ISSN 2350-1022

International Journal of Recent Research in Mathematics Computer Science and Information Technology
Vol. 11, Issue 2, pp: (74-83), Month: October 2024 – March 2025, Available at: www.paperpublications.org

A Web-Based Document Summarizing and Topic Prediction System Using Natural Language Processing

Taiwo Adigun[1,2], Einstein Ebereonwu[3]

[1]Department of Software Engineering, Babcock University, Ilishan-Remo, Nigeria.

[2]Department of Software Engineering, University of Lay Adventist of Kigali, Rwanda.

[3]Department of Computer Science, Royal Holloway University of London, London, United Kingdom.

DOI: https://doi.org/10.5281/zenodo.14863280

Published Date: 13-February-2025

Abstract: **The need to read through large volumes of information or other bodies of text and produce a summarized version of the documents is usually a daunting task for many. For instance, researchers often times find themselves reading through lines, paragraphs, pages or chapters of documents before finally coming across the key points which will help them with their research work. Most existing systems generate the summary only in text and a specific language. Therefore, this study provides a web-based document summarizing and topic prediction system with Text-To-Speech and auto-language detection features. The system provides two types of two types of summaries; Extractive and abstractive summaries. The abstractive summary feature makes use of the GPT-3 API while the extractive summary and topic prediction make use of sentence scoring. The language detection model is trained with language data found on Kaggle. The text-to-speech feature is implemented using a JavaScript library. The system is able to detect the topic of the body of text and displays it to the user, and identifies the language of the text entered while also providing an appropriate summary. Users are able to play audio summaries and download them as a word document. Experimental results of the system have found that both the extractive and abstractive summary features are able to capture the main points of any given body of text and present summaries that convey information about the entire body of text, leveraging on Artificial Intelligence (AI) with a focus on Natural Language Processing (NLP).**

Keywords: **Artificial Intelligence, Natural Language Processing, Topic Prediction, Language Detection, Text-to-Speech.**

1. BACKGROUND

In this modern world, technology is constantly increasing the efficiency, feasibility and ease of ways to carry out time consuming and brain tasking activities. With the increasing amount of information being generated through technology and the need for quick information processing, extracting the key information embedded in large body of text has become almost unachievable for individuals from various backgrounds such as education, medicine, tourism, law, and many more.

However, because of technological advancements and the existence of Artificial Intelligence (AI), with a focus on Natural Language Processing (NLP), the work of summarizing has become a very seamless task; systems are now very capable of churning large bodies of text while also withholding the semantics. Formerly, individuals had to read through multiple lines of text, absorb and digest the information it contained before proceeding to write a summary of essential points included in the original body of text or take any more actions, but due to the advancements of AI, this is no longer the case.

Artificial intelligence (AI) is the capacity of a machine controlled by a computer to perform jobs that generally necessitate human intelligence and judgement. (B.J. Copeland, 2022). AI is an advanced field which is more than capable of performing

several tasks in areas such as medicine, education, finance, e.t.c. Natural Language Processing (NLP) is a subfield of Artificial Intelligence which came into existence in the middle of 20th century it enables computers to interpret spoken words or written texts in a way comparable to that of humans (IBM Cloud Education, 2022). NLP has been used to achieve several ground-breaking achievements including but not limited to text translation from one language to another, development of chat bots, text summarization as in this case, and personal-assistants such as Apple's Siri, Amazon's Alexa, and Google's Google Assistant, which are capable of having real-time non-human controlled conversations with individuals, carrying out specific tasks such as setting a reminder for 12:30, calling a friend, texting a friend, and much more can all be achieved simply by asking a home assistant which are able to understand humans due to their AI integration. Therefore, this study will apply NLP in order to determine the key content of any body of text which will be used to coin out a proper summary of the text entered and attempt to predict the text topic of the body of text which will help curb the issue of having to consume a lot of unnecessary information before getting the key points required of a body of text.

2. TEXT SUMMARIZING APPROACHES

Text summarizing research can be traced back to the onset of Artificial Intelligence (AI). Such times were referred to as the "early enthusiasm, great expectations" (Stuart Jonathan Russell & Peter Norvig, 2009). The very first attempt at automated text summarization is credited to Luhn (Luhn, 1958). He discovered that statistical data derived from word frequencies can be used to identify the significance of sentences in a text corpus. Another very significant attempt was made by Edmundson (Edmundson, 1969) who highlighted the fact that relying solely on word frequencies for identifying important sentences in texts is insufficient.

Several approaches have been adopted in attempts to summarize texts. Such approaches can be broken down into 3 distinct categories namely; Summarizing by Empirical Methods, Machine Learning Based Approach, and Deep Learning Based Approach.

a. Empirical Methods Based Approach

The empirical method makes use of observation about the properties of the input or by application of linguistic theories. To identify the most appropriate sentences for a summary, researchers used information retrieval methods that calculate links between texts and parts of texts (Salton et al., 1997) or relied on graph-based ranking models (Rada Mihalcea & Paul Tarau, 2004). Analysis of anaphoric and coreferential links in texts (ANDO et al., 2005) or lexical repetition in texts (Barzilay & Elhadad, 1999) was also used to calculate a score for all sentences in texts and extract only those with the highest score. Rhetorical Structure Theory (RST) (MANN & THOMPSON, 1988) a theory that organizes text in primarily non-overlapping spans linked by rhetorical relations, has been successfully applied to the development of heuristics for selecting the most pertinent sentences for a summary (Alonso i Alemany & Fuentes Fort, 2003).

The main disadvantage of these methods is that they rely on researchers' intuitions about how to evaluate the importance of a sentence and employ approximations to implement complex linguistic theories such as RST. As a result, the summaries produced are not always of high quality. These methods were largely preferred during the early stages of the field's re-emergence, but they became less popular after the year 2000. However, successful applications of these approaches can be found later on, such as in (Lloret & Palomar, 2012). Furthermore, these methods are still used to generate features for machine learning-based summarization approaches and methods like TextRank. (Rada Mihalcea & Paul Tarau, 2004) are still used as baselines.

b. Machine Learning Based Approach

The concept of training a classifier capable of identifying which sentences should be included in a summary was first used in (Kupiec et al., 1995), but it was not widely used until annotated corpora became available. The corpora developed in the DUC, as well as the automatic evaluation metrics proposed in the DUC, had the greatest impact on the community because they enabled direct comparison of methods. Prior to the availability of these corpora, researchers had to create their own annotated resources, which were often tailored to the research questions they were attempting to answer (Simone Teufel, 1997). Researchers tried nearly every machine learning algorithm available for years in an attempt to produce better summaries. Bayesian classifiers (Neto et al., 2002) and decision trees (Neto et al., 2002) are among the methods tested, as are hidden Markov models (Conroy et al, 2001) and integer linear programming (Luo et al., 2018). (Knight & Marcu, 2002) modify the noisy channel used in Statistical Machine Translation to create a method for sentence compression, which is

seen as a first step toward automatically producing summaries. In (NASERASADI et al., 2019), the summarization process was also viewed as an optimization problem in which weights are learned from data.

c. Deep Learning Based Approach

Traditional machine learning approaches are still used, but they are gradually being replaced by deep learning methods. The introduction of neural approaches for text summarization, which occurred around 2015, marked the beginning of the third approach. This reflects changes in other fields of computational linguistics where the use of deep learning technologies has resulted in new and more accurate methods. There are numerous papers available now in which researchers attempt to improve on the state of the art by applying the most recent neural models for automatic text summarization. In some cases, the use of neural architectures is not entirely justified because the improvements are minor; however, when used correctly, the new methods allow researchers to produce better results.

Deep learning techniques are also used for extractive summarization. (Kobayashi et al., 2015) and (Yogatama et al., 2015) present two approaches that use the semantic information provided by word embeddings to propose unsupervised optimization algorithms for finding the best set of sentences given the search space created by the semantic representation of the sentences in the document.

3. RELATED WORKS

Anjali et al. (2019) developed a graph-based approach for keyword extraction from documents. This study describes an unsupervised method for extracting keywords from a document. The graph-based method combines the RAKE algorithm and Keyword Extraction using Collective Node Weight (KECNW), in which candidate keywords are extracted using the RAKE algorithm and keywords are selected using the KECNW model. The final result of this method is a list of keywords extracted from a document, along with their rank. However, keywords extraction is based on knowledge of stop words. Document summarization using textrank and semantic network was developed by Ashari & Riasetiawan (2017). The research used text rank algorithms, Semantic Networks, and Corpus Statistics to implement a document summarizing system. To measure the quality of the system output, the recall, precision, and F-Score of the summary were calculated using ROUGE-N methods. The style of writing, the selection of words and symbols in the document, and the length all have an impact on the quality of the summaries produced.

Gong et al. (2022) use sentence centrality to improving extractive document summarization. The sentence centrality is based on directed graphs, and it reflects both the sentence-document relationship and the sentence position information in the document. By using sentence centrality to improve sentence representation, relevance of sentences and documents was implicitly been increased. Experiments on the CNN/Daily Mail dataset revealed that EDS models with sentence centrality outperformed baseline models significantly. But it required heavy processing power. Enhancing biomedical text summarization using semantic relation extraction is a study conducted by Shang et al. (2011). Based on semantic relation extraction, this paper presents a method for generating text summaries for a given biomedical concept, such as H1N1 disease, from multiple biomedical literature. However, semantic relationship extraction efficiency depends solely on SemRep tools capacity.

Artificial intelligence for automatic text summarization study was developed by Day and Chen, (2018). An AI text summarization system architecture was created using three models: a statistical model, a machine learning model, and a deep learning model, and their performance were evaluated. Essay titles and abstracts are used to train an artificial intelligence deep learning model to generate candidate titles, which are then evaluated by ROUGE for performance. But it requires huge amounts of data. Nikolov and Hahnloser, (2019) developed an abstractive document summarization without parallel data. It is an abstractive summarization system that only uses large collections of example summaries and articles that do not match. The method includes an unsupervised sentence extractor that selects salient sentences to include in the final summary, as well as a sentence abstractor trained on pseudo-parallel and synthetic data that paraphrases each extracted sentence. A hybrid machine learning model for multi-document summarization was also proposed by Fattah, (2014). Three models are combined to create a hybrid model that ranks the results in order of importance. It is a method for improving content selection in multi-document automatic text summarization using statistical tools. The method employs a trainable summarizer, which considers several factors, including word similarity among sentences, word similarity among paragraphs, text format, cue-words, a score related to the frequency of terms in the entire document, the title, sentence

location, and the occurrence of non-essential information. Each of these sentence features' impact on the summarization task is investigated. These characteristics are then combined to build text summarizer models based on a maximum entropy model, a naive-Bayes classifier, and a support vector machine. Requires huge computational resources.

4. PROPOSED METHOD

For the purpose of developing this Web-Based Document Summarizing and Topic Prediction System Using Natural Language Processing, the model framework is described in Figure 1 below. The system provides an easy to use, interactive web-based document summarizing and topic prediction system. The system also provides a few additional features such as language detection, summary download and text-to-speech in order to help blind users or speed the summary consumption process for users who can hear. The users will be able to type text into the text area, upload text (.txt) or word (.docx) documents only for content summarizing, the system has no limit to how much a user can summarize for the supported languages.

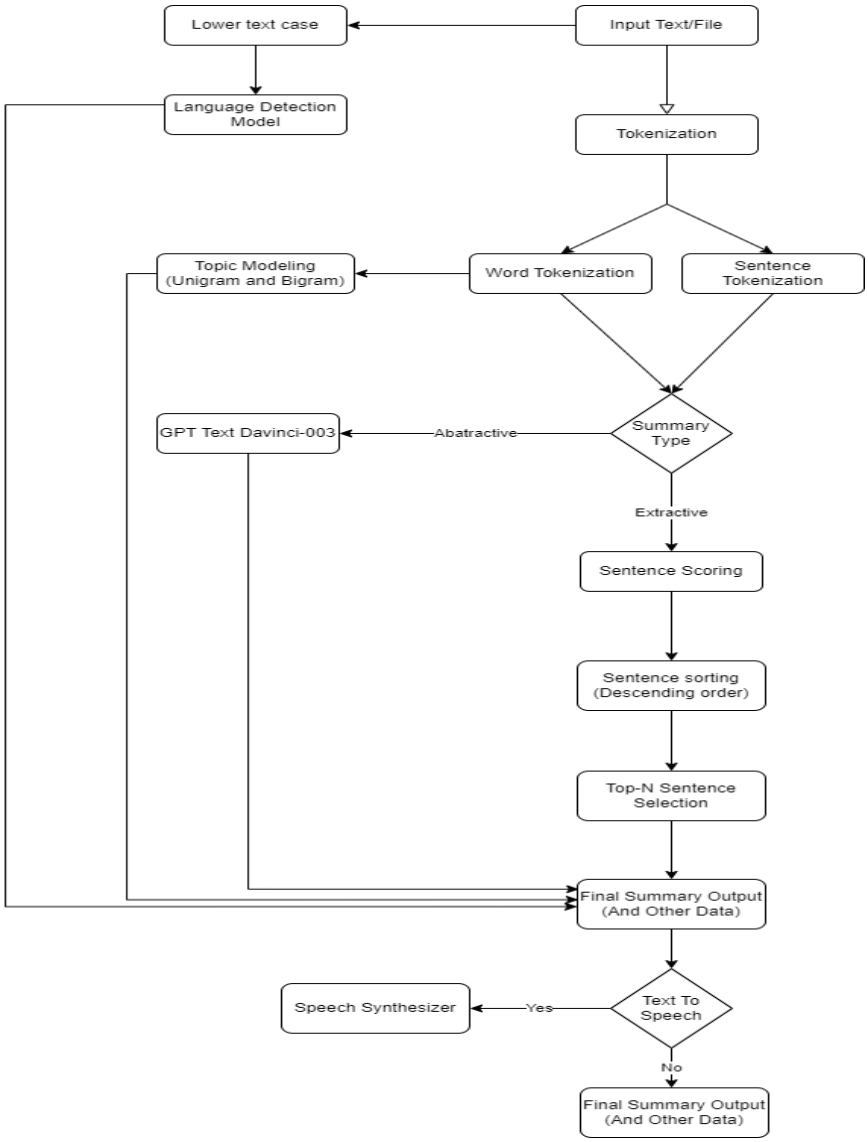

Figure 1: Text Summarizing and Topic Prediction Framework

4.1 Abstractive Summary Feature

The Abstractive summaries are generated using the GPT-3 Text-Davinci-003 API. OpenAI released the GPT-3 Text-DaVinci-003 model, which is an updated and improved version of the Text-DaVinci-002 model. This deep learning model is part of the GPT family was used to generate summary from text by paraphrasing the text into new sentences.

ISSN 2350-1022

International Journal of Recent Research in Mathematics Computer Science and Information Technology
Vol. 11, Issue 2, pp: (74-83), Month: October 2024 – March 2025, Available at: www.paperpublications.org

4.2 Extractive Summary Feature

The extractive text summary feature is achieved by scoring words excluding all stop words using the term frequency-inverse document frequency (tf-idf), those scored words are then used to score each sentence in order to determine importance of every sentence which is then returned in descending order of importance following the order in which they appear in the original text.

4.3 Implementation of Topic Prediction Feature

This feature is similar to the word scoring in the extractive summary feature however, it isn't just single words that are scored, bigrams are scored as well. Bigrams are words in pairs of two, when these two are scored, the bigram and single non-stop word that return with the highest scores are returned as the predicted topics.

4.4 Implementation of Language Detection Feature

A language model is responsible for this language detection feature. The language detection model was trained with the decision tree classifier using data from Kaggle. The dataset was downloaded from https://www.kaggle.com/datasets/emirhanai/language-detect-artificial-intelligence-software?select=language_detection.csv. It is made up of 21859 rows of text from 22 different languages. It has two columns; the text string and the language. The classification model is incorporated into the system to detect the language of the text.

4.5 Implementation of Text-To-Speech Feature

The text to speech feature was implemented using JavaScript "responsive text" API. Once text is fed into it, it analyzes it and produces the speech version of the text.

5. RESULTS AND DISCUSSION

This system can be useful for students, lecturers, accountants, businessmen & women, politicians, researchers, and many more individuals of various backgrounds who often have to read through large volumes of information or produce a summarized version of documents or other bodies of text. is a system that provides topic modeling, language detection, and summarization features. It offers two types of summaries, extractive and abstractive, both of which are capable of capturing the main points of any given body of text. It has been tested by users from various backgrounds and domains and has proven to be useful.

The following are snapshots of various parts of the webapp including the landing page, text summary page, file upload page, etc.

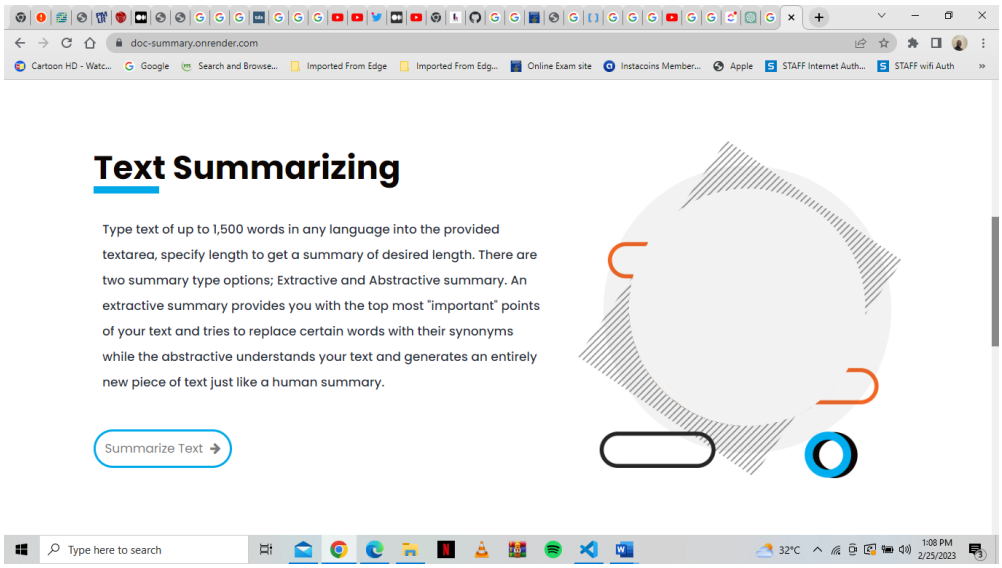

Figure 2: Landing page; Info on text summary

ISSN 2350-1022

International Journal of Recent Research in Mathematics Computer Science and Information Technology
Vol. 11, Issue 2, pp: (74-83), Month: October 2024 – March 2025, Available at: www.paperpublications.org

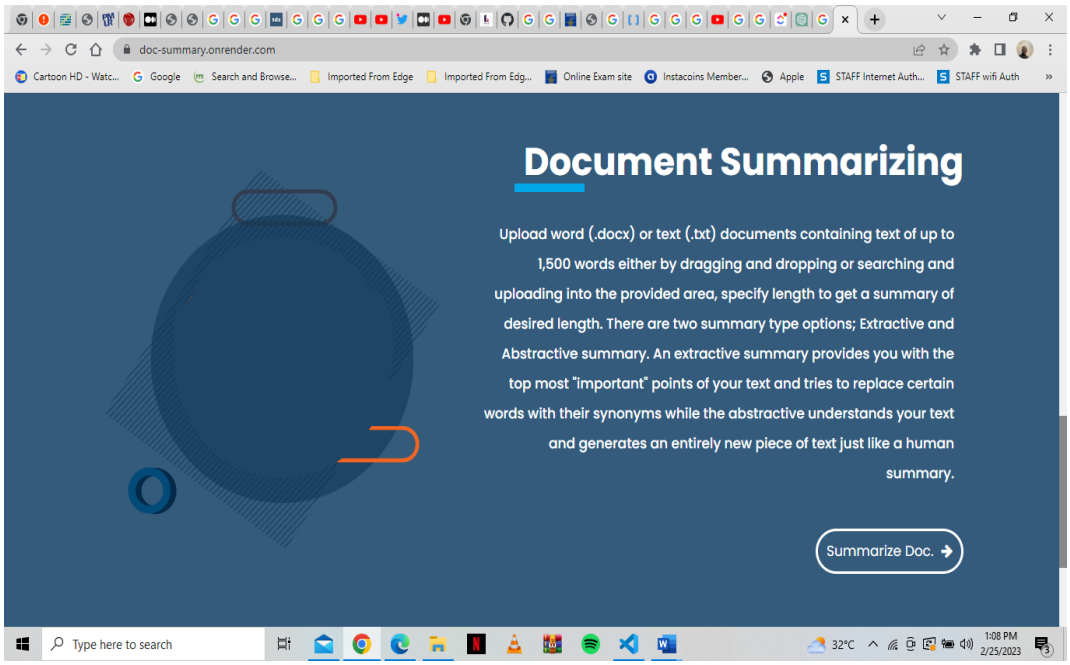

Figure 3: Landing page; Info on document summary.

As seen in *fig. 2,* a user can summarize text in any language however, such text cannot be longer than 1,500 words long per time or else only the first 1,500 words will be considered for summarizing purpose. In *fig. 3,* you can see that it is only possible to summarize text (.txt) and words (.doc, .docx) documents. Only the first 1,500 words in any documents with above 1,500 words will be considered for summary.

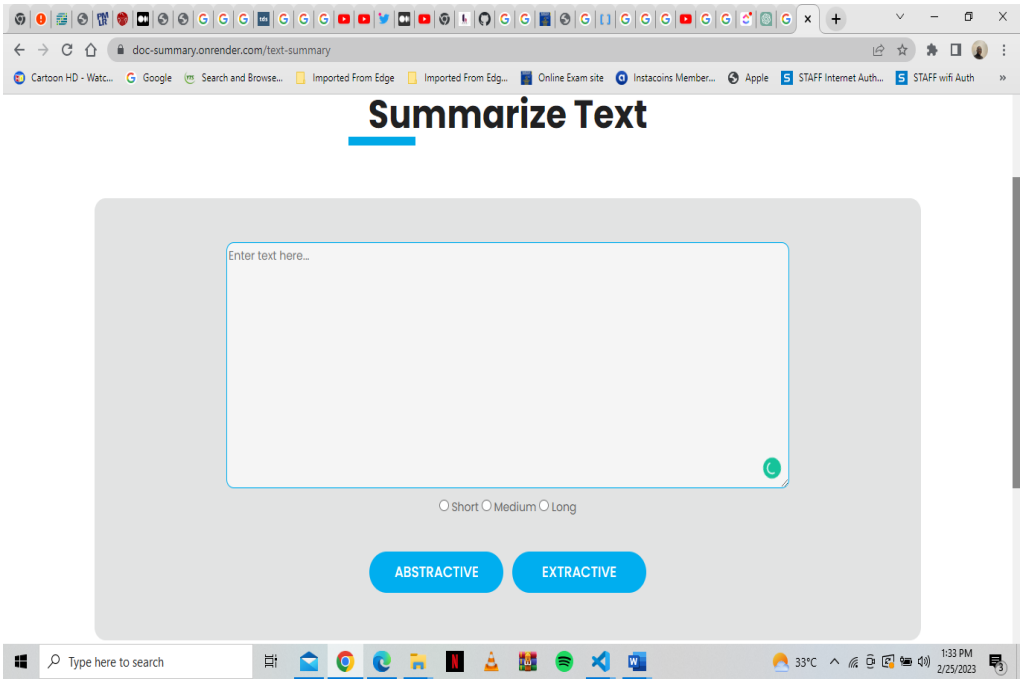

Figure 4: Text summary page

Figure 4 above shows the text summarizing page with the text area for text input, the summary selector bar and the summary types; Abstractive – for human like summaries and Extractive – for a condensed version of the original text which picks out key sentences and returns them back to user.

ISSN 2350-1022

International Journal of Recent Research in Mathematics Computer Science and Information Technology
Vol. 11, Issue 2, pp: (74-83), Month: October 2024 – March 2025, Available at: www.paperpublications.org

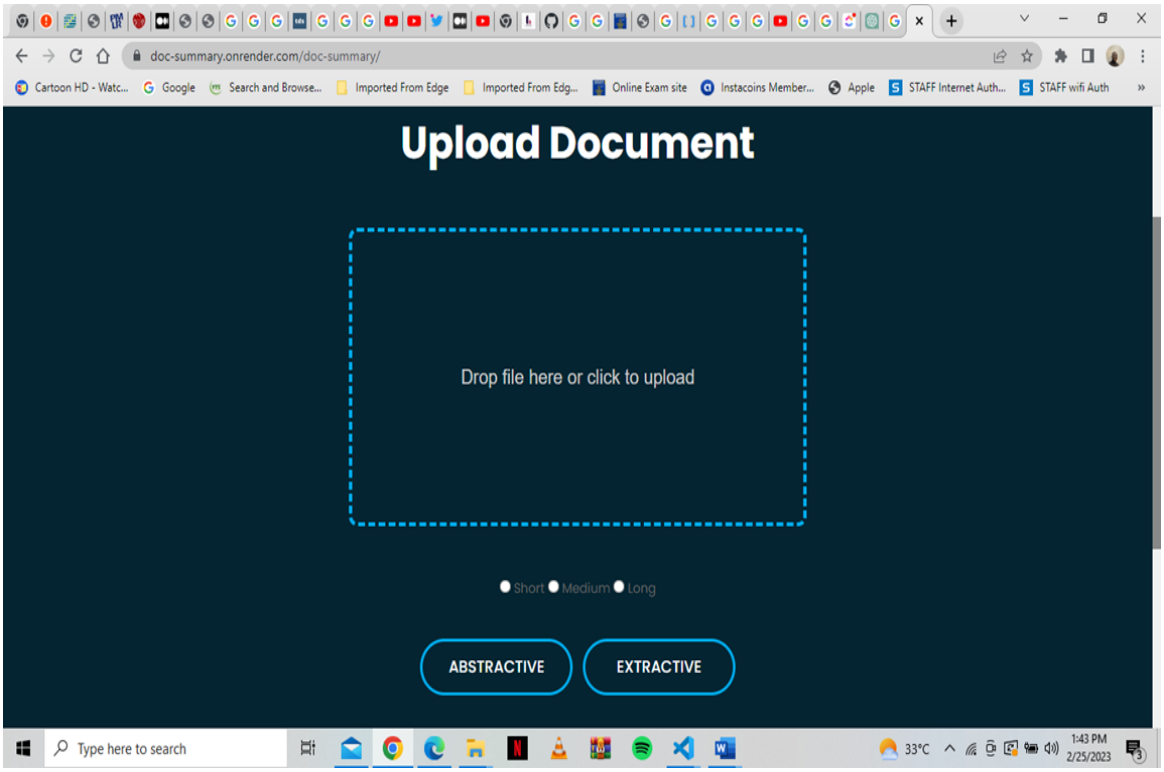

Figure 5: Document Summarizing page

Figure 5 above shows the document summary page where documents can be uploaded by dragging and dropping or by browsing through directories after clicking within the broken lines box. Users can only upload word (.doc, .docx) or text (.txt) documents.

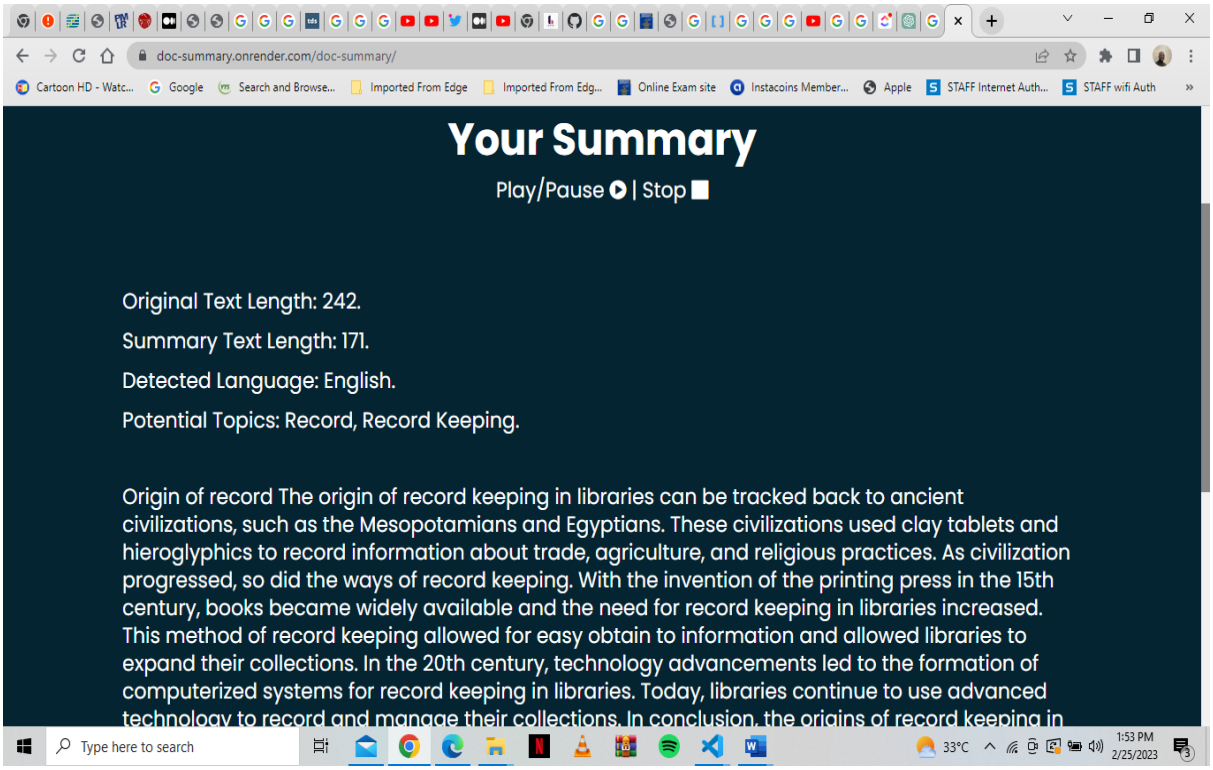

Figure 6: Summary result page.

The summary result page provides the user with key information about their text such as original text length, the summary length, the language of the text as detected by the AI model, potential topics for the body of text and finally the summary. Users can play their summary and can also download their summary.

Figure 7 below shows the QR code designed for gaining access into the system.

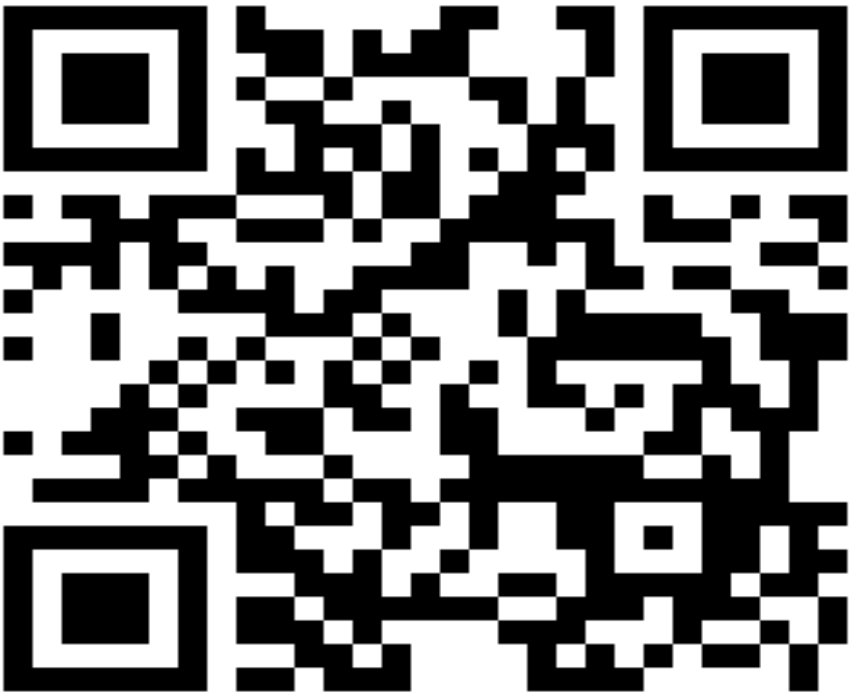

Figure 7: QR Code to Access System.

6. CONCLUSION

A text summarizing software is a tool that automatically condenses a large piece of text into a shorter, more concise version while retaining the main ideas and important details of the original text. There are two main types of text summarization which this system is capable of: extractive and abstractive. The system was designed to address issues such as; Information overload, Language barriers (for individuals who are not proficient in a particular language, a text summarizing software can provide a quick and efficient way to understand the main ideas). It highlights the basic functionalities of any good text summarizing system, with the added advantage of language detection and summary text-to-speech for tired or disabled users. This tool is for individuals and organizations looking to efficiently process and understand large amounts of information. With the ability to quickly and accurately summarize lengthy documents or articles, this software can save time, reduce information overload, and help individuals make more informed decisions.

REFERENCES

[1] Abdi, A., Idris, N., Alguliyev, R. M., & Aliguliyev, R. M. (2016). An Automated Summarization Assessment Algorithm for Identifying Summarizing Strategies. *PLOS ONE*, *11*(1), e0145809. https://doi.org/10.1371/journal.pone.0145809

[2] Alonso i Alemany, L., & Fuentes Fort, M. (2003). Integrating cohesion and coherence for automatic summarization. *Proceedings of the Tenth Conference on European Chapter of the Association for Computational Linguistics - EACL '03*, 1. https://doi.org/10.3115/1067737.1067739

[3] ANDO, R., BOGURAEV, B., BYRD, R., & NEFF, M. (2005). Visualization-enabled multi-document summarization by Iterative Residual Rescaling. *Natural Language Engineering*, *11*(1), 67–86. https://doi.org/10.1017/S1351324904003389

[4] Anjali, S., Meera, N. M., & Thushara, M. G. (2019). A Graph based Approach for Keyword Extraction from Documents. *2019 Second International Conference on Advanced Computational and Communication Paradigms (ICACCP)*, 1–4. https://doi.org/10.1109/ICACCP.2019.8882946

[5] Ashari, A., & Riasetiawan, M. (2017). Document Summarization using TextRank and Semantic Network. *International Journal of Intelligent Systems and Applications*, *9*(11), 26–33. https://doi.org/10.5815/ijisa.2017.11.04

[6] Barzilay, R., & Elhadad, M. (1999). Using Lexical Chains for Text Summarization. *The MIT Press, Cambridge, 111-121*.

[7] B.J. Copeland. (2022). Artificial intelligence. *Encyclopedia Britannica*.

[8] Chatterjee, N., & Mohan, S. (2007). Extraction-Based Single-Document Summarization Using Random Indexing. *19th IEEE International Conference on Tools with Artificial Intelligence(ICTAI 2007)*, 448–455. https://doi.org/10.1109/ICTAI.2007.28

[9] Day, M.-Y., & Chen, C.-Y. (2018). Artificial Intelligence for Automatic Text Summarization. *2018 IEEE International Conference on Information Reuse and Integration (IRI)*, 478–484. https://doi.org/10.1109/IRI.2018.00076

[10] Edmundson, H. P. (1969). New Methods in Automatic Extracting. *Journal of the ACM*, *16*(2), 264–285. https://doi.org/10.1145/321510.321519

[11] Fattah, M. A. (2014). A hybrid machine learning model for multi-document summarization. *Applied Intelligence*, *40*(4), 592–600. https://doi.org/10.1007/s10489-013-0490-0

[12] Gong, S., Zhu, Z., Qi, J., Tong, C., Lu, Q., & Wu, W. (2022). Improving extractive document summarization with sentence centrality. *PLOS ONE*, *17*(7), e0268278. https://doi.org/10.1371/journal.pone.0268278

[13] He, Z., Chen, C., Bu, J., Wang, C., Zhang, L., Cai, D., & He, X. (n.d.). *Document Summarization Based on Data Reconstruction*. www.aaai.org

[14] IBM Cloud Education. (2020). *Natural Language Processing (NLP)*.

[15] *IBM Cloud Education*. (2022).

[16] Jordana A. (2023). *What Is JavaScript? A Basic Introduction to JS for Beginners*.

[17] Jurafsky, D., & Martin, J. H. (2019). *Speech and Language Processing (3rd ed.). Pearson*.

[18] Kevin Humphreys, Robert Gaizauskas, & Saliha Azzam. (1997). *Event Coreference for Information Extraction*.

[19] Knight, K., & Marcu, D. (2002). Summarization beyond sentence extraction: A probabilistic approach to sentence compression. *Artificial Intelligence*, *139*(1), 91–107. https://doi.org/10.1016/S0004-3702(02)00222-9

[20] Kobayashi, H., Noguchi, M., & Yatsuka, T. (2015). *Summarization based on embedding distributions*.

[21] Kristian. (2022). *What is the GPT-3 text-davinci-003 model?*

[22] Kupiec, J., Pedersen, J., & Chen, F. (1995). A trainable document summarizer. *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval - SIGIR '95*, 68–73. https://doi.org/10.1145/215206.215333

[23] Lipton, Z. C., & Steinhardt, J. (2018). *Troubling Trends in Machine Learning Scholarship*.

[24] Lloret, E., & Palomar, M. (2012). Text summarisation in progress: a literature review. *Artificial Intelligence Review*, *37*(1), 1–41. https://doi.org/10.1007/s10462-011-9216-z

[25] Luhn, H. P. (1958). The Automatic Creation of Literature Abstracts. *IBM Journal of Research and Development*, *2*(2), 159–165. https://doi.org/10.1147/rd.22.0159

[26] Luo, W., Liu, F., Liu, Z., & Litman, D. (2018). *A Novel ILP Framework for Summarizing Content with High Lexical Variety*.

[27] MANN, W. C., & THOMPSON, S. A. (1988). Rhetorical Structure Theory: Toward a functional theory of text organization. *Text - Interdisciplinary Journal for the Study of Discourse*, *8*(3). https://doi.org/10.1515/text.1.1988.8.3.243

[28] MATTHEW DEERY. (2022). *What Is Flask and How Do Developers Use It? A Quick Guide*.

[29] NASERASADI, A., KHOSRAVI, H., & SADEGHI, F. (2019). Extractive multi-document summarization based on textual entailment and sentence compression via knapsack problem. *Natural Language Engineering*, *25*(1), 121–146. https://doi.org/10.1017/S1351324918000414

[30] Neto, J. L., Freitas, A. A., & Kaestner, C. A. A. (2002). *Automatic Text Summarization Using a Machine Learning Approach* (pp. 205–215). https://doi.org/10.1007/3-540-36127-8_20

[31] Nikolov, N. I., & Hahnloser, R. H. R. (2019). *Abstractive Document Summarization without Parallel Data*.

[32] Oxford Learner's Dictionaries. (n.d.). *Summary Definition*.

[33] Prasasthy K B. (2021). *Brief history of Text Summarization*.

[34] Rada Mihalcea, & Paul Tarau. (2004). TextRank: Bringing Order into Text. *Association for Computational Linguistics*, *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*.

[35] REITER, E., & DALE, R. (1997). Building applied natural language generation systems. *Natural Language Engineering*, *3*(1), S1351324997001502. https://doi.org/10.1017/S1351324997001502

[36] Salton, G., Singhal, A., Mitra, M., & Buckley, C. (1997). Automatic text structuring and summarization. *Information Processing & Management*, *33*(2), 193–207. https://doi.org/10.1016/S0306-4573(96)00062-3

[37] Sarah Lewis. (2019). Prototyping Model. *SearchCIO*.

[38] Shang, Y., Li, Y., Lin, H., & Yang, Z. (2011). Enhancing Biomedical Text Summarization Using Semantic Relation Extraction. *PLoS ONE*, *6*(8), e23862. https://doi.org/10.1371/journal.pone.0023862

[39] Simone Teufel. (1997). *Sentence extraction as a classification task*.

[40] SRIHARI, R. K., LI, W., CORNELL, T., & NIU, C. (2008). InfoXtract: A customizable intermediate level information extraction engine. *Natural Language Engineering*, *14*(01). https://doi.org/10.1017/S1351324906004116

[41] Stuart Jonathan Russell, & Peter Norvig. (2009). *Artificial Intelligence: A Modern Approach*.

[42] T. Berners-Lee, & D. Connolly. (1995). *Hypertext Markup Language - 2.0*.

[43] The Britannica Dictionary. (n.d.). *Summary Definition*.

[44] Xu, S., Jiang, H., & Lau, F. C. M. (2009). User-oriented document summarization through vision-based eye-tracking. *Proceedings of the 14th International Conference on Intelligent User Interfaces*, 7–16. https://doi.org/10.1145/1502650.1502656

[45] Yogatama, D., Liu, F., & Smith, N. A. (2015). Extractive Summarization by Maximizing Semantic Volume. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 1961–1966. https://doi.org/10.18653/v1/D15-1228