World Journal of Advanced Research and Reviews
eISSN: 2581-9615 CODEN (USA): WJARAI
Cross Ref DOI: 10.30574/wjarr
Journal homepage: https://wjarr.com/

(Research Article)

A boundary-based tokenization technique for extractive text summarization

Nnaemeka M Oparauwah *, Juliet N Odii, Ikechukwu I Ayogu and Vitalis C Iwuchukwu

Department of Computer Science, School of Information and Communication Technology, Federal University of Technology, P.M.B 1526, Owerri, Nigeria.

World Journal of Advanced Research and Reviews, 2021, 11(02), 303–312

Publication history: Received on 25 June 2021; revised on 10 August 2021; accepted on 12 August 2021

Article DOI: https://doi.org/10.30574/wjarr.2021.11.2.0351

Abstract

The need to extract and manage vital information contained in copious volumes of text documents has given birth to several automatic text summarization (ATS) approaches. ATS has found application in academic research, medical health records analysis, content creation and search engine optimization, finance and media. This study presents a boundary-based tokenization method for extractive text summarization. The proposed method performs word tokenization by defining word boundaries in place of specific delimiters. An extractive summarization algorithm was further developed based on the proposed boundary-based tokenization method, as well as word length consideration to control redundancy in summary output. Experimental results showed that the proposed approach enhanced word tokenization by enhancing the selection of appropriate keywords from text document to be used for summarization.

Keywords: Boundary-based; Tokenization; Extractive; Automatic; Text; Summarization

1. Introduction

There is plenty of information available in every sphere of human life most especially on the internet. Important information can be considered and isolated by creating summary from the available pool of information. As textual content continue to grow, it has become a burden on the research community to manage this growth by implementing smarter and improved text summarization solutions [1]. In view of this, the research community, is constantly developing new approaches for automatic text summarization (ATS). ATS is the use of a computer program in creating shorter text from an original text document without losing the most important content of the original document [2]. A summary therefore, is a shorter text that bears vital and relevant information from an original longer text [3].

The general classification of automatic text summarization techniques are of two types; extractive and abstractive techniques. The extractive method selects vital and important sentences and paragraphs from an original document and concatenates them to produce the summary [4]. Abstractive techniques on the other hand requires an intuitive understanding of the main concepts in a document which leads to a new representation of an original text document as summary [5].

Extractive text summarization is predominant, hence, this study explored the technique, and considered the importance of preprocessing stage of the natural language processing pipeline during extractive automatic text summarization as a vital stage in the summarization process. This paper therefore, attempts to improve the process of word tokenization by implementing a boundary-based tokenization method. The objective is to ensure that the right word tokens are isolated at the early stage of ATS in view of sentence selection for the final summary. Proper word tokenization leads to

* Corresponding author: Nnaemeka M Oparauwah
Department of Computer Science, School of Information and Communication Technology, Federal University of Technology, P.M.B 1526, Owerri, Nigeria.

the selection of good keywords, which in turn results to the selection of appropriate sentences for the final summary [6].

2. Extractive Text Summarization

In simple term, extractive summarization creates summaries by simply extracting important sentences from the original text document without changing the content of original text [4]. More of what an extractive summarization method does is to make reliable decision for each sentence whether or not it will be included in the generated summary. One of the methods to obtain suitable sentences is to assign some numerical measure of a sentence for the summary called sentence scoring and then select the best sentences to form document summary [7].

The extractive technique is grouped into two stages, viz; pre-processing stage and processing stage. Pre-Processing deals with the structural interpretation of the original text which usually involves some morphological analysis in terms of word/sentence boundary identification, tokenization, stop-word removal, stemming/lemmatization etc. The processing stage identifies and calculates sentence features influencing the relevance of sentences, assigns weights to these features, scores and ranks weighted sentences, after which the highly ranked sentences are selected for the final summary [8].

2.1. Sentence Features for Extractive Text Summarization

Some sentence features of text(s) are important considerations when creating an automatic summary. These features are determinants to sentence selection or rejection in automatic text summarization. They include: Title word, Content word (Keyword), Sentence length, Sentence position, Proper Noun, Upper-case word, Cue-Phrase, Font based, Pronouns, Presence of non-essential information, Sentence-to-Sentence Cohesion [8].

2.2. Evaluation of Summary

Evaluation of summaries is an important aspect of text summarization. However, evaluating summaries, either manually or automatically is usually a tough task, of which the main difficulty comes from the impossibility of building a fair standard against which the results of summarization systems can be compared [9]. Furthermore, and because textual data are unstructured, it is also very hard to determine what a correct summary is. Approaches to evaluation are classified into extrinsic and intrinsic. Extrinsic classification judges an auto-generated summary on how well it contributes to the accomplishment of a particular task. Intrinsic on the other hand, judges summary quality directly without reference to any particular task. Evaluation of performance of automatic summary can also be measured using precision, recall and F-score [8]. Precision is the number of sentences found in both the system and in an ideal summary divided by the number of sentences in the system summary. Recall is the number of sentences found in both system and ideal summaries divided by the number of sentences in the ideal summary. F-score is a combo of the two measures [10].

A set of metrics called Recall Oriented Understudy of Gisting Evaluation (ROUGE) was introduced in 2004 at the Document Understanding Conference (DUC) [11], it became the standard of automatic evaluation of the summaries, and gives a score based on the similarity in the sequences of words between a human-written model summary and the machine summary. However, ROUGE has been demonstrated to be less sufficient as a precise indicator for evaluation of a quality summary [12]. In recent studies, human evaluators have been shown to be effective in summary evaluation. This was demonstrated in IBM Science Summarizer project [13], where 12 authors from the NLP community were tasked to evaluate summaries of two papers that they co-authored. Another study where human evaluation technique have been successfully applied is [14]. In fact, a general policy to evaluate the quality of a summarization system is absent in most existing models [15]. Authors and researchers usually provide different approaches for summary evaluation. In some, summary quality is evaluated grammatically and based on its relevance to a particular user. If the system summary output is satisfactory then it satisfies the need of that user [15].

2.3. Related Works

In a recent study by Manju *et al.* [16] extractive text summarization technique was adopted to make sense of the sensitive part of the document by neglecting the irrelevant and redundant sentences. The researchers proposed a framework for extracting summary from multiple documents in the Malayalam Language by a sentence extraction algorithm that selects the top ranked sentences which also has maximum diversities. Since the work was based on multi document summarization, a graph-based method (TextRank Algorithm) which has been demonstrated by Mihalcea and Tarau [17] to perform well was used to rank and select sentences to be included in the summary.

A study by Vaghasiya [18] attempted to simplify scholarly literatures by proposing a novel model that has specific number of input files with a uniform structure, aiming to summarize, and then simplify into one document. Firstly, the system establishes the structure of the papers and also identifies text from some critical sections of the documents. After extraction of the texts, it is further tokenized and weighed based on relevance before constructing the required summary. The constructed summary further goes through separate simplification system where complex words are further broken into simpler terms. The proposed system was implemented as a web application and uses four different techniques to summarize the text (Data Aggregation, Information Extraction, Processing, Summarization and Evaluation) and two different techniques to simplify a given text (Context Breakdown and Term Re-writing) [18].

In another study, Nguyen *et al.* [6] proposed a model for improving the quality of the scoring step, which is factored in, during sentence selection. It was stated that a good scoring step for sentences could increase chances of extracting high-quality sentences for summaries. The proposed model takes advantage of local information (inside a single document) and global information (on the whole corpus). The combination of these information permits defining a rich set of sentence features used for learning [6]. Under a learning-to-rank formulation, the researchers' model learns to estimate the level of importance for different sentences, after which it performs sentence ranking and summaries are finally extracted by isolating the highest ranking sentences with diversity.

Bashir *et al.* [19] conducted a study which developed a feature extraction model using Naïve Bayes model to automatically summarize Hausa language texts. The study attempted automatic summarization using 10 Hausa Language texts as dataset with some sentence features. The sentence features adopted in the study for the summarization process are keyword, title and cue phrases. Although the results of the study were within objectives, morphological structure of the Hausa language was not considered. The researcher further concluded that automatic text summarization attempted on Hausa Language dataset is better if morphological analysis is considered.

Oyekan *et al.* [3] studied the imminent communication gap in children news rendering and therefore, proposed a text summarization technique in an attempt to bridge the gap and meet with the challenges. The study attempted the use of a suitable parsing model that can summarize texts based on variations in children age-grade. The age grades considered were the Lower and Middle School levels). The system was able to summarize texts with character length of about 20,000 for lower school level children, and about 500,000-character length for middle school level children.

Selvani Deepthi [20] performed extractive text summarization using modified weighing and sentence symmetric feature". In this work, the author laid emphasis on summarization of different research papers of various fields. The researcher showed three distinctive algorithms for summarization with results detected for each algorithm. The author perceived that sentence score and feature scores used for the summarization process are determined on the basis of the statistical approaches. The researcher used extractive approaches to overcome few challenges like handling large amount of text data as well as reducing the presence of unnecessary sentences in the summary.

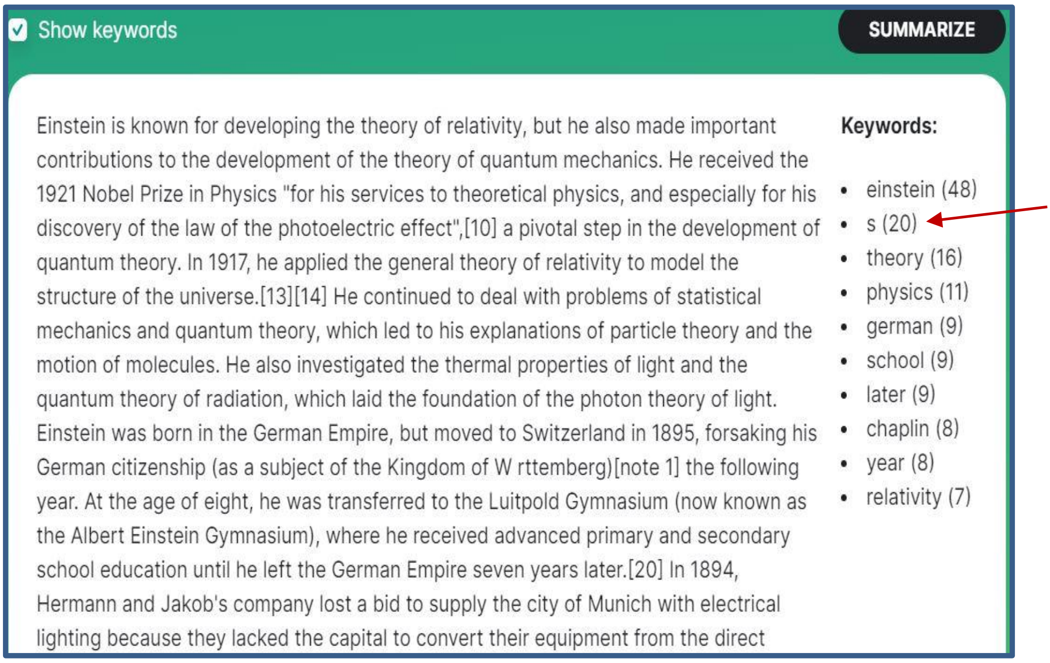
3. Material and methods

3.1. Design Description

This study, in proposing a boundary-based tokenization method for text preprocessing during summarization examined an anonymous existing system model and found the need to improve its text preprocessing stage. As seen in figure 1, the existing system which performs automatic text summarization by extraction selects letter 's' as a keyword during summarization. Such scenario is presumably a product of the delimiter based split method of tokenization. The delimiter based split method tokenizes by splitting words based on two delimiters (white space and comma). This degenerated to the selection of letter 's' as a keyword.

Keywords have been established in scientific literature to be vital in representing the main discourse of a text or document, and also one of the vital sentence features to be considered when selecting sentences to be included in the final summary [8]. A keyword is explained by Siddiqi and Sharan [21] to be a word that succinctly and accurately describes the subject, or an aspect of the subject, discussed in a document. Letter 's' therefore, is not a keyword and should be adequately cleaned off during the preprocessing stage of ATS.

We thus, modelled in mathematical terms, the delimiter based split method of tokenization and further attempt an improvement by introducing a boundary-based tokenization technique.



Einstein is known for developing the theory of relativity, but he also made important contributions to the development of the theory of quantum mechanics. He received the 1921 Nobel Prize in Physics "for his services to theoretical physics, and especially for his discovery of the law of the photoelectric effect",[10] a pivotal step in the development of quantum theory. In 1917, he applied the general theory of relativity to model the structure of the universe.[13][14] He continued to deal with problems of statistical mechanics and quantum theory, which led to his explanations of particle theory and the motion of molecules. He also investigated the thermal properties of light and the quantum theory of radiation, which laid the foundation of the photon theory of light. Einstein was born in the German Empire, but moved to Switzerland in 1895, forsaking his German citizenship (as a subject of the Kingdom of W rttemberg)[note 1] the following year. At the age of eight, he was transferred to the Luitpold Gymnasium (now known as the Albert Einstein Gymnasium), where he received advanced primary and secondary school education until he left the German Empire seven years later.[20] In 1894, Hermann and Jakob's company lost a bid to supply the city of Munich with electrical lighting because they lacked the capital to convert their equipment from the direct

Keywords:

- einstein (48)
- s (20)
- theory (16)
- physics (11)
- german (9)
- school (9)
- later (9)
- chaplin (8)
- year (8)
- relativity (7)

Selecting letter 's' as keyword

Figure 1 Snapshot of existing system output

3.2. Modeling the tokenization problem

The delimiter-based split function splits the set of strings, L, into subsets, l_i, based on the set of delimiters, D. Given that:

$$word(w_i) = f(l_i, \partial_i) \text{ such that } l_i > 0 \; eq\ 1.$$

$$word_{total}(w_T) = \sum_{i=0}^{n} f(l_i, \partial_i) \; eq\ 2.$$

$$\sum_{i=0}^{n} l_i = \sum_{i=0}^{n} \partial_i \; eq\ 3.$$

The delimiter, D, is a set of 2 elements, ∂_1, ∂_2

D = (∂_1, ∂_2),

where; ∂_1 = whitespace

∂_2 = Comma

Tokenization based on equation (1) applied in the sentence, "*I am John's brother, James*" would give:

l_0	∂_1	l_1	∂_1	l_2	∂_1	l_3	∂_2	l_4	∂_1
I		am		John's		Brother	,	James	

$$w_i = f(l_i, \partial_i)$$
$$w_0 = l_0 , \partial_1 = I$$
$$w_1 = l_1 , \partial_1 = am$$
$$w_2 = l_2 , \partial_1 = John's$$
$$w_3 = l_3 , \partial_2 = brother$$
$$w_4 = l_4 , \partial_1 = James$$

Notice the error in w_2 , where the $f(l_i, \partial_i)$ identifies the possessive *John's* as a single word. This poses a problem down the line in the NLP preprocessing, leading to the kind of redundancy noticeable in the output of the existing system. We solve this problem by implementing a boundary-based tokenization method. This method gives room for expansion of word boundaries in a text, and not just by two delimiters.

3.3. Improving NLP Preprocessing Pipeline

Implementing a *Boundary Based Tokenization* Method in place of a Delimiter Based Split Function in Python. We define a word, w_i , thus:

$$w_i \ = f(b_i \, , l_i \, , b_{i+1}) \ \Leftrightarrow \ l_i > 0 \ eq\ 4.$$

Where the set of boundaries, B, is defined thus;

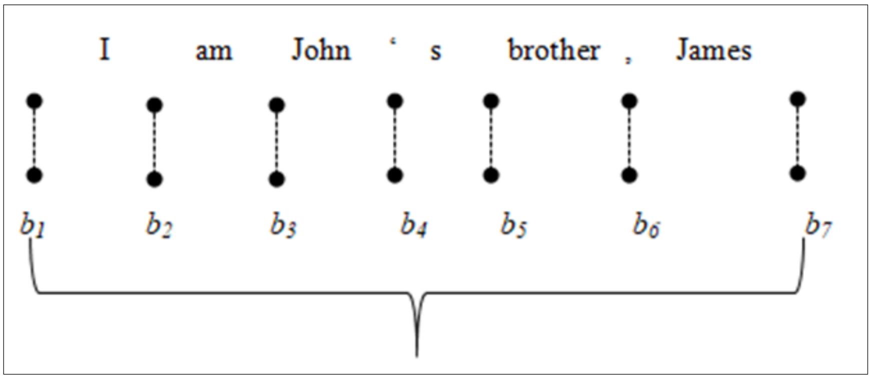
$$B = \ (L)' = \{b_i, b_{i+1} \, , b_{i+2}\} \ eq\ 5.$$

$$l_i \ \in L;$$

$$L = \{a - z, A - Z, -\} \ eq\ 6.$$

Such that L ensures that l_i is a string of letters, outside of which any element would serve as a boundary.

Applying the boundary-based method in the sentence “I am John’s brother, James”



‘I’, ‘am’, ‘John’, ‘s’, ‘brother’, ‘James’

Having isolated the string, ‘s’ using the boundary-based method, it can then be eliminated down the line in the preprocessing pipeline by filtering using some defined conditions abstracted in code.

3.4. Proposed Summarization Algorithm

Step 1: Perform word tokenization based on proposed boundary-based method

Step 2: Calculate word frequency, f_w, of words, w, appearing in the document, T

$$fw_i = \sum_{i=0}^{n} w_i + 1$$

Where n = number of times W_i appears in T

Step 3: Find word with maximum frequency, W_{max}

Step 4: Find word density distribution, $Pw_i = \frac{W_i}{W_{max}}$

Step 5: $\forall \ w_i \ \in \ s_j$

$Ps_j = \sum Pw_i$

Where; s_j = Sentence in Document; Ps_j= Sentence density distribution/score

```
Step 6: Rank $s_j$ by their scores

# To take care of redundancy, set word length in limit;

Step 7: Set word length in limit, q, for $s_j$ to be considered

Do {

    If length ($S_j$) > q: Pass

    else:    add $S_j$ $\longrightarrow$ Summary Text

}

Where q = integer
```

3.5. Evaluation method

The human evaluation method was adopted as has been proven effective by [13, 14]. This evaluation was conducted in form of a basic Turing test. This is a test of a machine's ability to exhibit intelligent behavior equivalent to, or indistinguishable from, that of a human. Turing test proposes that a human evaluator would judge natural language conversations between a human and a machine designed to generate human-like responses. The evaluator would be aware that one of the two partners in conversation is a machine, and all participants would be separated from one another [22]. If the evaluator cannot reliably distinguish the output of the machine from that of the human, then the machine is judged to have passed the test. The test results do not depend on the machine's ability to give accurate and perfect answers to questions, but to what degree its answers would resemble those a human would give. Hence, text documents of varying lengths (800 – 2000 words) were selected and summarized using the existing and new system respectively. The outputs of the two systems were presented to two different groups of assessors who were tasked to semantically assess the various outputs. 50% of the assessors were English Language and Linguistics lecturers, while the other 50% were randomly selected lecturers of other disciplines. The responses (semantic scores) were reported on a scale of 0-10 and further analyzed with R programming software (evaluation results shown in table 1). The results of the evaluation are also discussed.

4. Results

Table 1 Table of evaluation results

	Input/text length	Text Summarizer	Summary Length	Semantic Score
1	877	A	218	5
		B	271	5
2	1155	A	273	6
		B	300	4
3	1206	A	300	7
		B	300	5
4	1272	A	316	8
		B	300	6
5	1357	A	329	7
		B	300	5
6	1482	A	342	8
		B	300	5
7	1564	A	365	6

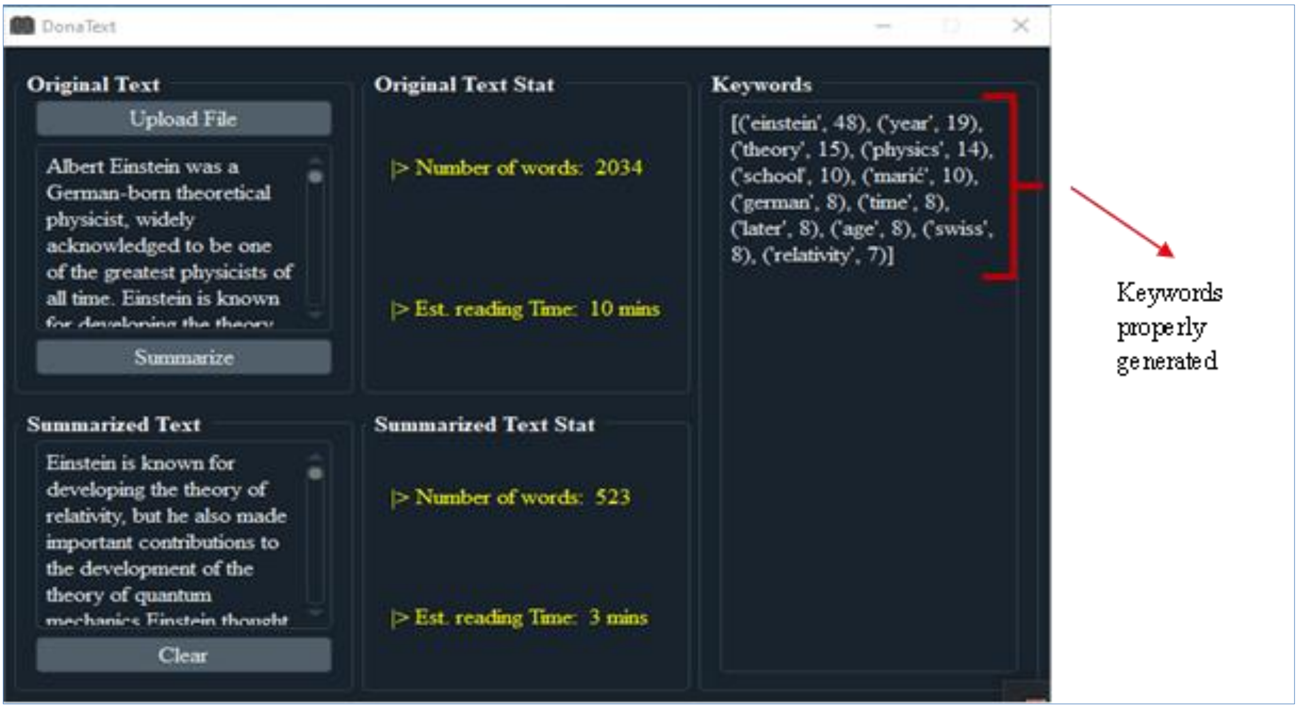
		B	327	6
8	1654	A	377	7
		B	327	8
9	1773	A	396	5
		B	327	4
10	1987	A	421	8
		B	327	6

Key: A = New system; B = Existing system

5. Discussion

5.1. Tokenization

The snapshot of the proposed summarization system in figure 2 shows the effectiveness of the implemented boundary-based tokenization method on the proposed algorithm for this study. In comparison with the existing system, the proposed summarization system is seen to have efficiently handled word tokenization using the boundary-based approach leading to the proper selection of keywords. The selected keywords is in line with the definition of [21] on what a keyword should entail. It further establishes the findings of [6] on the relevance of tokenization during preprocessing stage of automatic text summarization.



Albert Einstein was a German-born theoretical physicist, widely acknowledged to be one of the greatest physicists of all time. Einstein is known for developing the theory

▷ Number of words: 2034

▷ Est. reading Time: 10 mins

[('einstein', 48), ('year', 19), ('theory', 15), ('physics', 14), ('school', 10), ('marić', 10), ('german', 8), ('time', 8), ('later', 8), ('age', 8), ('swiss', 8), ('relativity', 7)]

Einstein is known for developing the theory of relativity, but he also made important contributions to the development of the theory of quantum mechanics Einstein thought

▷ Number of words: 523

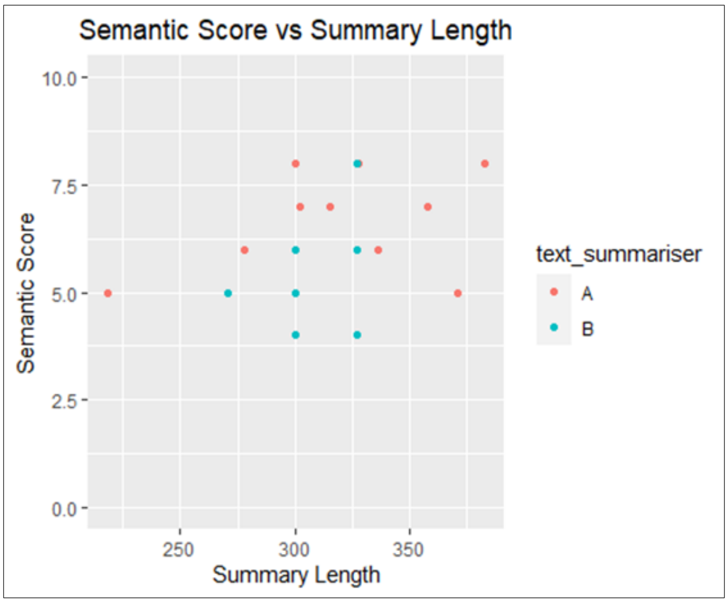
▷ Est. reading Time: 3 mins

Keywords properly generated

Figure 2 Screenshot of the summarization tool

5.2. Dependency between semantic score and summary length

After implementing the boundary-based tokenization method on the proposed summarization algorithm, it is seen from the scatter plot of figure 3 that, based on evaluators' assessment, the new system achieved better semantic scores with respect to summary length, compared to the existing system.

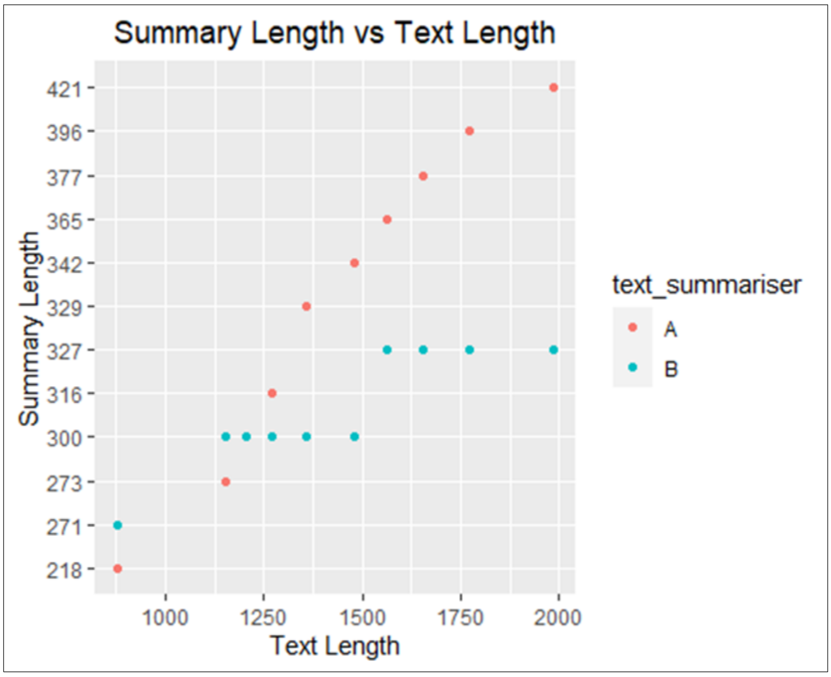


Key: A = New system; B = Existing system

Figure 3 Scatter plot of semantic score vs summary length

5.3. Influence of input text length on summary length

Static points were observed on the scatter plots of figure 4 for the existing system with respect to increasing input text length. The existing system returned the same summary for specific range of input text irrespective of any additional text to its original input content. This is presumably because the existing system's algorithm permits it to perform summarization with respect to predefined / specific text length and therefore ignores further increases in text length until the next predefined text length is reached. Although the summary length of the new system appeared to be slightly higher, as text length increased, it showed the intelligence of the new system to dynamically handle copious increases in text length during summarization. This is in line with the findings of previous researchers [12, 23] on the need for automatic text summarizers in cushioning the problems accompanied with managing huge amounts of textual data.



Key: A = New system; B = Existing system

Figure 4 Scatter plot of summary length vs text length

6. Conclusion

The implementation of boundary-based tokenization in the proposed extractive text summarization algorithm enhanced word tokenization leading to efficient keywords generation. This study therefore, provides a smart extractive text summarization tool which utilizes the proposed approach in generating summary for text documents based on proper word tokenization and quality sentence selection.

Compliance with ethical standards

Acknowledgments

The authors are thankful to the anonymous reviewers of this work for their reviews.

Disclosure of conflict of interest

The authors declare that no known competing interest exists.

References

[1] El-Kassas WS, Salama CR, Rafea AA, Mohamed H K. Automatic text summarization: A comprehensive survey. In Expert Systems with Applications. 2021; 165.

[2] Anjusha P, Mahajan AR. Rule based Text Summarizer for History Documents. International Journal of Innovations in Engineering and Technology. 2016; 7(4): 512-516.

[3] Oyekan AO, Enikuomehin AO, Aribisala BS. A review of documented findings on Text Summarization for Children news rendering. 2017; 8(1).

[4] Gaikwad DK, Mahender CN. A Review Paper on Text Summarization. International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE). 2016; 5(3): 154–160.

[5] Verma R, Lee D. Extractive Summarization: Limits, Compression, Generalized Model and Heuristics. Computacion y Sistemas. 2017; 21(4): 787–798.

[6] Nguyen MT, Nguyen THN, Nguyen HD, Nguyen VH. Learning to Estimate the Importance of Sentences for Multi-Document Summarization. Proceedings of 2018 10th International Conference on Knowledge and Systems Engineering, KSE. November 2018; 31–36.

[7] Ramezani M, Feizi-Derakhshi MR. Automated text summarization: An overview. Applied Artificial Intelligence. 2014; 28(2): 178–215.

[8] Saziyabegum S, Priti S. Literature Review on Extractive Text Summarization Approaches. International Journal of Computer Applications. 2016; 156(12): 28–36.

[9] Moratanch N, Chitrakala S. A survey on extractive text summarization. International Conference on Computer, Communication, and Signal Processing: Special Focus on IoT, ICCCSP. 2017.

[10] Steinberger J, Ježek K. Evaluation measures for text summarization. Computing and Informatics. 2012; 28(2): 251-275.

[11] Lin CY. ROUGE: A Package for Automatic Evaluation of Summaries. Spain, In Proceedings of the Workshop on Text Summarization Branches Out. 2004.

[12] Böhm F, Gao Y, Meyer CM, Shapira O, Dagan I, Gurevych I. Better rewards yield better summaries: Learning to summarise without references. EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference. 2020; 3110–3120.

[13] Erera S, Shmueli-Scheuer M, Feigenblat G, Nakash OP, Boni O, Roitman H, Cohen D, Weiner B, Mass Y, Rivlin O, Lev G, Jerbi A, Herzig J, Hou Y, Jochim C, Gleize M, Bonin F, Ganguly D, Konopnicki D. A summarization system for scientific documents. EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, Proceedings of System Demonstrations. 2020; 211–216.

[14] Chaganty AT, Mussmann S, Liang P. The price of debiasing automatic metrics in natural language evaluation. ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers). 2018; 1: 643–653.

[15] Kanitha DK, Mubarak DMN. An Overview of Extractive Based Automatic Text Summarization Systems. International Journal of Computer Science and Information Technology. 2016; 8(5): 33–44.

[16] Manju K, Peter SD, Idicula SM. A framework for generating extractive summary from multiple malayalam documents. Information (Switzerland). 2021; 12(1): 1–16.

[17] Mihalcea R Tarau P. Textrank: Bringing order into text. In Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, Barcelona, Spain. 2004; 404–411.

[18] Vaghasiya NB. Extractive Summarization and Simplification of Scholarly Literature. May 2020.

[19] Bashir M, Rozaimee A, Malini W, Isa W. Automatic Hausa LanguageText Summarization Based on Feature Extraction using Naïve Bayes Model. 2017; 35(9): 2074–2080.

[20] Selvani Deepthi RY. Extractive Text Summarization Using Modified Weighing and Sentence Symmetric Feature Methods. International Journal of Modern Education and Computer Science. 2015; 7(10): 33–39.

[21] Siddiqi S, Sharan A. Keyword and Keyphrase Extraction Techniques: A Literature Review. International Journal of Computer Applications. 2015; 109(2): 18–23.

[22] Saygin AP, Cicekli I, Akman V. "Turing Test: 50 Years Later" (PDF), Minds and Machines. 2000; 10(4): 463–518.

[23] Moradi M, Ghadiri N. Text Summarization in the Biomedical Domain. 2019; 1–12.