

International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

Volume 5, Issue 12, April 2025



Podcast Insights: Transcription and Summarization

Mahir Vashisht

Maharshi Dayanand University, Rohtak, Haryana

Abstract: This initiative, titled "Transforming Podcast Audio into Textual Summaries", focuses on streamlining the summarization of podcast episodes to enable users to efficiently comprehend the core ideas of extended audio material. The process starts with acquiring data, either in the form of podcast transcripts or audio recordings. Transcripts undergo preprocessing to eliminate irrelevant symbols and formatting. For audio, advanced speech-to-text technologies are utilized to transcribe spoken content. Summarization employs both extractive and abstractive approaches to generate concise outputs

Keywords: Podcast.

I. INTRODUCTION

In the current era of rapid digital advancement, audio content, especially podcasts, has surged in popularity. With their wide array of subjects and perspectives, podcasts offer a rich source of knowledge and enjoyment. Yet, the audio-only format presents obstacles: it is less accessible for individuals with hearing challenges, and locating specific details in long recordings can be both time-intensive and overwhelming. To address this issue, our initiative focuses on transforming podcast audio into text and creating concise summaries.

II. MOTIVATION

In the current era of rapid information exchange, audio content, especially podcasts, has gained immense popularity due to its variety of subjects and perspectives. However, the audio-only nature of podcasts presents accessibility challenges, particularly for individuals with hearing difficulties. Additionally, navigating lengthy recordings to locate specific details can be cumbersome. This project addresses these issues by converting podcast audio into text and producing succinct summaries.

The sheer volume of available audio content contributes to information overload, making it difficult for users to process without feeling overwhelmed. Language barriers further complicate access for non-proficient speakers. In a time-constrained world, efficiency is paramount, and advancements in artificial intelligence and machine learning offer opportunities to deliver brief, effective podcast summaries. This approach also enhances the visibility of high-quality but lengthy podcasts that may otherwise be overlooked due to time constraints.

III. APPLICATIONS

- Efficiency for Users: Summarized podcasts allow users to quickly understand key points without listening to entire episodes, saving valuable time.
- **Inclusivity**: Text summaries improve accessibility for those with hearing impairments or a preference for reading, expanding the podcast audience.
- **Content Development**: Summaries serve as valuable resources for creators, offering insights into industry trends and inspiration from influential voices.
- Educational Support: Summaries provide concise topic overviews, benefiting students and lifelong learners as supplemental learning tools.
- **Industry Monitoring**: Summaries enable media professionals to stay informed about industry developments quickly, offering insights into trends and news.

Copyright to IJARSCT www.ijarsct.co.in



DOI: 10.48175/IJARSCT-25929





International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

Volume 5, Issue 12, April 2025



IV. PROBLEM STATEMENT

Summarizing podcasts is essential because it saves time by providing key insights without requiring full listening, improves accessibility for those who prefer reading, enhances comprehension by distilling complex discussions into clear points, aids recall and easy sharing of important ideas, and boosts content visibility through SEO, making it more discoverable and engaging. It ensures valuable information reaches a wider audience in a convenient and efficient way. Podcasts, being audio-based, pose accessibility challenges for individuals with hearing impairments or those facing language barriers. The vast array of podcast topics leads to information overload, complicating the identification of relevant material. Time-sensitive podcasts into curricula or analyses due to the time-intensive nature of audio.

V. OBJECTIVES

- Build a system to convert podcast audio into text summaries, enhancing accessibility for hearing-impaired individuals and non-native speakers.
- Design an automated summarization algorithm to extract essential ideas and discussions, reducing information overload with concise summaries.
- Create a summarization tool tailored for education and research, delivering detailed yet brief summaries to support learning and academic work.
- Develop a real-time summarization system to process newly released episodes, ensuring timely and relevant summaries for users.



VI. SYSTEM ARCHITECTURE

Copyright to IJARSCT www.ijarsct.co.in



DOI: 10.48175/IJARSCT-25929





International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

Volume 5, Issue 12, April 2025



The system architecture comprises five critical stages, outlining the workflow and interactions between components:

- **Initiation**: The process begins when a user selects a podcast for summarization, triggering data retrieval and subsequent processing.
- Audio Retrieval: The system fetches the audio file from a podcast server or local storage, providing the raw content for analysis.
- **Speech-to-Text Conversion**: Advanced speech recognition technology transcribes audio into text, enabling further natural language processing.
- **Text Structuring**: The transcribed text is organized into a coherent format suitable for analysis, preserving meaning and context.
- Key Information Extraction: Natural language processing identifies and extracts critical topics and sentences, prioritizing essential content.
- **Summary Creation**: A summarization algorithm condenses the extracted information into a concise, coherent summary, retaining core insights.
- **Completion**: The process concludes with the delivery of the summary, ready for user interaction or distribution via the front-end interface.



VII. SEQUENCE DIAGRAM

1. Submit or Access Podcast File: The summarization process starts when a user shows interest in a specific podcast episode, typically by interacting with an application or interface. This interaction might involve selecting an episode or entering search keywords. Once the request is made, the system proceeds to locate and obtain the corresponding audio—either by downloading the file, accessing the podcast feed, or streaming it, depending on the platform's architecture. The user's choice is often guided by episode titles, descriptions, or tailored recommendations. This stage is essential as it connects user intent with system functionality, laying the groundwork for subsequent steps such as audio

Copyright to IJARSCT www.ijarsct.co.in



DOI: 10.48175/IJARSCT-25929





International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

Volume 5, Issue 12, April 2025



processing and content summarization. The quality of the system's response to the initial request greatly influences the user's overall experience and satisfaction.

2. Acquiring the Podcast Audio from the User: This phase focuses on obtaining the audio file of the podcast episode specified by the user. It plays a crucial role in preparing for the subsequent summarization process. Once the user initiates a request, the system activates the podcast player component, which interacts with external sources—such as podcast directories or databases—to find and retrieve the appropriate audio. These interactions, typically represented in a sequence diagram, involve communication between the user, the podcast player, and external platforms. Based on the system's design, the audio may be accessed via download, real-time streaming, or from a previously cached version. The podcast player acts as the central link in this process, ensuring successful data acquisition and enabling downstream functions like transcription and summarization. The efficiency of this step is essential for delivering fast and smooth access to the content, thereby improving the overall user experience.

3. Transcribe Audio into Text: Converting audio into text is a key step in the podcast summarization workflow. In this phase, the spoken content from the podcast is transcribed into written form, enabling downstream processing and analysis. After the audio is obtained, speech-to-text tools or APIs are employed by the audio processing module to generate a textual transcript. This transcript serves as the basis for later summarization tasks. The sequence diagram highlights the interaction between the audio processing system and external transcription services. Translating audio into machine-readable text makes it possible to apply natural language processing techniques. The accuracy of this transcription is critical, as errors can affect the quality and coherence of the final summary.

4. Extract Key Information and Generate Summary: The step titled "Identify Key Points and Summarize the File" serves as the central function of the podcast summarization system. In this phase, the system extracts critical information and produces a brief, informative summary of the episode. Once the audio is transcribed into text, natural language processing and machine learning techniques are applied to examine the content. These methods detect main ideas, recurring themes, and significant details throughout the transcript. A sequence diagram outlines the flow of messages and the interactions between various components involved in this process. The summarization algorithm works by eliminating redundancy, emphasizing important information, and distilling the core message of the episode. This generates a clear and concise summary for the user. The diagram also illustrates how different system components—such as audio processing, text analysis, and summarization—collaborate to fulfill this task. Effectively carrying out this stage results in a meaningful summary that reflects the user's interests and expectations.

5. Sending the Summarized File Back to the Server: This step, depicted in the sequence diagram, marks the final phase of the podcast summarization workflow prior to user access. Once the summarization module extracts key insights and creates a concise version, it sends the resulting text or data structure back to the server that initiated the podcast processing. This action finalizes the summarization operation, allowing the server to store the summary for future use, analysis, or caching, depending on how the system is architected. The server then enables user access to the summary, ensuring it can be easily retrieved and displayed via the user interface. By completing this step, the summary becomes fully integrated into the system, making it readily accessible. The sequence diagram highlights the two-way interaction between the server and the summarization module, showcasing the organized process that converts user input into a concise, usable output.

6. Presenting/Receiving the Summary: The "Presenting/Receiving the Summary" stage in the sequence diagram represents the final step of the podcast summarization workflow, where the system provides the generated summary to the end user. Once the summarized content is sent back to the server, the user interface communicates with it to retrieve or request the summary. The server responds by forwarding the condensed version of the podcast to the front-end, which could be a web app, mobile interface, or any other user-facing platform. This interaction is illustrated in the sequence diagram, showing the exchange of information between the server and the interface. The summary is then displayed in a clear and accessible format, enabling users to easily grasp the key points of the episode. This phase ensures that the summarization process delivers practical value, offering a concise overview to the listener. The diagram effectively demonstrates how backend operations and the user interface are connected to deliver a coherent and informative experience.

Copyright to IJARSCT www.ijarsct.co.in



DOI: 10.48175/IJARSCT-25929





2. Retrieve

from user

nodcast file

3. Analyze

audio and convert it into text format

 Identify key points and summarize the

file

1. Request or Provide

podcast file

6. Display/Recieve summary



5. Return

summarized file back to the server

VIII. CONCLUSION

This project provides a comprehensive solution to improve podcast accessibility and usability. By delivering concise text summaries, users can quickly understand episode content without listening to full recordings. The system identifies key themes and topics, offering a clear overview of the podcast's focus. Sentiment analysis and entity recognition add depth, providing insights into the discussion's tone and context. Generated metadata, such as keywords and categories, enhances content discoverability. Optional transcripts further promote inclusivity for hearing-impaired users and those preferring text. These features collectively create an enriched, user-centric podcast experience, meeting the diverse needs of modern audiences.

REFERENCES

- [1]. Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, *3*, 993–1022.
- [2]. Gupta, V., & Lehal, G. S. (2013). Automatic text summarization system for Punjabi language. *Journal of Emerging Technologies in Web Intelligence*, 5(3), 257–271.
- [3]. Kallimani, J. S., & Srinivasa, K. G. (2010). Information retrieval by text summarization for an Indian regional language. In *International Conference on Natural Language Processing & Knowledge Engineering* (pp. 1–4). IEEE.
- [4]. Liu, Y., & Lapata, M. (2019). Text summarization with pretrained encoders. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 3730–3740.
- [5]. Mittermaier, M., Serles, U., & Riss, U. (2023). Accessibility in digital audio: A review of podcast inclusivity for hearing-impaired audiences. *Journal of Accessibility and Design*, 5(2), 45–60.
- [6]. Nenkova, A., & McKeown, K. (2011). Automatic summarization. *Foundations and Trends in Information Retrieval*, 5(2–3), 103–233.
- [7]. Ramesh, G., Doddapaneni, S., Bheemaraj, A., Jobanputra, M., Ak, R., Sharma, A., Sahoo, S., Diddee, H., Kakwani, D., Kumar, N., et al. (2022). Samanantar: The largest publicly available parallel corpora collection for 11 Indic languages. *Transactions of the Association for Computational Linguistics*, 10, 145–162.

Copyright to IJARSCT www.ijarsct.co.in



DOI: 10.48175/IJARSCT-25929





International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

Volume 5, Issue 12, April 2025



- [8]. See, A., Liu, P. J., & Manning, C. D. (2017). Get to the point: Summarization with pointer-generator networks. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, 1073–1083.
- [9]. Shah, P., & Desai, N. (2016). A survey of automatic text summarization techniques for Indian and foreign languages. In *International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)* (pp. 375–380). IEEE.
- [10]. W3C. (2018). *Web Content Accessibility Guidelines (WCAG) 2.1*. World Wide Web Consortium. https://www.w3.org/TR/WCAG21/

Copyright to IJARSCT www.ijarsct.co.in



