Journal of Innovative Image Processing (ISSN: 2582-4252)
www.irojournals.com/iroiip/

TranscribEase – Speech and Document Summarizer

Nisarga S[1], Arpitha Devangavi[2], Prerana M Otageri[3], Prarthana N[4]

[1,3,4] Students, AI&ML, BNM Institute of Technology, VTU, Bengaluru, India

[2]Assistant Professor, AI&ML, BNM Institute of Technology, VTU, Bengaluru, India

E-mail: [1]nisargashashidhar@gmail.com, [2]arpithadevangavi@bnmit.in, [3]peruotu@gmail.com, [4]prarthananrupathunga@gmail.com

Abstract

This research introduces a summarization app using Flask and Hugging Face API. It summarizes audio, PDFs, and text efficiently. It features robust audio summarization by converting speech to text and then summarizing it. It also supports PDF and text summarization. The pre-trained NLP models like DistilBART, offers accurate and concise summaries. The user interface is intuitive, allowing seamless interaction. Overall, it's a comprehensive tool for extracting insights from various content sources, enhancing productivity.

Keywords: Flask, Hugging Face API, audio processing, PDF handling, NLP models, DistilBART, speech-to-text, summarization, productivity

1. Introduction

In today's fast-paced, information-rich environment, the challenge of extracting crucial insights from enormous data sets is constant. TranscribEase, is a powerful tool for summarizing the information built with Flask that represents a significant advancement in information management. This research is not limited to simply text condensing; it also has the ability to compress audio and PDF files. With the use of cutting-edge resources such as the DistilBART model and the Hugging Face API, users may quickly produce succinct summaries from a variety of sources, be it recording conversations or extracting information from publications.

DOI: https://doi.org/10.36548/jiip.2024.2.008
Received: 16.04.2024, received in revised form: 05.05.2024, accepted: 21.05.2024, published: 01.06.2024

TranscribEase also guarantees a flawless user experience in addition to summarizing. Users can engage with their content in a simple way thanks to features like audio summary and PDF display. To improve accessibility and usability, audio recordings are transcribed and summarized, while PDF documents are processed, compressed, and displayed with their summaries. Through the combination of cutting-edge technology and intuitive design, TranscribEase enables people to effectively navigate the digital sea of information. It's a catalyst for knowledgeable decision-making and active interaction with knowledge, not merely a tool for extracting data. In the effort to close the knowledge gap and create a more knowledgeable and capable society, TranscribEase emerges as a beacon of hope.

1.1 Motivation

Through speech-to-text conversion, the research improves inclusivity while addressing accessibility by bridging the gap between spoken language and written representation. By providing text summarization, it increases productivity by enabling users to quickly understand key concepts and save important time. Furthermore, by distilling information into succinct summaries, it helps users prioritize and make defensible decisions by preventing information overload. Additionally, it helps with knowledge retention by offering a way to reinforce learning by going over important details again. A more effective information processing experience is fostered by the project's flexibility in handling a variety of content formats, including PDFs. In general, it seeks to empower people through encouraging accessibility, boosting output, preventing information overload, strengthening retention of knowledge, and enabling interaction with a variety of content formats.

1.2 Problem Statement

An increasing number of people are in need of automated text summarization solutions to effectively extract important insights from complicated documents due to the massive amount of textual data that is produced every day. Our comprehensive text summarization system attempts to solve the problem of human summarization that is prone to error by enabling users to extract important information quickly and accurately. In the current digital environment, accessibility is essential, particularly for people with hearing impairments who have trouble accessing audio content. Our project aims to close this gap by giving audio summarization priority and offering resources for productive interaction with audio-based

materials such as podcasts and lectures. Our ultimate objective is to democratize textual information access, improving the lives of people with disabilities, students, professionals, and organizations in a variety of settings in a noticeable way.

1.3 Objectives

This project uses a strategic set of objectives to attempt to address the problems associated with information processing. First and foremost, the goal is to create a strong speech-to-text engine that can reliably translate a wide range of speech patterns and accents while navigating challenges such as background noise and specialized vocabulary. The project aims to apply efficient text summarization strategies by utilizing both extractive and abstractive techniques to reduce complex textual content to brief and enlightening summaries. Thirdly, an interface that is easy to use is needed to enable smooth transitions between text summarization and speech-to-text capabilities. This interface will have user-friendly features that make it simple to upload audio files, by choosing the preferred method of summarization, and easily access summaries that have been generated.

Finally, the project places a strong emphasis on multi-format compatibility, making sure that in addition to speech transcription, it can efficiently handle PDF documents for text summarization. By achieving these goals, the project hopes to offer a complete solution that will enable users to effectively navigate the large amount of information that is available.

1.4 Use Case

TranscribEase's effective summarizing features are advantageous to a variety of user groups. Key concepts from academic papers and lecture notes are easily understood by researchers and students. Summarizing presentations and meeting minutes is how business professionals increase productivity. For their multimedia content, content creators glean insights from podcasts and interviews. Experts in law and medicine examine documents and reports with efficiency. People with disabilities can efficiently access audio and textual content. TranscribEase helps language learners understand more effectively. It boosts individual productivity, helps schools accommodate different learning styles, supports podcasters with SEO and content accessibility, and assists conference planners in compiling discussions and insights from events.

1.5 Summary

Information extraction from a variety of sources, including text documents, audio files, and PDFs, is revolutionized by TranscribEase. Its cutting-edge speech-to-text technology and user-friendly interface enable users to effectively summarize content for a variety of uses. The main objective of the project is to accurately transcribe audio using state-of-the-art speech recognition technology. Users upload audio recordings, which are instantly transformed into editable text and easily summarized for quick understanding and memory retention. TranscribEase also expands its capabilities to include summarizing PDFs and text documents, supporting various domains. It's a smooth way to effectively process and assimilate complex information. TranscribEase is a useful tool for professionals, educators, content creators, students, and anyone looking to increase accessibility and productivity because of its robust performance and user-friendly design. It seeks to transform information processing and summarization for increased convenience and efficiency by utilizing AI and NLP.

2. Related Work

B. S. Anami et al. focuses on developing a Part-of-Speech tagging system for the Kannada language to improve the accuracy of text document summarization. The proposed system utilizes a combination of rule-based and machine learning approaches to accurately tag the parts of speech in Kannada text. Through experimentation and evaluation, it was observed that the developed POS tagging system significantly improved the accuracy of text summarization for Kannada documents. The results demonstrate the potential of the proposed approach for enhancing natural language processing tasks in Kannada, contributing to the broader goal of advancing linguistic technology for under-resourced languages. [1]

K. Padmanandam et al. meticulously address that several recent studies have highlighted the significance of visualization methods in enhancing the summarization process and improving the accessibility and understanding of information. The literature survey for this paper reveals a growing interest in the application of visualization techniques in the field of text summarization. Researchers have emphasized the potential of visualization approaches, such as the SRRDWC method, in effectively summarizing large volumes of text while preserving key content. Additionally, the incorporation of visualization in text summarization

has shown promising outcomes in various industries, including news media, research, and information retrieval. [2]

M. F. Mridha et al. present their paper “A Survey of Automatic Text Summarization: Progress, Process and Challenges," in IEEE Access in 2021. Automatic text summarization has seen significant progress in recent years, with the development of various approaches such as extractive, abstractive, and mixed methods. Extractive methods involve selecting and combining important sentences from the original text, whereas abstractive methods involve generating new, shorter sentences that convey the core meaning of the original content. Mixed methods aim to combine the strengths of both extractive and abstractive approaches. Challenges in automatic text summarization include maintaining coherence and fluency in generated summaries, handling multi- document summarization, and coping with the varying linguistic styles and genres of input text. [3]

Kavita Namdeo et al. in his paper presents various text summarization approaches using natural language processing techniques. Examine classical methods such as extraction-based summarization and abstraction- based summarization, as well as state- of-the-art approaches like deep learning and transformer-based models. Additionally, Evaluate the performance of these methods on different types of text data and discuss their advantages and limitations. Overall, this analysis aims to provide a comprehensive overview of text summarization techniques and their application in NLP. [4]

Vinnarasu et al. present their findings in the paper “Speech to Text Conversion and Summarization for Effective Understanding and Documentation" featured in the International Journal of Electrical and Computer Engineering. The study strategically employs machine learning methods, including Random Forests, Support Vector Machines (SVM), and TF-IDF, in the context of text summarization, providing a comprehensive examination of their effectiveness. Particularly notable is the incorporation of Seq2Seq models, a form of neural network architecture, showcasing the integration of advanced deep learning techniques. The paper not only delves into the technical aspects of these methodologies but also emphasizes their practical applications for enhancing comprehension and documentation processes. This research stands out in our literature survey as it bridges traditional machine learning with

contemporary deep learning approaches, offering a holistic perspective on the evolving landscape of natural language processing and information retrieval. [5]

K. Balasundaram et al. present their findings in the paper “Speech Document Summarization using Neural Network", featured in the 4th International Conference on Information Technology Research (ICITR), Moratuwa, Sri Lanka, 2019. This paper investigates speech document summarization through the lens of neural networks, specifically exploring the efficacy of Bidirectional Long Short-Term Memory Recurrent Neural Network (Bidirectional LSTM-RNN) and Convolutional Neural Network (CNN). The study delves into abstractive text summarization techniques, leveraging word embedding and recurrent neural networks to enhance the understanding of spoken content. [6]

This paper by K. Khadilkar et al. introduces a method for automatically summarizing speeches and essays using a knowledge graph-based approach. The study incorporates advanced techniques in natural language processing (NLP), including Named Entity Recognition (NER), Speech Analysis, and Knowledge Graphs. By focusing on Latent Semantic Analysis, the paper contributes to the merging of knowledge graphs and summarization methods. [7]

This paper, authored by A. Bhat et al. and published in IEEE, introduces a cost-effective audio-visual summarizer that leverages Natural Language Processing (NLP). The proposed approach integrates hardware infrastructure, an object detection classifier, the Open Source Computer Vision (OpenCV) library, speech recognition, and NLP techniques to summarize presentations and seminars. This innovative solution offers an efficient method for summarizing content. [8]

This IEEE publication by J. Choi et al. describes the design of a program that converts voice to text and manages it using the Google Cloud Speech API. The methodology involves developing a mobile app, processing on the server-side, integrating the API, and creating a web service. The paper highlights the practical implementation of the Google Cloud Speech API for text conversion. [9]

C. -I. Tsai et al., present an extractive speech summarization method using Convolutional Neural Network (CNN) techniques. The paper discusses the combination of CNNs and MLP for sentence selection, along with the incorporation of prosodic and lexical

features, to achieve abstractive text summarization. This research contributes to the progress of summarization methods by enhancing the capabilities of summarizing speech content. [10]

N. Sharma et al. present a real-time speech-to-text conversion system using bidirectional Kalman filter in Matlab. The methodology involves segmenting speech, applying speech detection algorithms, creating an acoustical model, and applying the bidirectional Kalman filter for noise reduction. This work emphasizes the application of advanced NLP techniques, particularly in the context of real-time speech processing. [11]

S. Manne et al. introduce a novel automatic text summarization system with feature terms identification. The methodology involves extractive summarization using Hidden Markov Model (HMM) tagger, Part-of-Speech (POS) tagging, and term frequency, showcasing the application of NLP and automatic text summarization based on POS tagging. [12]

Y. Liu et al. explore the use of N-best recognition output for extractive summarization and keyword extraction in meeting speech. The methodology involves employing Maximum Marginal Relevance (MMR) as the summarization technique, TF-IDF, and n-best recognition hypotheses for abstractive text summarization, contributing to advancements in meeting speech analysis. [13]

J. J. Zhang et al. introduces extractive speech summarization by active learning. The methodology incorporates active learning, acoustic and linguistic features, duration, speaking rate, F0 information, energy information, sentence length, TFIDF, and cosine similarity for extractive text summarization. This work provides insights into leveraging active learning strategies for improving the summarization outcomes. [14]

J. Mrozinski et al. focuses on automatic sentence segmentation of speech for automatic summarization, particularly in the context of broadcast news stories. The study discusses automatic summarization, language model training, and optimization for abstractive text summarization, contributing to advancements in spoken content analysis. [15]

C. M. Taskiran et al. discusses automated video program summarization using speech transcripts. The study explores summary visualization methods and speedup of playback, emphasizing the application of NLP techniques for speech recognition model development.

The work showcases practical applications of speech-to-text conversion for video content summarization. [16]

McKeown et al. explores text-to-speech summarization, particularly in the context of broadcast news stories. The study delves into automatic summarization, LM training, and optimization, highlighting advancements in abstractive text summarization. [17]

The paper "Speech summarization" by Furui et al provides an in-depth analysis of the challenges and techniques involved in speech summarization. The authors delve into the various methods used for extracting key information from spoken language and condensing it into a concise form. They explore the application of speech summarization in diverse fields such as automatic transcription, information retrieval, and audio indexing. The paper also discusses the potential impact of speech summarization on improving accessibility for individuals with hearing impairments and the potential role it can play in enhancing the efficiency of speech-based technologies. [18]

Chia-Hsin Hsieh et al. presents a cost-effective audio-visual summarizer utilizing Natural Language Processing (NLP). The approach combines hardware infrastructure, object detection classifier, Open-Source Computer Vision (OpenCV) library, speech recognition, and NLP techniques for summarizing presentations and seminars, providing a novel solution for efficient content summarization. [19]

3. Proposed Work

3.1 System Design

The architectural diagram of our system is shown in Figure 1 and is further explained in the ensuing explanation.

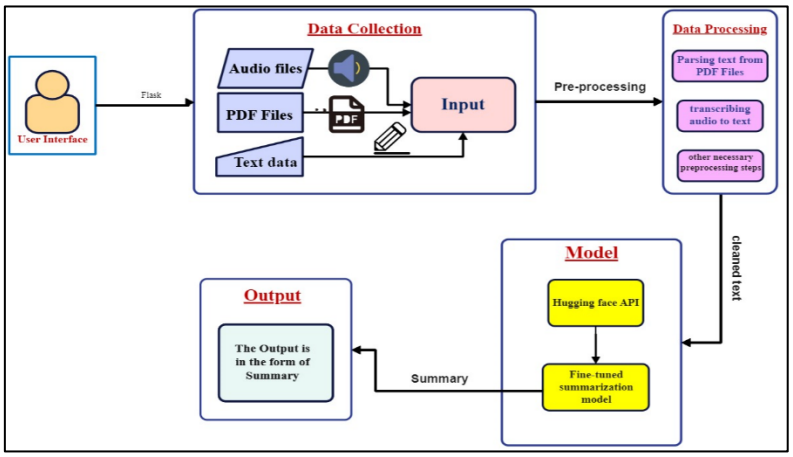

Figure 1. System Design

3.2 Dataset

Example 1: Rewriting News Articles

Input Text: A 500-word BBC news article about a scientific advance in fusion energy research

Summary of Results: The summary may include the main findings of the study, such as the kind of fusion reaction that was accomplished, the possible ramifications for the generation of clean energy, and any unresolved issues.

Example 2: Recapitulating a Real Wikipedia Article

Input Text: A 300-word passage from Wikipedia about the life and works of William Shakespeare

Summary: The synopsis could offer a succinct rundown of Shakespeare's life, his most well-known plays and literary works, as well as his enduring influence on English literature.

Example 3: Product Description Summarization

Input Text: A thorough 200-word synopsis of a new smartphone model from the company website.

Summarized Output: The main characteristics of the phone, such as its screen size, processor, camera specifications, battery life, and any special selling points, may be highlighted in the summary.

Example 4: Summarizing Customer Reviews

Input Text: A compilation of restaurant patron testimonials (approximately ten, each ranging from fifty to one hundred words).

Summary: The summary may reveal recurring themes in the evaluations, like compliments on the food, setting, and service, or drawbacks like lengthy waits or a small selection of options.

Example 5: Condensing Social Media Posts and Other User-Generated Content

Input Text: A collection of tweets (about ten, with a character limit of 280) regarding a recent athletic event.

Final Product: A succinct synopsis of the online conversation, a mention of significant events or players, and an overall feeling about the game could all be found in this summary.

These are only a few instances; the quality of the input text and the particular parameters that the DistilBART model uses will determine how well the summarization works.

a. User Interface

Three input modalities are available to users through the user interface: text, voice, and PDF. This interface acts as the user's entry point within the application. The interface is a web browser or a mobile app, which makes it easy for users to enter data. Because to the interface's intuitive and user-friendly design, users can quickly browse through the many options. The user interface is developed using tools and frameworks like HTML, CSS, and JavaScript to provide cross-platform and cross-device compatibility.

b. Flask Application

The Flask application manages incoming requests and regulates data flow, acting as the system's central nervous system. When a user chooses a specific input type, the request is sent to the relevant endpoint so that the Flask application can process it further. Building web

apps is made easier using Flask's lightweight and adaptable framework, which enables effective request and response handling.

c. Data Collection

Data collection involves capturing and storing user inputs in a selected format (text, audio or PDF) for further processing. Tools and libraries like queries and bottle are used to collect input data and save it in a suitable format. The collected data is then forwarded to the next stage for pre-processing.

d. Data Pre-processing

Pre-processing tasks are performed to ensure that the input data is suitable for the summarization model. This includes tasks such as data normalization, formatting conversion, and text cleaning. The input data is pre-processed to guarantee accuracy and consistency in the summarization process, improving the quality of the generated summaries.

e. Summarization Model

Pre-processing errands are performed to guarantee that the input information is appropriate for the summarization demonstrate. This incorporates assignments such as information normalization, organizing transformation, and content cleaning. The input information is pre-processed to ensure exactness and consistency within the summarization handle, making strides the quality of the produced rundowns.

f. Output

The Flask application gives the user with the condensed yield, completing the interaction circle. Users can see the outline on their device through the user interface, empowering them to rapidly get to and process the summarized substance. The yield is conveyed in a clear and brief arrange, upgrading client involvement and ease of use.

In general, this design utilizes the Hugging Face API to get to the DistilBART show for summarization, with Flask taking care of request handling, information collection, and reaction administration. The application coordinating different apparatuses, libraries, and

stages to form an effective and down to earth device for producing outlines from diverse input sources.

Table 1. ROUGE Score

	Recall(R)	**Precision(P)**	**F1-score(F)**
ROUGE-1	0.35	0.354	0.352
ROUGE-2	0.05	0.045	0.048
ROUGE-L	0.2875	0.291	0.289

ROUGE (Recall-Oriented Understudy for Gisting Assessment) is a set of measurements commonly utilized to assess the quality of content summarization frameworks by comparing produced rundowns to reference (ground truth) outlines. Here's a breakdown of the ROUGE scores mentioned in the Table 1:

I. ROUGE-1 (Unigram Overlap)

- Recall (R): 0.35

- This demonstrates that 35% of unigrams within the reference summaries are also present within the produced outlines.

- Precision (P): 0.354

- This implies that 35.4% of the unigrams within the produced outlines are to display within the reference rundowns.

- F1-score (F): 0.352

- The F1-score is the consonant mean of recall and precision, providing a single metric to degree by and large execution. It equalizes both recall and precision, with the next score showing way better execution.

II. ROUGE-2 (Bigram Overlap)

- Recall (R): 0.05

- As it were 5% of bigrams (sets of adjoining words) within the reference summaries are too show within the created summaries.

- Precision (P): 0.045

- Essentially, 4.5% of the bigrams within the created outlines are moreover show within the reference outlines.

- F1-score (F): 0.048

- The F1-score here shows the adjust between recall and precision for bigrams.

III. ROUGE-L (Longest Common Subsequence Overlap)

- Recall (R): 0.2875

- Around 28.75% of the longest common subsequences (arrangements of words showing up in both the created and reference outlines) are display within the created outlines.

- Precision (P): 0.291

- So also, 29.1% of the longest common subsequences within the created rundowns are moreover show within the reference rundowns.

- F1-score (F): 0.289

- The F1-score for ROUGE-L reflects the by and large adjust between review and accuracy for the longest common subsequence.

In summary, higher ROUGE scores for the most part demonstrate superior understanding between the created and reference rundowns. In any case, the translation of these scores can shift depending on the particular prerequisites and setting of the summarization errand.

3.3 Deep Learning Techniques' Performance Scores for Text Summarization Model

The performance ratings of the suggested deep learning techniques for the text summarization model—more especially, the distilbart-12-6-CNN model—are compiled in the following table. These ratings were taken from the Hugging Face page of the model.

Model Name	MM Params	Inference Time (MS)	Speedup	Rouge-2	Rouge-L
distilbart-12-6-cnn	306	307	1.24	21.26	30.59

3.4 Performance Evaluation Metrics

MM Params: The model's parameter count, expressed in millions.

Measured in milliseconds (MS), the inference time is the amount of time it takes the model to produce a summary.

Relative processing speed gain over a baseline model is known as speedup.

Rouge-2: The F1 score indicates how many bigrams overlap between the generated summary and the reference summary.

Rouge-L: The F1 score indicates how closely the generated summary and the reference summary overlap in terms of the longest common subsequence (LCS).

These metrics provide a comprehensive picture of the accuracy and efficiency of the model's performance. The distilbart-12-6-CNN model has good scores in both Rouge-2 and Rouge-L metrics, indicating a well-balanced trade-off between speed and summarization quality.

3.4 Dataflow Diagram

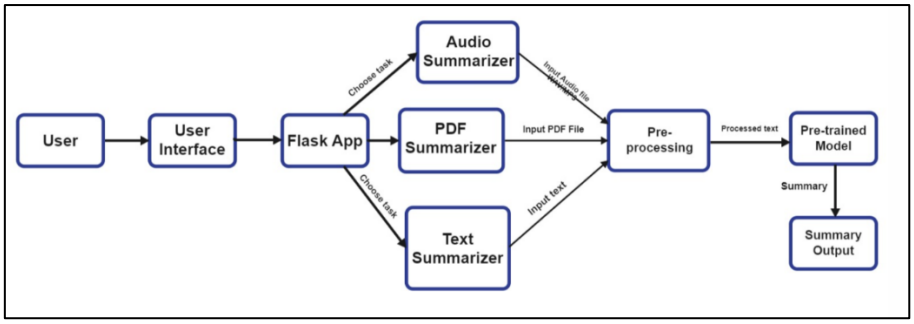

Figure 2. Dataflow Diagram

This is a detailed explanation of the steps involved in using this web application:

1) **User Interaction:** Through an intuitive web interface, users access the summarization application and select the type of summarization task they want to complete.

2) **Flask Application:** With its ability to handle requests and provide routing, the Flask application acts as the system's core. It controls how the user interface communicates with the different endpoints in charge of handling distinct summarization tasks.

3) **Select Task (Audio, Text, or PDF Summarizer):** Users are greeted with a simple and straightforward interface when they launch the application, which provides them with three options to choose from: audio, text, or PDF summarization. Every choice directs users to a unique endpoint designed to handle the corresponding kind of input.

4) **Audio Summarizer Endpoint:** Users who choose this option are redirected to an audio file upload endpoint in commonly supported formats. The Flask application installs the required libraries—like the speech recognition library—dynamically upon upload. The audio material is transformed into text format by the speech recognition library, making it appropriate for summarization. The final text is then sent to the summarization model for processing, and the user sees the generated summary on the same page in real time.

5) **Text Summarizer Endpoint:** When a user chooses this option, a text input field appears where they can type or paste the text they want to summarize. Without any intermediate steps, the input text is fed straight into the summarization model for processing. The user can instantly review and validate the summary by viewing it on the same page as it is generated.

6) **PDF Summarizer Endpoint:** A link is provided for users to upload their PDF files to this endpoint in order to summarize PDF documents. The PyPDF2 library is used to process the uploaded PDF files and extract text from the documents. Large PDF files may have their content chunked for faster processing, guaranteeing peak performance. The user is presented with both the extracted PDF content and the generated summary in an easy-to-understand format, making comparison and verification simple.

7) **Summarization Model (Hugging Face API):** The fundamental element for producing the summaries is the summarization model offered by the Hugging Face API, independent of the input type that the user chooses. The model uses cutting-edge methods for natural language processing to condense the input content into useful summaries that are suited to the needs of the user.

8) **Output:** The user is immediately notified of the summary results through the Flask application interface in the form of succinct summaries. Users can examine the summaries and engage in additional activities, like going back to the main page to complete more summarization tasks or investigating the application's other features.

9) **Model Training:** Provisions are made for the summarization model to be further trained using additional data, in addition to the application's main flow. This feature makes it possible for the model to be continuously enhanced and improved over time, guaranteeing its efficacy and applicability in dealing with changing user needs and content kinds.

The seamless flow of data and interactions within the summarization application as we can see in the dataflow diagram shown in Figure 2 is highlighted by this expanded data flow diagram, which offers a thorough overview of the entire process from user interaction to summarization model utilization.

4. Results and Discussion

You can find the most significant information from your uploaded content in the results section. This section is like having your own personal information concierge; it condenses long audio files, PDFs, or even plain text into a concise overview.

The summary that appears changes based on the kind of content you uploaded:

- **For Audio:** Picture someone summarizing a conversation for you. The audio is summarized in the results section, with a focus on the key topics covered. It functions as a kind of cheat sheet for the most important lessons from the recording.
- **For PDF:** Consider this a streamlined rendition of the PDF. The document's main ideas are succinctly summarized in the results section, which extracts the essence of the document. You quickly grasp the main points without having to sift through pages.
- **For Text:** The results section serves as a brief summary if you uploaded plain text. It summarizes the main points made in the text and helps you save time by concentrating on what matters most.

The DistilBART model is what makes this insightful summary so magical. With the use of this potent technology, the uploaded content is analyzed to determine its most important features. It then distills the essential information from the audio, PDF, or text into a succinct but informative summary.

Practical Takeaways: You can rapidly understand the main point of the content by using the results section. This enables you to:

- Using the information that has been summarized, make wise decisions.
- You can avoid having to read or listen to the entire content, saving you valuable time.
- Review the main ideas from a previous text, document, or conversation.

All things considered, the results section serves as your entry point for effective data retrieval. It gives you easily readable summaries so you can maximize the value of the content you upload.

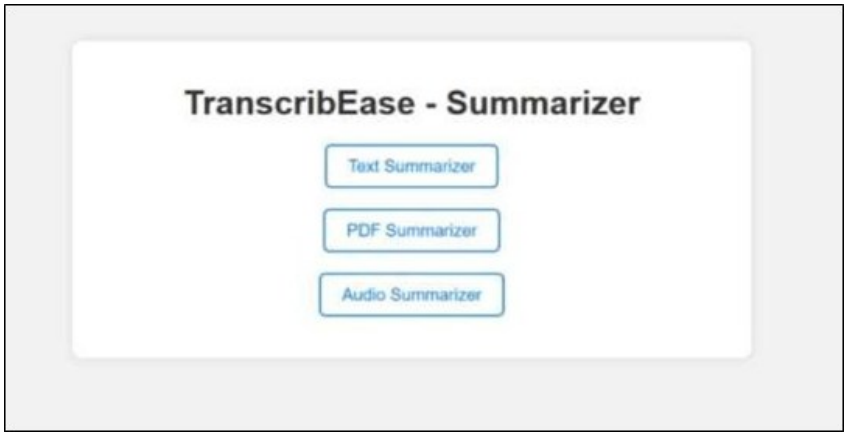

Figure 3. Home Page

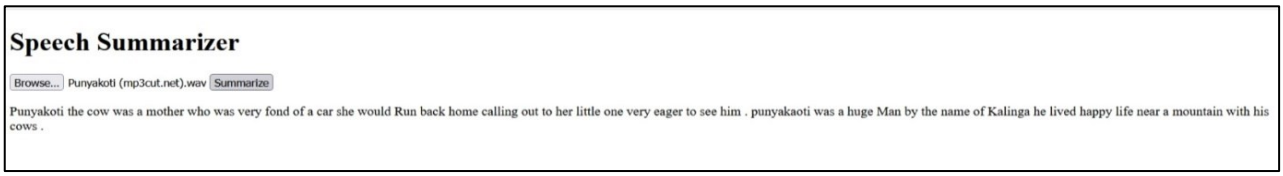

Figure 4. Speech to Text Summarization

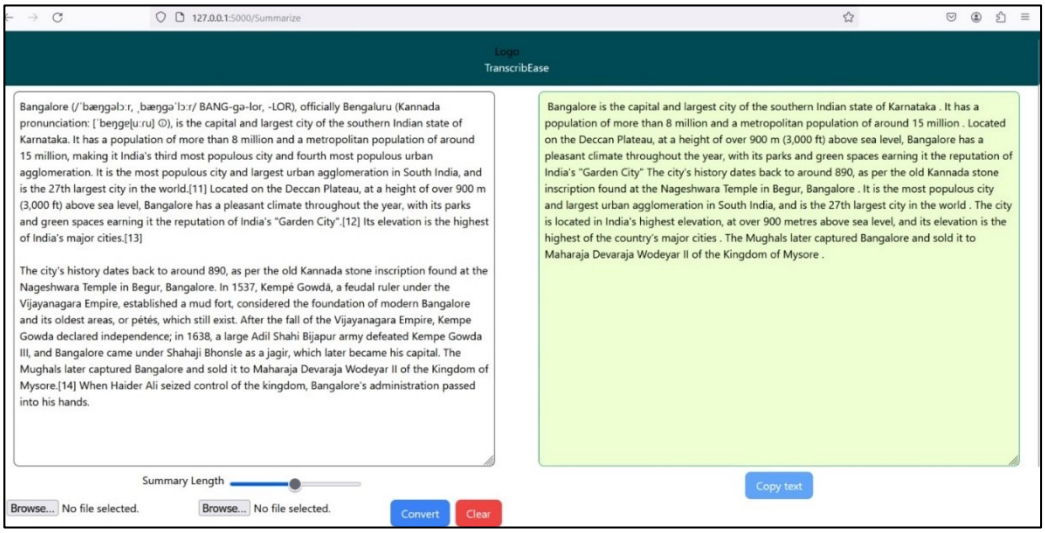

Figure 5. Summarization of Text

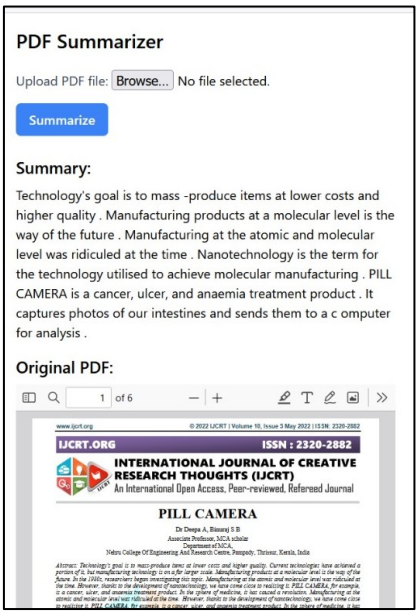

Figure 6. PDF Summarization

The main focus of this project is a web application made to make it easier to extract important data from a variety of sources, such as PDF and audio files. Let's dissect the various elements and how they work together:

- **User Interaction:** A user-friendly interface as shown in Figure 3 lets users upload PDF or audio files and interact with the web application. The provided code snippets imply the existence of this interface, even though it is not explicitly visible.

- **Flask Application:** The application's main processing unit for user requests is the Flask framework. Flask handles the file reception and processing process when a user uploads a file.

- **Audio Processing:** If the file you uploaded is audio, it will be processed to turn voice to text. The recognize_speech function accomplishes this by accurately transcribing the audio content, probably with the help of a speech recognition service. This is shown in Figure 4 where once the file is uploaded and Summarize button is clicked, the summary of the audio file gets generated below.

- **PDF Processing:** The extract_text_from_pdf function is used if the uploaded file is a PDF and the program supports PDF summarization as shown in Figure 6. Using the PyPDF2 library, this function extracts text from each PDF page so that it can be summarized.

- **Text Summarization:** The summarize_text function receives the text content after it has been retrieved, irrespective of its format (PDF or audio). This function creates a succinct synopsis of the given text by using the transformers library and the DistilBART model. The Summary of the copy pasted text is shown in the right-side window separately once the Convert Button is clicked. This Summary can be copied to clipboard directly as shown in the Figure 5.

- **Response to User:** Lastly, the user interface receives the generated summary back for display. Furthermore, in the event that a PDF file was uploaded, the interface might additionally show the original PDF file next to the summary for reference.

The project's overall goal is to give users a smooth experience when summarizing a variety of content sources by utilizing cutting-edge technologies like speech recognition and natural language processing to speed up the information retrieval process.

5. Conclusion

An important accomplishment is the development of the summarizer application via the Hugging Face API, which uses the DistilBART model. With Flask and the Hugging Face API, the project aims to create succinct summaries of text, audio, and PDF files. Ensuring robust

functionality, challenges such as data pre-processing and API integration were tackled. For practical uses such as document analysis and content summarization, the application appears promising. Future improvements could focus on sentiment analysis and entity recognition while optimizing for large-scale data sets. With its useful approach to text summarization, this project has the potential to improve productivity and information retrieval in a variety of fields.

References

[1] J. R, B. S. Anami and P. B. K, "Text Document Summarization Using POS tagging for Kannada Text Documents," 2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 2021, pp. 423-426, doi: 10.1109/Confluence51648.2021.9377106.

[2] K. Padmanandam, S. P. V. D. S. Bheri, L. Vegesna and K. Sruthi, "A Speech Recognized Dynamic Word Cloud Visualization for Text Summarization," 2021 6th International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India, 2021, pp. 609-613, doi: 10.1109/ICICT50816.2021.9358693.

[3] M. F. Mridha, A. A. Lima, K. Nur, S. C. Das, M. Hasan and M. M. Kabir, "A Survey of Automatic Text Summarization: Progress, Process and Challenges," in IEEE Access, vol. 9, pp. 156043-156070, 2021, doi: 10.1109/ACCESS.2021.3129786.

[4] Jadhav, Kavita Namdeo. "An Analysis of Text Summarization Approaches using Natural Language Processing." International Journal of Ingenious Research, Invention and Development, vol. 1, no. 2, 2023, p. 12. ISSN: 2583-648X

[5] Vinnarasu, A., and Deepa V. Jose. "Speech to Text Conversion and Summarization for Effective Understanding and Documentation." International Journal of Electrical and Computer Engineering, vol. 9, no. 5, [Year], pp. 3642-3648, doi: 10.11591/ijece.v9i5.pp3642-3648.

[6] K. Balasundaram and C. R. J. Amalraj, "Speech Document Summarization using Neural Network," 2019 4th International Conference on Information Technology Research (ICITR), Moratuwa, Sri Lanka, 2019, pp. 1-3, doi: 10.1109/ICITR49409.2019.9407785.

[7] K. Khadilkar, S. Kulkarni and S. Venkatraman, "A Knowledge Graph Based Approach for Automatic Speech and Essay Summarization," 2019 IEEE 5th International Conference for Convergence in Technology (I2CT), Bombay, India, 2019, pp. 1-6, doi: 10.1109/I2CT45611.2019.9033908.

[8] A. Bhat, A. C. Rao, A. Bhaskar, V. Adithya and D. Pratiba, "A Cost-Effective Audio-Visual Summarizer for Summarization of Presentations and Seminars," 2018 3rdInternational Conference on Computational Systems and Information Technology for Sustainable Solutions (CSITSS), Bengaluru, India, 2018, pp. 271-276, doi: 10.1109/CSITSS.2018.8768740.

[9] J. Choi, H. Gill, S. Ou, Y. Song and J. Lee, "Design of Voice to Text Conversion and Management Program Based on Google Cloud Speech API," 2018 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 2018, pp. 1452-1453, doi: 10.1109/CSCI46756.2018.00286.

[10] C. -I. Tsai, H. -T. Hung, K. -Y. Chen and B. Chen, "Extractive speech summarization leveraging convolutional neural network techniques," 2016 IEEE Spoken Language Technology Workshop (SLT), San Diego, CA, USA, 2016, pp. 158-164, doi: 10.1109/SLT.2016.7846259.

[11] N. Sharma and S. Sardana, "A real time speech to text conversion system using bidirectional Kalman filter in Matlab," 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Jaipur, India, 2016, pp. 2353-2357, doi: 10.1109/ICACCI.2016.7732406.

[12] S. Manne, S. M. Z. Pervez and S. S. Fatima, "A novel automatic text summarization system with feature terms identification," 2011 Annual IEEE India Conference, Hyderabad, India, 2011, pp. 1-6, doi: 10.1109/INDCON.2011.6139386.

[13] Y. Liu, S. Xie and F. Liu, "Using n-best recognition output for extractive summarization and keyword extraction in meeting speech," 2010 IEEE International Conference on Acoustics, Speech and Signal Processing, Dallas, TX, USA, 2010, pp. 5310-5313, doi: 10.1109/ICASSP.2010.5494972.

[14] J. J. Zhang, R. H. Y. Chan and P. Fung, "Extractive speech summarization by active learning," 2009 IEEE Workshop on Automatic Speech Recognition & Understanding, Moreno, Italy, 2009, pp. 392-397, doi: 10.1109/ASRU.2009.5373269.

[15] J. Mrozinski, E. W. D. Whittaker, P. Chatain and S. Furui, "Automatic Sentence Segmentation of Speech for Automatic Summarization," 2006 IEEE International

Conference on Acoustics Speech and Signal Processing Proceedings, Toulouse, France, 2006, pp. I-I, doi: 10.1109/ICASSP.2006.1660187.

[16] C. M. Taskiran, Z. Pizlo, A. Amir, D. Ponceleon and E. J. Delp, "Automated video program summarization using speech transcripts," in IEEE Transactions on Multimedia, vol. 8, no. 4, pp. 775-791, Aug. 2006, doi: 10.1109/TMM.2006.876282.

[17] McKeown, K., Hirschberg, J., Galley, M., & Maskey, S. (2005). From Text to Speech Summarization. Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). IEEE International Conference on, 5. doi:10.1109/ICASSP.2005.1416474.

[18] S. Furui, T. Kikuchi, Y. Shinnaka and C. Hori, "Speech-to-text and speech-to-speech summarization of spontaneous speech," in IEEE Transactions on Speech and Audio Processing, vol. 12, no. 4, pp. 401-408, July 2004, doi: 10.1109/TSA.2004.828699.

[19] Chia-Hsin Hsieh, Chien-Lin Huang and Chung-Hsien Wu, "Spoken document summarization using topic-related corpus and semantic dependency grammar," 2004 International Symposium on Chinese Spoken Language Processing, Hong Kong, China, 2004, pp. 333-336, doi: 10.1109/CHINSL.2004.1409654.

Author's Biography

Nisarga S - Final-year student pursuing a Bachelor's degree in Artificial Intelligence and Machine Learning at BNMIT, Bangalore. Passion for exploring the intricacies of AI, ML, and related domains. Primary interests lie in the intersection of cybersecurity and machine learning. Fascinated by the challenges posed by cyber threats and the potential of AI to mitigate them. Deep research into various techniques and algorithms.

Mrs Arpitha Devangavi - Highly esteemed Assistant Professor at B.N.M Institute of Technology (BNMIT), Bangalore, specializing in Artificial Intelligence & Machine Learning. Over seven years of experience in the field. Passionate about Machine Learning, Natural Language Processing, and Artificial Intelligence. Dynamic teaching methodologies and active involvement in research. Committed to fostering innovation and excellence within BNMIT's academic community. Dedicated to nurturing the next generation of technology leaders through mentorship and engagement in student-centric activities. Pushing the boundaries of knowledge in AI and ML.

Prerana M Otageri - Final-year student pursuing a Bachelor's degree in Artificial Intelligence and Machine Learning at BNMIT, Bangalore. Exceptional dedication and passion

for exploring the intricacies of AI, ML, and related domains. Primary interests lie in the intersection of Information Security and machine learning. Fascinated by the challenges posed by cyber threats and the potential of AI to mitigate them. Deep research into various techniques and algorithms. Also interested in learning and diving deep into big data analysis.

Prarthana N - Final-year student pursuing a Bachelor's degree in Artificial Intelligence and Machine Learning at BNMIT, Bangalore. Exceptional dedication and passion for exploring the intricacies of AI, ML, and related domains. Primary interests lie in Artificial Intelligence and Machine Learning. Fascinated by the challenges posed by cyber threats and the potential of AI to mitigate them. Deep research into various techniques and algorithms in Java development.