issues, such as the development of algorithms to assure some degree of justice in summary [6]. To enhance the readability of the summary, an automated feature-based extractive heading-wise summarizer is provided via a technique that uses local grading and rating to increase coherence. Employing local rating and local scoring, it creates a heading-by-heading synopsis of the input material. To quickly locate the information you need, just scan the headings of a page. The experiment's results unmistakably demonstrate that heading-wise summarizer outperforms principal summarizer, MS-word summarizer, unrestricted summarizer, and auto summarizer in terms of

Webology, Volume 18, Number 5, 2021
ISSN: 1735-188X
DOI: 10.29121/WEB/V18I5/44

Accuracy, recall, and f-measure [7].

Van Britsom, in his study on data merging, developed a method based on the NEWSUM Algorithm. It's an algorithm that creates a summary of related texts by partitioning a collection of documents into smaller groups. The process consists of three steps: finding the topics, modifying them, and summarising them using clusters [8]. Abstractive summarization and extractive summarising are the major essential ways to summarise texts automatically. They are represented in Anusha Banalkotkar's unique methodology for efficient text document summarization as a service. This work covers many complex summarising methods, including coherent, legible, understandable, multi-disciplinary, and machine learning [9].

Proposed System

The proposed system utilizes many modules to construct a summary of a set of documents automatically. Most existing automatic text summarising systems create summaries through an extraction mechanism. Extraction summaries are often created using sentence extraction methods. Assigning a numerical measure, or sentence score, to a sentence for the summary and then selecting the best sentences to construct a document summary, depending on the compression rate is one way to generate acceptable sentences. In the extraction process, the compression rate plays a crucial role in determining how closely the summary's length matches that of the original text. As compression rates rise, the summary grows in size, and more trivial information is crammed in. As the compression rate rises, less information may be included in the summary. In reality, the readability of the summary is satisfactory when the compression rate is minimum.

The previous system had limitations, such as the fact that it could only read text files as input. If we try to enter a file format other than plain text (such as a PDF or a word document), we get an error notice saying so. To solve these issues, we presented a new system that accepts input from text, PDF, and word files. Following are the three primary stages involved in the system. Figure 1 depicts the architecture of the proposed approach.

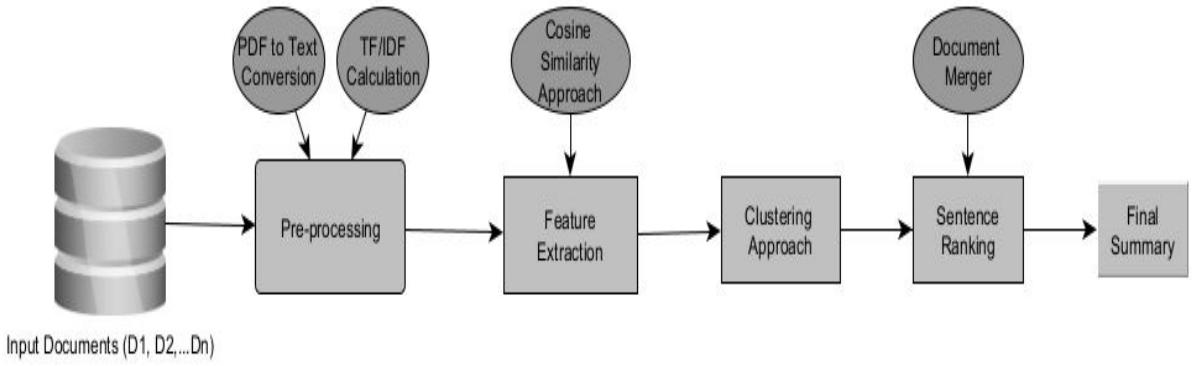

Figure 1: Proposed architecture

Pre-Processing

At this stage, we conduct the following operations on numerous input documents:

Webology, Volume 18, Number 5, 2021
ISSN: 1735-188X
DOI: 10.29121/WEB/V18I5/44

(i) Tokenization:
Tokenization separates the text into individual lexical terms delineated by punctuation, such as white space, commas, dashes, dots, and so on [3].

(ii) Stop Word Removal:
When dealing with natural language data, stop words are eliminated either at the outset or at the end (text). Keyword searching benefits from the elimination of stop words [2].

(iii) Stemming Suffixes:
This process identifies the root form of each word irrespective of its prefix and suffix.
In this case, we employ the removal of enlisted suffixes as a subject identification technique.
For instance,
Works and Working have their root form in the term "Work"; where "s" and "ing," which are appended to the subject play, should be eliminated for clarity.

Feature Extraction
In this step, we use a cosine similarity metric to determine which documents may be extracted based on their similarities.

Cosine similarity calculates the value in the range [-1, 1] that best characterizes the degree of similarity between two phrases or documents. Specifically, you see a Cosine Similarity. The formulas used to get the TF and IDF from the cosine similarity matrix are given in equations (1), (2), and (3), respectively,

TF (term, document) = Frequency of term / Number of Terms

$$TF_i = \frac{n_i}{\sum_1^n k} \quad (1)$$

Inverse document frequency (IDF) determines how frequent or uncommon a certain word is across all texts. The IDF (term, document) is learned by calculating the logarithm of the ratio between the total number of documents and the number of documents that include the word under study.

IDF (term, document) = log (Total Count of Document / Count of Document comprising the given term)

$$IDF_i = \log\left(\frac{D_n}{Count_D\{t \in D\}}\right) \quad (2)$$

The TF-IDF of a given the word is calculated by multiplying its TF value by its IDF value. TF-effectiveness IDF grows with both the frequency with which a word appears in a given document and the breadth of contexts in which that phrase is used.

$$\text{TF-IDF} = \text{TF} * \text{IDF} \quad (3)$$

Sentence Ranking and Summary Generation:
When a document's relevance and similarity to other documents are verified, the relevant phrases are extracted and combined using a cosine similarity metric. Finally, the summarizer will choose the

Webology, Volume 18, Number 5, 2021
ISSN: 1735-188X
DOI: 10.29121/WEB/V18I5/44

highest-scoring sentences from the text and combine these phrases into a single summary.

Experimental Results

It will be compared with the current system to demonstrate the efficacy of the suggested technique, and the performance analysis will be reviewed. To analyze our results, we will develop our dataset and include data from the industry-standard OpinosisDataset1.2.

The two clustering approaches, WeightSum (which creates a cluster based on the weight of documents) and threshold clustering (which uses a predetermined threshold value), are compared and contrasted in Table 1, and their visual representation is depicted in Figure 2. There is a recommendation for the clustering method known as threshold clustering. This section presents the outcomes and analyses of OpinosisDataset1.2.

Table 1: Performance of Clustering Approach in terms of Accuracy

Clustering Approach	**Accuracy (%)**
WeightSum	82
Threshold (Proposed)	**95**

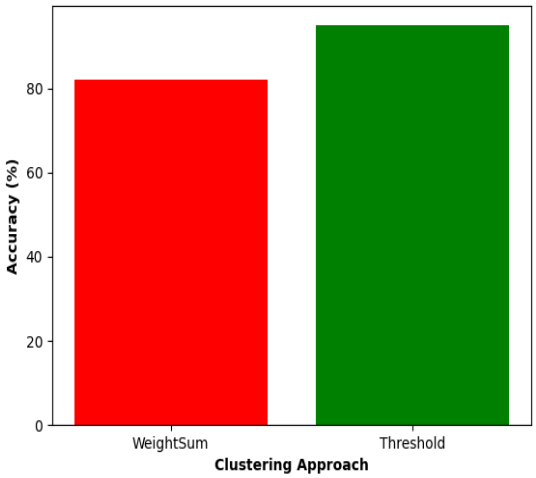

Figure 2: Performance of Clustering Approach in terms of Accuracy

When a measurement, computation, or specification results match the expected value, we say that it is accurate (true). For the most part, the proposed methodology can guarantee Accuracy of 95%. In Table 2, we can see the proposed system's Accuracy on a range of document counts (5, 10, 20), which is graphically shown in Figure 3. All files range in size from 2kb to 10kb.

Webology, Volume 18, Number 5, 2021
ISSN: 1735-188X
DOI: 10.29121/WEB/V18I5/44

Table 2: Accuracy for different numbers of input documents

Number of Documents	Accuracy (%)
5	94.33
10	90.57
20	85.13

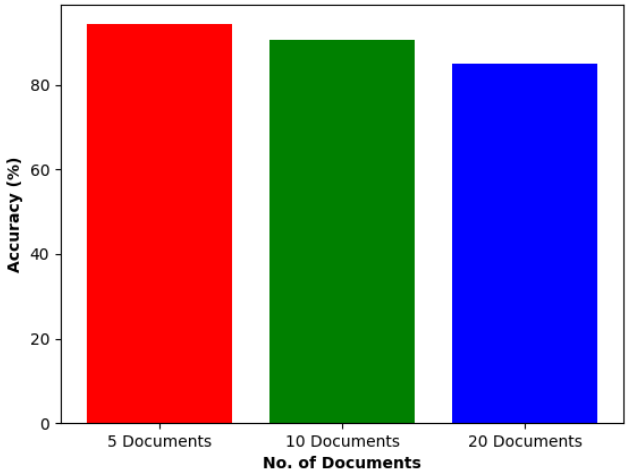

Figure 3: Accuracy with respect to different numbers of input documents

Conclusion

Information may be obtained by generalizing from previously acquired context. Today, we need to quickly distill insights from growing troves of data, whether they're organized or not. We suggest a new system that gets rid of the problems with the old one. The project's goal is to employ text mining's various tools to develop a method that will help the document's author with the document's rough structure, which will help express the document's key points. Therefore, we resort to utilizing document merger summarization for this work. It saves us time and produces quality results.

References

1. HtetMyet Lynn 1 , Chang Choi 2 , Pankoo Kim "An improved method of automatic text summarization for web contents using lexical chain with semantic-related terms", Springer-Verlag Berlin Heidelberg 2017
2. Xiaojun Wan 1 , FuliLuo 2 , Xue Sun Songfang Huang3 , Jin-ge Yao "Cross-language document summarization via extraction and ranking of multiple summaries" Springer-Verlag London 2018
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

Webology, Volume 18, Number 5, 2021
ISSN: 1735-188X
DOI: 10.29121/WEB/V18I5/44

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. Ferreira, Rafael, Luciano de Souza Cabral, Rafael DueireLins, Gabriel Pereira e Silva, Fred Freitas, George DC Cavalcanti, Rinaldo Lima, Steven J. Simske, and Luciano Favaro. "Assessing sentence scoring techniques for extractive text summarization." Expert systems with applications 40, no. 14 (2013): 5755-5764.
10. Gupta, V. K., &Siddiqui, T. J. (2012, December). "Multi-document summarization using sentence clustering", In Intelligent Human Computer Interaction (IHCI), 2012 4th International Conference on (pp. 1-5). IEEE.
11. Min-Yuh Day Department of Information Management Tamkang University New Taipei City, Taiwan myday@mail.tku.edu.tw Chao-Yu Chen Department of Information Management Tamkang University New Taipei City, Taiwan susan.cy.chen@gmail.tw "Artificial Intelligence for Automatic Text Summarization",2018 IEEE International Conference on Information Reuse and Integration for Data Science
12. Xiaoping SunandHaiZhuge*, Senior Member, IEEE Laboratory of Cyber-Physical-Social Intelligence, Guangzhou University, China Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, University of Chinese Academy of Sciences, Chinese Academy of Sciences, China System Analytics Research Institute, Aston University, UK "Summarization of Scientific Paper through Reinforcement Ranking on Semantic Link Network",IEEE 2018
13. Ahmad T. Al-Taani (PhD, MSc, BSc) Professor of Computer Science (Artificial Intelligence) Faculty of Information Technology and Computer Sciences Yarmouk University, Jordan. ahmadta@yu.edu.jo "Automatic Text Summarization Approaches",IEEE 2017
14. AlokRanjan Pal Dept. of Computer Science and Engineering College of Engineering and Management, KolaghatKolaghat, India chhaandasik@gmail.com DigantaSaha Dept. of Computer Science and Engineering Jadavpur University Kolkata, India neruda0101@yahoo.com "An Approach to Automatic Text Summarization using WordNet", IEEE 2014
15. Prakhar Sethi1, Sameer Sonawane2, Saumitra Khanwalker3, R. B. Keskar4 Department of Computer Science Engineering, Visvesvaraya National Institute of Technology, India 1 prakhar.sethi2@gmail.com, 2 sameer9311@gmail.com, 3 theapogee2011@gmail.com, 4 rbkeskar@cse.vnit.ac.in "Automatic Text Summarization of News Articles", IEEE 2017

Webology, Volume 18, Number 5, 2021
ISSN: 1735-188X
DOI: 10.29121/WEB/V18I5/44

16. Yue Hu and Xiaojun Wan "PPSGen: Learning-Based Presentation Slides Generation for Academic Papers", IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 27, NO. 4, APRIL 2015
17. Daan Van Britsom, AntoonBronselaer, and Guy De Tre "Using Data Merging Techniques for Generating Multidocument Summarizations", IEEE TRANSACTIONS ON FUZZY SYSTEMS, VOL. 23, NO. 3, JUNE 2015
18. NingZhong, Yuefeng Li, and Sheng-Tang Wu "Effective Pattern Discovery for Text Mining", IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 24, NO. 1, JANUARY 2012