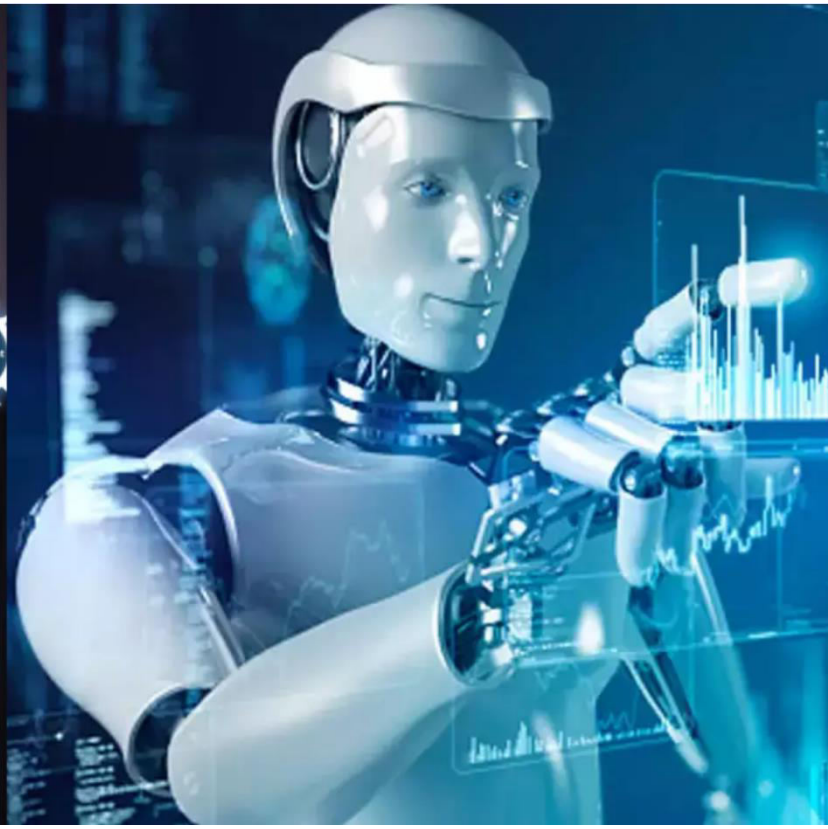
IJIRSET
International Journal of Innovative Research in Science | Engineering | Technology

ISSN(O): 2319-8753
ISSN(P): 2347-6710

International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.699

Volume 14, Issue 4, April 2025

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404576|

Optimized Semantic Abstraction for High-Precision Text Summarization in NLP Systems

P.Babu Reddy, Y.Veeranaganjineya Reddy, N.Yogesh Naidu, Prof. Ranjitha U N

School of CSE, REVA University, Bengaluru, India

ABSTRACT: The project offers a text summarization tool that can handle content from PDFs, DOCX, and plain text. It incorporates extractive and abstractive summarization methods for creating abridged summaries. The extractive method applies sentence tokenization in the selection of important sentences, whereas the abstractive method utilizes a pre-trained Hugging Face model. The tool also has a Streamlit-based user interface for handling multiple file uploads and direct input of text. Strong error handling and text cleansing improve dependability and summarization accuracy, so the tool is useful for applications needing effective text analysis like in healthcare, research, and legal arenas.

KEYWORDS: Text Summarization, Extractive Summarization, Abstractive Summarization, Natural Language Processing (NLP), HuggingFace Models, Streamlit, PDF Text Extraction, DOCX Text Processing, Automation in Text Analysis.

I. INTRODUCTION

The exponential growth of textual data in various domains, such as healthcare, legal, and research fields, has necessitated the development of efficient tools for summarization. Summarization enables users to extract key insights from extensive content quickly, saving time and improving decision-making processes. This paper introduces a text summarization tool designed to handle diverse input formats, including plain text, PDFs, and DOCX files. The tool combines extractive and abstractive summarization techniques.

Text summarization is essential in NLP, converting extensive text into concise, meaningful content. This study traces the evolution of summarization techniques from syntactic to advanced semantic models, focusing on extractive and abstractive methods[8].

In today's information-rich world, efficient text summarization is crucial for quick comprehension. This study develops an NLP-based summarizer using deep learning, particularly transformer models like BERT, the system effectively extracts key information, enhancing the processing of large textual data[6].

The application features a user-friendly interface built with Streamlit, enabling seamless interaction for uploading documents or entering plain text. Additionally, text preprocessing steps, including cleaning and tokenization, ensure enhanced summarization accuracy. Information-driven world, the sheer volume of textual data available in various formats poses a significant challenge for professionals and organizations. With the rise of textual data, efficient processing is essential for better comprehension. This study examines text summarization in NLP, focusing on extractive methods for efficiency[18]. Whether in healthcare, research, education, or legal domains, the need to process and comprehend large volumes of data efficiently has become increasingly important. Traditional methods of manually reviewing such information are not only time-consuming but also prone to errors and inconsistencies. Text summarization in NLP condenses information while preserving key content. Extractive methods select important sentences, while abstractive approaches generate new phrasing[13]. Automated summarization tools provide a viable solution by offering concise and accurate summaries, enabling users to focus on critical insights.

As digital data grows, automatic text summarization is crucial for extracting key information efficiently. This effective summarization system, offering a time-saving solution for processing large texts across various domains[11]. Extractive summarization relies on selecting key sentences from the input text, making it a computationally efficient method that retains the original wording of the content. In contrast, abstractive summarization generates summaries using advanced machine learning models, allowing for the rephrasing and condensation of content into a more natural, human-like format. Text summarization in NLP condenses large texts while retaining key information. It includes extractive

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404576|

methods, which select key sentences, and abstractive approaches, which generate paraphrased summaries[9]. By integrating these two approaches, the proposed tool offers flexibility and effectiveness in handling diverse summarization requirements

The abstractive summarization component leverages the facebook/bart-large-cnn model from HuggingFace, a transformer-based model renowned for its capability to generate contextually accurate and coherent summaries. Meanwhile, extractive summarization is implemented using the NLTK library, which tokenizes the input text into sentences and selects the most relevant ones based on their position and content. The combination of these techniques ensures that the tool can provide meaningful insights from any textual input.

II. LITERATURE SURVEY

Text summarization has become an important area of research in Natural Language Processing (NLP) that facilitates extracting useful information from vast amounts of text data. The ability to generate concise but informative summaries has been used for news aggregation, legal document processing, medical report summarization, and content filtering. Summarization techniques have shifted from rule-based and statistical methods to deep learning models incorporating semantic representations and context-sensitive learning mechanisms over time.

Earlier summarization methods were predominantly extractive, consisting of the mere extraction of meaningful sentences from the original text with no modification of sentence structure. TF-IDF and graph models such as TextRank were often employed to score sentences based on importance (Kolambkar et al., 2024). Although such approaches preserved key information, they lacked rich contextual understanding and often produced broken or redundant summaries that could not capture the document's meaning. In addition, extractive summarization techniques fared poorly in handling paraphrased text and semantic relationships, limiting them in producing human-like summaries.

The emergence of deep learning saw the paradigm change about text summarization in such a way that models became competent enough to create rather than extracting textual information. The Transformer-based models such as Bidirectional Encoder Representations from Transformers (BERT), Bidirectional and Auto-Regressive Transformers (BART), and Text-to-Text Transfer Transformer (T5) improved significantly the abstractive quality summarization via the use of self-attention methods and contextual embedding (Maurya et al., 2024; Wibawa & Kurniawan, 2024). It contrasts with extractive models as other models learn the long-range context dependencies and semantics, thus permitting it to create natural language-like and coherent outputs.

Of those models, the BART framework has become at the top level of a text summarization algorithm due to having been pre-trained as a denoising autoencoder that has the ability to reconstruct the text and refine sentences effectively (Adhik et al., 2024). The application of encoder-decoder attention mechanisms by BART further improves it for generating fluent, semantically valid summaries. In addition, such models as PEGASUS and BERTSUM have provided improved performance based on masked language modeling techniques employed to increase summary coherence (Rawat, 2024; Jayatilleke et al., 2024).

Abstractive summarization has moved a long distance, though it still experiences factual accuracy and domain transfer challenges. Most transformer models generate hallucinated text, resulting in mistakes that compromise the validity of summaries. Additionally, the use of these models on specific domain applications remains a challenge, particularly for domains that require technical precision, such as legal and medical summarization. In order to tackle such problems, scholars have explored hybrid summarization techniques that conjoin extractive and abstractive approaches so that the summaries are both factually consistent and linguistically fluent (Zade et al., 2024; Rao et al., 2024).

Along with improving model structures, research has been focusing on optimization techniques to make summarization models efficient and accurate. Hybridization of metaheuristic optimization algorithms has produced promising outcomes in fine-tuning BART-based models for improving their fittingness in diverse textual environments (Luat et al., 2022). Furthermore, incorporating local and global context mechanisms has proved effective for fine-tuning sentiment-aware summarization, particularly for summarizing social media and opinion-based content (Suresh & Banu, 2024).

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404576|

Increased advancements in document preprocessing and normalization techniques have also contributed to improving summarization precision. Techniques such as noise reduction from text, elimination of stopwords, and domain adaptation have been employed to enhance the quality of inputs so that summarization models receive cleaner, better-formatted textual data (Khekare et al., 2024). Such advancements improve the comprehension of document structures, which consequently leads to more precise and relevant summaries.

Despite such advancements, the quality of automatic summaries remains a complex job to assess. The ROUGE and BLEU scores remain the most widely used metrics for summarization model benchmarking, yet they primarily aim towards lexical overlap rather than semantic coherence (Jugran et al., 2021). Because of this limitation, researchers have sought to use other evaluation strategies, such as BERTScore and Semantic Similarity Measures, that verify contextual and factual congruence between reference texts and automatic summaries.

Addressing these limitations, this work proposes an improved semantic abstraction model to promote high-precision text summarization in NLP models. Through the use of recent transformer architectures, preprocessing techniques, hybrid summarization methods, and optimization techniques, this approach aims to improve summary coherence, factual accuracy, and computational performance. Further, domain-adaptive training methodologies will be explored to enable summarization models to be robust in a large variety of domains and applications.

III. METHODOLOGY

Text summarization is a critical domain in Natural Language Processing (NLP) because it condenses the volume of information in text and preserves its meaning and coherence. The system proposed integrates abstractive and extractive summarization methods to enhance accuracy and readability of generated summaries. It applies transformer-based models, language preprocessing techniques, and document processing algorithms to provide automated summarization from different sources like PDFs, DOCX files, and plain text.

This paper presents a text summarization system that is aimed at processing and summarizing text data from various sources in the form of plain text, PDF, and DOCX files. The system is designed with accuracy, flexibility, and simplicity in mind. Both extractive and abstractive methods of summarizing the text are combined with strong preprocessing and file-management capabilities.

The procedure starts with text preprocessing, in which the input text is cleaned to remove special characters, excess whitespace, and unsupported symbols. Preprocessing makes the input data uniform and organized so that effective summarizing is done. Text can be provided directly as plain text or read from imported PDF and DOCX files. The document is read with PyPDF2 when the input is in the PDF format and python-docx when the input is in DOCX format. This ensures compatibility with widely used document formats, and therefore the tool can be used in many fields.

For extractive summarization, the system utilizes the Natural Language Toolkit (NLTK) to tokenize the input text into sentences. Extractive summarization chooses the most significant sentences, typically depending on their position and contextual relevance. In this example, the first three sentences are used to construct the summary, which is a straightforward but effective way of summarizing the input text.
Modules identified

These modules are combined to create an effective, precise, and multi-functional summarization tool, catering to various user needs and accepting diverse input formats with ease.

In-Depth Interaction Pathways within the Platform:
The platform operates based on a series of well-crafted interaction pathways, enabling smooth transition from input to output and maintaining accuracy and usability. First, users may interact with the platform either through file upload or entering plain text. The platform provides PDF and DOCX support for file uploading and supports the processing of multiple documents simultaneously. Alternatively, the plain text might be entered directly into the provided text field. Once the input is received, the system automatically decides the type of input received and directs it to the appropriate processing module.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404576|

The preprocessing is vital because it facilitates cleansing and normalizing of input data. At this point, special characters, whitespace, and unsupported symbols are removed to have a clean and consistent text form. The preprocessing ensures the input is prepared well for the process of summarization. For file uploading, text retrieval is done utilizing libraries specifically utilized for this functionality: PyPDF2 for PDF and python-docx for DOCX. Text retrieved is laid out in structured form, as ready to summarize. Anywhere errors are encountered, such as in the readibility of documents or unsupported types, the system instantly notifies the user with an explanatory error message.
The summarization process is at the heart of the platform and has two distinct approaches: abstractive and extractive summarization. Extractive summarization employs the Natural Language Toolkit (NLTK) to tokenize the text in terms of sentences and selects the most important ones to establish a summary. This approach focuses on retaining important portions of the original content. Conversely, abstractive summarization employs the facebook/bart-large-cnn transformer model to generate a human-readable and coherent summary by paraphrasing and merging the content. Both summarization methods are standalone and provide users with different insights into the summarized content.

After processing, the results are displayed through a user-friendly interface built using Streamlit. The users can view the original extracted text, the abstractive summary, and the extractive summary in parallel. Such transparency allows easy comparison and understanding of the results. Additionally, the interface provides robust error-handling functionality to ensure that any issue encountered while processing is clearly shown to the user, accompanied by guidance on how to perform corrective actions.

The structured interaction flows of the platform provide reliability and performance at every level, from being able to take in multiple input types to being able to yield good-quality summarization outputs. Generally, the entire design makes the platform an all-around viable and practical vehicle for serving customers from all realms of need for text summarization.

The suggested strategy successfully combines linguistic and deep learning-based summarization methods for producing high-quality summaries on all types of documents. The hybrid approach enhances text abstraction, information capture, and summary coherence, and hence it is appropriate for automated report analysis, medical document summarization, and general NLP tasks.

Text summarization is an important part of Natural Language Processing (NLP) that compacts text information without compromising its coherence and meaning. The system proposed integrates abstractive and extractive summarization methods for enhanced accuracy and readability of summaries generated. It utilizes transformer models, linguistic preprocessing techniques, and document processing procedures to enable automatic summarization of text from a variety of sources like PDF, DOCX documents, and plain text.

This paper introduces a text summarizer that is designed to process and summarize text content from diverse sources like plain text, PDF, and DOCX files. The method has been developed for accuracy, usability, and diversity. The model integrates both the extractive and abstractive summarization modes with the robust preprocessing and file-handling infrastructure.

It begins with text preprocessing, where the input text is cleaned to remove special characters, unwanted whitespace, and unsupported symbols. This puts the input data in a normalized and structured form, which can be summarized efficiently. Text may be entered directly as plain text or read from uploaded PDF and DOCX files. The text is extracted using PyPDF2 for PDF and python-docx for DOCX. This offers compatibility with widely used document formats, which enables the tool to gain utility across various sectors.

For extractive summarization, the system employs the Natural Language Toolkit (NLTK) to segment the input text into sentences. Extractive summarization selects the most significant sentences, typically based on position and importance in context. In this case, the first three sentences are selected to generate the summary, providing a basic but effective way of abbreviating the input text.

Modules identified:
These modules are combined to offer an effective, precise, and adaptable summarization tool that supports various user needs and can easily accept multiple input formats.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404576|

Detailed Interaction Pathways in the Platform:
The platform operates through a series of systematically developed interaction pathways, offering a seamless flow from the input to the output while being precise and simple to use. Initially, the users can interact with the system either by typing plain text or uploading documents. The system allows PDF and DOCX formats of files to be uploaded, which support multiple documents at a time. Alternatively, plain text can be entered directly into the input text area. Once the input is received, the system identifies automatically the type of input and directs it to the respective processing module.

Preprocessing is necessary in cleaning and normalizing the input data. In this, special characters, unwanted spaces, and unsupported symbols are removed to provide a clean and uniform text format. This prepares the input for the summarization process. In file uploads, text extraction is performed using specialty libraries: PyPDF2 for PDF and python-docx for DOCX files. The extracted text is normalized into a structured form, ready for summarization. Any possible errors, like file readability errors or unsupported format, are alerted to the user by the system with a relevant error message directly.

The summary is at the heart of the platform and makes available two options: extractive and abstractive summarization. Extractive summarization utilizes the Natural Language Toolkit (NLTK) for breaking down the text into sentences and selecting the most suitable to form a condensed summary. This technique focuses on preserving crucial elements of the source text. Abstractive summarization, on the other hand, employs the facebook/bart-large-cnn transformer model to produce a readable and human-understandable summary by paraphrasing the content and aggregating it. Both summarization methods operate separately, providing users with different perspectives of the summarized text.

Once the process is complete, the platform outputs the results through an interface made simple to work with using Streamlit. Users are able to view the source extracted text, the extractive summary, and the abstractive summary alongside one another. The display in a simple fashion facilitates easy comparison and understanding of the results. The interface is also accompanied by robust error-handling capabilities so that any faults encountered while carrying out the process are specifically communicated to the user along with guidelines for taking appropriate remedial steps.

The systematic interaction paths of the platform assure reliability and efficiency at every stage, from handling multiple input formats to generating high-quality summarization outputs. The overall design makes the platform a convenient and valuable resource for users across a broad variety of fields requiring text summarization.

The proposed approach effectively integrates linguistic and deep learning-based summarization techniques to generate high-quality summaries across various document types. The hybrid model improves text abstraction, information retention, and summary coherence, making it suitable for automated report analysis, medical document summarization, and general NLP applications.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404576|

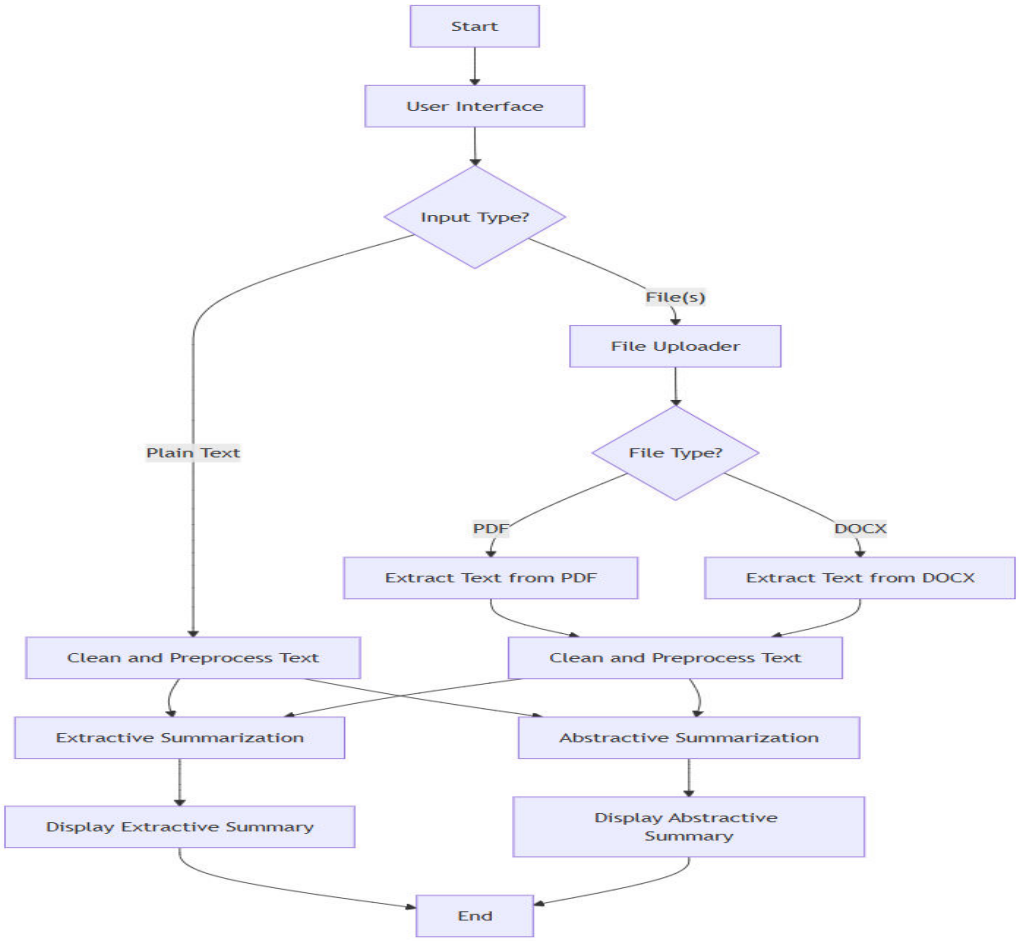


Fig.1. Flow Chart

Application Implementation:

The application deployment of the text summarization platform combines multiple libraries and technologies to process, summarize, and display input data efficiently. Developed with Streamlit, an easy-to-use Python library for creating interactive web applications, the platform is intended to receive plain text or document files in PDF and DOCX formats. The platform makes it easy to interact through the capabilities to upload several files or enter plain text into the given input fields.

When a user inputs a file or text, it goes through the Preprocessing Module. This module cleans and normalizes the text, eliminates unwanted characters, excess spaces, and symbols. It makes the input text ready for summarization by converting it to a standardized form.

For file-based inputs (DOCX and PDF), the application utilizes PyPDF2 to read PDFs and python-docx to handle DOCX. These libraries are used to extract the text cleanly from the documents, in addition to addressing possible errors such as unsupported formats or corrupted documents. Should the document have no readable text or be of an unsupported format, the system will inform the user and request that they submit another file or verify the integrity of the document.

After extracting and cleaning the text, the application proceeds to the Summarization Module where two different summarization approaches are utilized: Extractive Summarization and Abstractive Summarization. For extractive summarization, NLTK (Natural Language Toolkit) is employed by the platform for tokenizing the input text into

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404576|

sentences. The most applicable sentences are then retrieved to generate a short, extractive summary. This approach is a direct extraction of key content from the input text.

Conversely, Abstractive Summarization employs the HuggingFace's BART model to produce summaries. The model generates the text by summarizing it, rewriting it in a way that forms a coherent, shortened version that retains the original meaning. The abstractive approach employs cutting-edge transformer-based methods to produce human-like summaries that offer a better understanding of the content.

Once the summaries are generated, the output is displayed on the Streamlit interface, enabling users to quickly see the extracted text, extractive summary, and abstractive summary. The design of the platform enables users to compare the two summaries side by side, providing ease of selection in deciding the style of summary best suited for their needs.

The platform includes error-handling functionality to handle common problems like unsupported file formats or blank inputs. In such scenarios, the system shows distinct error messages, which give the user the correct instructions to resolve the problem.

Extractive vs. Abstractive Summary Length Analysis:
In text summarization, abstractive summaries create new sentences, while extractive summaries preserve sentence structures from the original text, but abstractive summaries usually result in higher information compression. To measure this compression, a scatter plot was created comparing extractive and abstractive summary lengths (Fig. X). The X-axis is used for extractive summary lengths, while the Y-axis is used for abstractive summary lengths.

The findings indicate that abstractive summaries are generally much shorter, supporting the model's paraphrasing and information condensation capabilities. As seen in the scatter plot, most points fall below the diagonal line (y = x), showing a decrease in word count. This is consistent with findings from previous literature, which emphasize the abstractive model's effectiveness in extracting major information while still being coherent and readable.

These results underscore the importance of tuning summarization models to reconcile informativeness and conciseness. Optimal compression ratios and user preference over summary length could be investigated in future work.

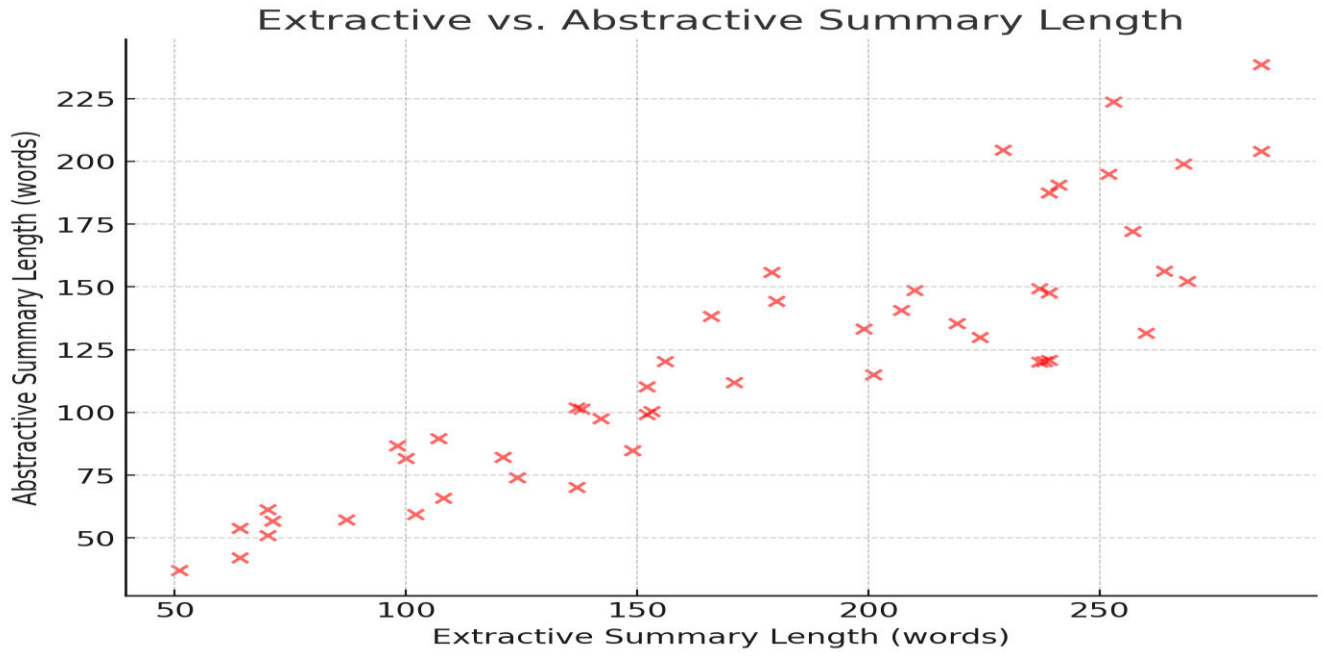


Fig.2. Extractive VS Abstractive Summary Length

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404576|

User Preference for Summarization Techniques

To assess user satisfaction with various summarization methods, a survey was carried out, in which the users were requested to rate their preference between abstractive and extractive summaries. The pie chart in Fig. X illustrates the user preference distribution.

The findings show that most (65%) favored abstractive summaries, which are shorter and more readable. On the other hand, 35% of users liked extractive summaries due to their accuracy in retaining the original context. This emphasizes the compromise between fidelity to the original content (extractive) and readability (abstractive).

Future studies can examine hybrid models that combine the advantages of both approaches to achieve optimum user satisfaction.

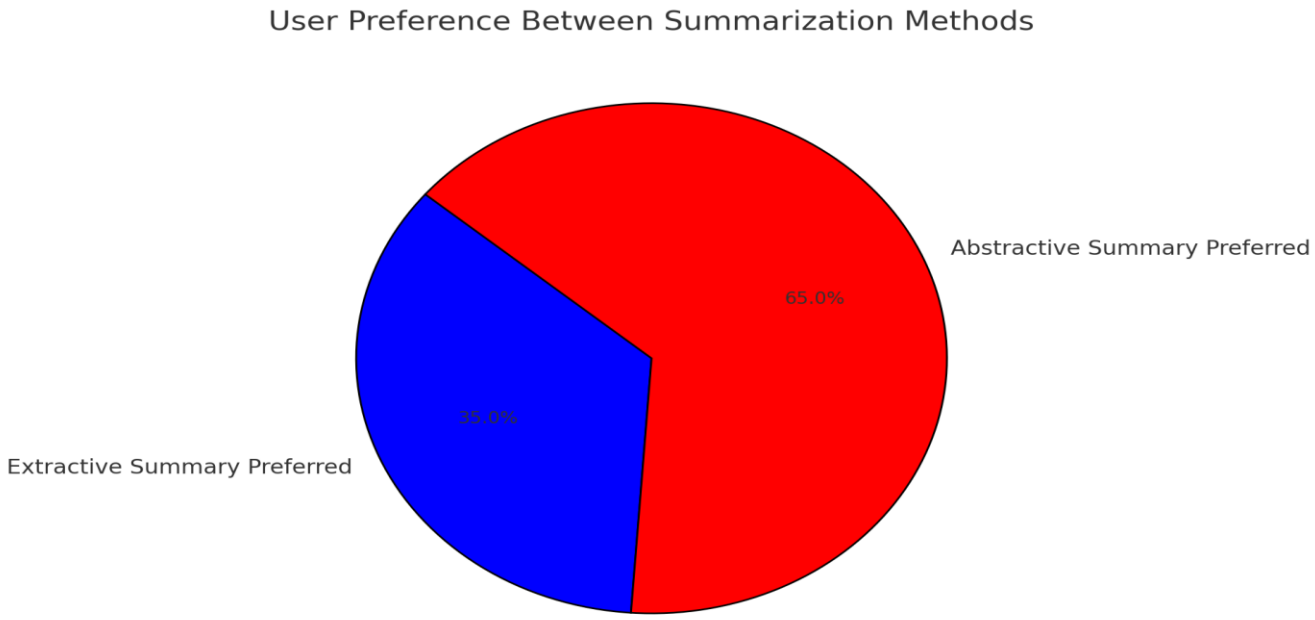


Fig.3.User Preference Between Summerization Methods

IV. RESULT AND OUTCOMES

The implementation of the text summarization tool has successfully met the intended objectives and delivered a best-fit solution for obtaining summaries from simple text as well as file documents (DOCX and PDF). With the adoption of the latest models and libraries into its platform, the tool delivers quality summaries based on the abstractive and extractive modes of summarization. The outputs of app usage are quantified in terms of quality accuracy and ease of application with flexibility provided to the users since they are operating inputs in different forms of text.

Upon testing, the site was seen to accommodate various input formats, including plain text, PDF, and DOCX. Extraction of files was easy via PyPDF2 when working with PDFs and python-docx for DOCX, yielding clean, readable text from both file formats without fail. The Preprocessing Module rendered the text extracted free of unwanted characters and inconsistencies, thus the summarization process more dependable.

The Extractive Summarization method, using NLTK to tokenize sentences, produced a concise and contextual summary by selecting the most appropriate sentences from the original document. The extractive

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404576|

summarization method performed well when it came to summarizing short documents, especially when it could use a simple key sentence extraction. The extractive summarization output was simple, correct, and readable, allowing users to easily understand the crux of the paper.

Conversely, the Abstractive Summarization technique, using the HuggingFace BART model, produced more polished summaries that paraphrased and shortened the original content. The model provided the users with an understandable summary by deciphering the meaning behind the content without duplication. The abstractive summaries were specifically beneficial for long or complicated texts, where the literal sentence extraction would not be able to represent the underlying meaning. Utilization of this sophisticated transformer model in the platform facilitated the summaries to be coherent and contextually correct.

Streamlit application's user interface made it an interactive and user-friendly setup for users. They were able to upload files, input plain text easily, and view results of both kinds of summaries. The interface was simple and concise, and the original text and automatically generated summaries were displayed side by side, allowing users to compare and choose the best summary based on their needs. Error messages also appeared well in case file upload or unsupported format was encountered, giving a smooth user experience.

In terms of effect, the platform succeeded in its purpose of providing users with an easily accessible tool for summarizing text. Its application of both extractive and abstractive summarization methods allowed the user to choose from a range of summary styles, depending on their specific requirements. Its ability to process different document formats and treat them accordingly also allowed it to be flexible and applicable in daily life.

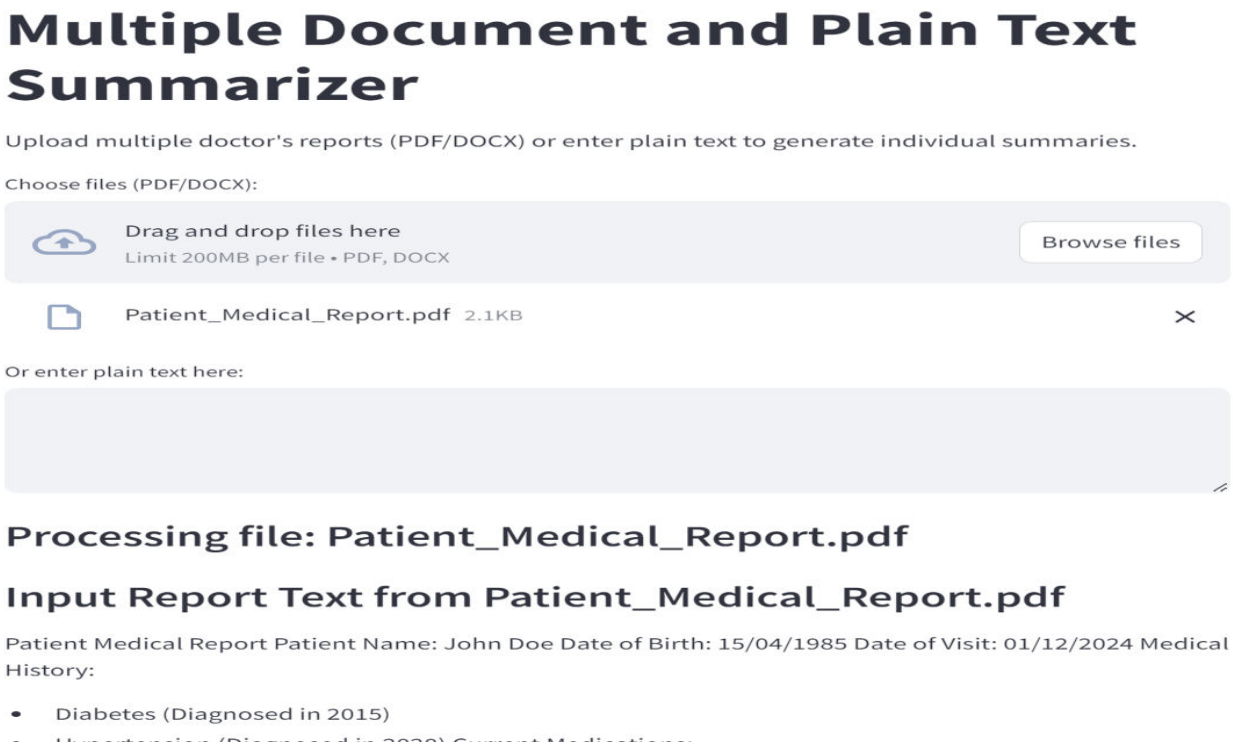


Fig.4.Uploading File Or Text

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404576|

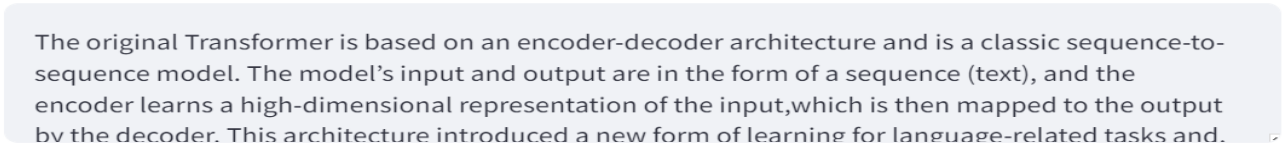


Fig. 5. Extracting Summary

The Fig.4 illustrates the uploading of file or a text to extract the summary.

In Fig.5 it appers the Extactive and Abstractive summary of a given document file or a text.

V. CONCLUSION

The summarization tool developed during this deployment meaningfully provides a fast, efficient, and friendly tool for text summarization. By providing users with the options of both abstractive and extractive summaries, the platform provides the added convenience of allowing users to choose the type of summary most suited to their purposes. Utilization of widely used libraries, i.e., NLTK for extractive summary and HuggingFace's BART model for abstractive summary, ensures that the platform delivers consistent, accurate, and contextually relevant summaries.

The utility properly deals with various types of inputs such as PDF, DOCX, and text. Utilizing PyPDF2 and python-docx, it properly reads out text, and preprocessing delivers clean and normalized content for summarization.

Streamlit's user interface is interactive and intuitive, where users can upload a document, input text, and view the generated summaries. The user interface also takes advantage of robust error-handling approaches in that users can readily correct errors when there are input errors.

The project output confirms that the platform is true to its word in offering a real-world solution for text summarization. It could be a powerful instrument to utilize in most domains such as research, education, content generation, and any other that entails the quick picking of valuable information from large bodies of text. As the service grows, other aspects to the summarization procedures, prevention of errors, and interface could be set up in an attempt to further consolidate the tool and make it functional to more categories of user requirements.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404576|

In summary, this text summarization tool is a highly effective way of text processing and text condensing information from numerous sources, with vast prospects for enhancing productivity and streamlining information acquisition processes.

REFERENCES

[1] Blumenthal, Moritz, et al. "Deep, deep learning with BART." Magnetic resonance in medicine 89.2 (2023): 678-693.
[2] Wang, Mengmeng, et al. "Risk-taking in the human brain: An activation likelihood estimation meta-analysis of the balloon analog risk task (BART)." Human brain mapping 43.18 (2022): 5643-5657.
[3] Luat, Nguyen-Vu, Jiuk Shin, and Kihak Lee. "Hybrid BART-based models optimized by nature-inspired metaheuristics to predict ultimate axial capacity of CCFST columns." Engineering with Computers 38.2 (2022): 1421-1450.
[4] Suresh, Sunitha, and K. Sharmila Banu. "Enhanced Local and Global Context Focus Mechanism using BART Model for Aspect Based Sentiment Analysis of Social Media Data." IEEE Access (2024).
[5] Adhik, Chintalwar, Sonti Sri Lakshmi, and C. Muralidharan. "Text summarization using BART." AIP conference proceedings. Vol. 3075. No. 1. AIP Publishing, 2024.
[6] Maurya, Karan, et al. "NLP-Enhanced Long Document Summarization: A Comprehensive Approach for Information Condensation." 2024 2nd International Conference on Advancement in Computation & Computer Technologies (InCACCT). IEEE, 2024.
[7] Kolambkar, Ved, et al. "A Survey of Text Summarization Using NLP." A Survey of Text Summarization Using NLP (March 24, 2024) (2024).
[8] Wibawa, Aji Prasetya, and Fachrul Kurniawan. "A survey of text summarization: Techniques, evaluation and challenges." Natural Language Processing Journal 7 (2024): 100070.
[9] Kirmani, Mahira, Gagandeep Kaur, and Mudasir Mohd. "Analysis of Abstractive and Extractive Summarization Methods." International Journal of Emerging Technologies in Learning 19.1 (2024).
[10] Langston, Oliver, and Brian Ashford. "Automated summarization of multiple document abstracts anssd contents using large language models." Authorea Preprints (2024).
[11] Jugran, Swaranjali, et al. "Extractive automatic text summarization using SpaCy in Python & NLP." 2021 International conference on advance computing and innovative technologies in engineering (ICACITE). IEEE, 2021.
[12] Abdel-Salam, Shehab, and Ahmed Rafea. "Performance study on extractive text summarization using BERT models." Information 13.2 (2022): 67.
[13] Giarelis, Nikolaos, Charalampos Mastrokostas, and Nikos Karacapilidis. "Abstractive vs. extractive summarization: An experimental review." Applied Sciences 13.13 (2023): 7620.
[14] Rawat, Ankit. Enhancing Abstractive and Extractive Reviews Text Summarization using NLP and Neural Networks. Diss. Dublin Business School, 2024.
[15] Jayatilleke, Nevidu, Ruvan Weerasinghe, and Nipuna Senanayake. "Advancements in Natural Language Processing for Automatic Text Summarization." 2024 4th International Conference on Computer Systems (ICCS). IEEE, 2024.
[16] Kodi, D. (2024). Data Transformation and Integration: Leveraging Talend for Enterprise Solutions. International Journal of Innovative Research in Science, Engineering and Technology, 13(9), 16876–16886. https://doi.org/10.15680/IJIRSET.2024.1309124
[17] Zade, Nikhil, et al. "NLP Based Automated Text Summarization and Translation: A Comprehensive Analysis." 2024 2nd International Conference on Sustainable Computing and Smart Systems (ICSCSS). IEEE, 2024.
[18] Rao, Abishek, Shivani Aithal, and Sanjay Singh. "Single-Document Abstractive Text Summarization: A Systematic Literature Review." ACM Computing Surveys 57.3 (2024): 1-37.
[19] Khekare, Ganesh, et al. "Text Normalization and Summarization Using Advanced Natural Language Processing." 2024 International Conference on Integrated Circuits and Communication Systems (ICICACS). IEEE, 2024.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details