PODCAST TRANSCRIPTION AND SUMMARIZATION WITH SPEECH SYNTHESIS

Dr. P.V. Kashid[1], Ajit Chourashiya[2], Darshan Ugalmugale [3], Harshvardhan Dalvi [4],Pallavi Dumbare [5]

*1 Assistant Professor, Department of Information Technology, Sir Visvesvaraya Institute of Technology, Nashik, Maharashtra, India
*2,3,4,5 Department of Information Technology, Sir Visvesvaraya Institute of Technology, Nashik, Maharashtra, India

---

Abstract - The proposed system first transcribes the audio from a podcast using automatic speech recognition (ASR), then summarizes the transcript using extractive text summarization, and finally returns the audio associated with the text summary. An important technology to realize human computer interaction is the technology of converting a given text into natural speech, that is speech synthesis. After transcription, the text is processed through natural language processing (NLP) algorithms to extract the most important points, key topics, and relevant information from the podcast.

***Key Words*:** Podcasts, audio summarization, automatic speech recognition, automatic summarization.

1.INTRODUCTION

Podcast transcription and summarization with speech synthesis represent an integrated approach to enhancing the accessibility and utility of podcast content. Through transcription, the spoken words in a podcast are meticulously converted into text, enabling a broader audience to engage with the material, including those with hearing impairments and individuals who prefer reading over listening. Suppose user queries for information on the internet he may get thousands of result documents which may not necessarily relevant to his concern.[1] To find appropriate information, a user needs to search through the entire documents this causes information overload problem which leads to wastage of time and efforts. To deal with this dilemma, automatic text summarization plays a vital role. Simultaneously, summarization condenses the key points and main ideas of a podcast episode into a concise format, providing a quick overview for busy individuals or those seeking a rapid grasp of the episode's highlights. The integration of speech synthesis technology further augments this process by utilizing artificial intelligence to convert the written text back into natural-sounding speech, replicating the original podcast experience for users who prefer auditory consumption. This amalgamation of transcription, summarization, and speech synthesis not only broadens the accessibility of podcasts but also facilitates content repurposing and amplifies search engine optimization efforts. It empowers content creators and listeners alike by fostering a more comprehensive and inclusive podcast experience, catering to diverse preferences and accessibility needs.[2]

2. PRIOR WORK

Podcast summarization has only recently gained interest as a research problem. As far as we know, our work is the first to tackle the problem of audio summarization for podcasts. Most prior methods for podcast summarization focus on generating text

summaries using abstractive summarization, where the original content has been rephrased. A
previous work also made use of ASR and text summarization but to generate short one-sentence summaries in the context of document retrieval for podcast search, and not for podcast audio summarization. [3]

A dataset has also been recently introduced for podcast search and abstractive text summarization. While text summaries can inform listeners about the topics in a podcast episode, they cannot convey subjective attributes that are typically found in the audio itself. Audio summaries would not only allow that, but they would also let users more conveniently listen to a summary rather than read one, for example, while cooking at home or driving. the basic assumption is that the vocal tract transfer function can be satisfactorily modeled by simulating formant frequencies and formant amplitudes. The synthesis thus consists of the artificial reconstruction of the formant characteristics to be produced. This is done by exciting a set of resonators by a voicing source or noise generator to achieve the desired speech spectrum, and by controlling the excitation source to simulate either voicing or voicelessness.[4]

The addition of a set of anti-resonators furthermore allows the simulation of nasal tract effects, fricatives and plosives. The specification of about 20 or more such parameters can lead to a satisfactory restitution of the speech signal. The advantage of this technique is that its parameters are highly correlated with the production and propagation of sound in the oral tract. The main current drawback of this approach is that automatic techniques of specifying formant parameters are still largely unsatisfactory, and that consequently, the majority of parameters must still be manually optimized[5].

3. METHODOLOGY

A. Podcast Summarization Architecture

Podcast Summarization first generates a transcript of the podcast audio using an ASR module and parses the text transcript into individual sentences [6]. It then uses a text summarization model to select relevant sentences, along with their time offsets in the audio, and generates the final audio summary associated with the text summary. Each stage is discussed in detail below.

1) Automatic Speech Recognition:

ASR methods perform the task of automatic speech-to-text transcription. As the purpose of this work is not to develop a new ASR system or improve on an existing one, we choose to use a well known and publicly-available solution for this task, namely AWS Transcribe.

2) Text Processing:

The transcripts obtained from the ASR module contain the text for the individual words and punctuation marks, their start and end times in the audio, and their confidence scores regarding the prediction. We choose to use an open-source library for NLP, namely spaCy3, to parse the text into individual sentences with their corresponding start and end times. Additionally, we force a sentence break when a pause of over two seconds between words occurs.

3) Text Summarization:

We generate text summaries by selecting relevant sentences from the transcripts, using automatic extractive summarization. We used the recently-proposed PreSumm4 model, which builds upon BERT to obtain a sentence level encoding, and stacks inter-sentence Transformer layers to capture document-level features for summarization. We fine-tune the

model to specifically handle podcasts summaries, by using our own dataset of summarized podcast transcripts.

B. Dataset Creation

Since there is no dataset available for extractive summarization of podcasts, we curated our own to support the development and evaluation of our system. We selected 19 unique podcast series from different genres, with an average (and standard deviation) of 16.3±6.28 episodes per series. The dataset contains 309 different podcast episodes, with an average duration of 36.5±19.8 minutes per episode, for a total of 188 hours of audio[7]. We built an annotation tool that presented an annotator with the sequence of sentences obtained from the transcript of an episode, as well as the metadata from the podcast feed and the original audio. Each sentence was paired with the respective audio segment using its time offsets. Additionally, the annotation tool dynamically generated audio and text summaries based on the annotator's selection, letting them verify their choices.

C. Model Training

We begin with the Pre Summ model (which was pretrained on the CNN/Dail -yMail dataset). We fine-tune the model on our podcast dataset for 10,000 steps, as described in. We use a learning rate of 10−4 and keep all the other parameters to their default values. The position embeddings of length 512 were deemed sufficient for our application, as the annotations in our dataset were contained within the first 512 tokens, even for longer episodes. Model checkpoints are saved and evaluated on the test set every 1,000 steps.[6] The best performing checkpoint parameters are used to report the performance of our system. We select the top-k sentences from the rank-ordered candidates to create the final summary.

D. Ablation Studies

1) Effect of number of candidate sentences: Similar to Pre Summ, we select the top-k sentences with the highest scores as our predictions. We study the effect of varying the number of sentences selected to represent the summary from the rank-ordered candidates in the model prediction. In our experiment, k ∈ (5, 9, 12, 15) was varied and the ROUGE- (1/2/L) F-scores are reported.

Metric	R-1 F1	R-2 F1	R-L F1
LEAD-k (baseline)			
$k = 5$	0.28 (0.02)	0.17 (0.03)	0.27(0.02)
$k = 9$	0.40 (0.03)	0.26 (0.04)	0.39(0.03)
$k = 12$	0.47 (0.03)	0.32 (0.03)	0.46(0.02)
$k = 15$	**0.52 (0.03)**	**0.39 (0.04)**	**0.51(0.03)**
PreSumm ($k = 12$)			
No FT	0.53(0.02)	0.38(0.02)	0.52 (0.02)
FT	0.63(0.03)	0.51 (0.03)	0.62(0.03)
FT + Aug	**0.64 (0.02)**	**0.53 (0.03)**	**0.63 (0.02)**
PreSumm (FT + Aug)			
$k = 5$	0.56 (0.03)	0.46 (0.04)	0.55 (0.03)
$k = 9$	0.63 (0.02)	0.52 (0.03)	0.62 (0.02)
$k = 12$	**0.64 (0.02)**	**0.53 (0.03)**	**0.63 (0.02)**
$k = 15$	0.63 (0.02)	0.53 (0.03)	0.62 (0.02)

Table 1 MEAN (AND STANDARD DEVIATION) OF THE ROUGE-(1/2/L) F-SCORES, FOR THE BASELINE, PRESUMM WITH k = 12, AND PRESUMM (FT + AUG) WITH k ∈ (5, 9, 12, 15), FOR A 5-FOLD CROSS VALIDATION. HIGHER SCORES ARE BETTER. BOLD VALUES ARE THE HIGHEST.

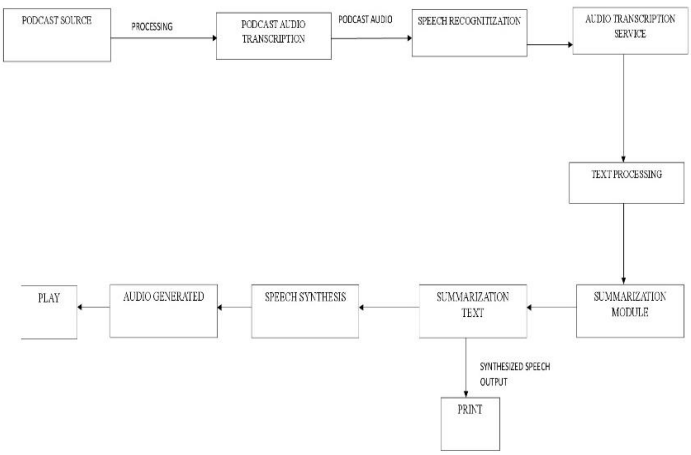

Fig. 1:- System Architecture

4. CONCLUSIONS

Our evaluation showed that the sentences selected by PodSumm largely agree with the ground truth. These extractive text summaries can then easily be used to produce the final audio summaries without altering the original content of the podcast, retaining subjective attributes such as production quality and presentation style [8]. This is a key difference and benefit of our system in comparison with other recent works on podcast summarization which can only provide text summaries and will alter the original content. Future works should include the creation of larger datasets for extractive summarization to help with the development of more accurate models. Given the subjective nature of audio summaries, gathering multiple annotations from different annotators for the same episode could further help to improve performance. More robust and podcast data-specific ASR and automatic summarization tools and/or proper post-processing mechanisms could correct potential transcription and summarization errors. Additional techniques such as speech/music classification and speaker diarization could also be beneficial as podcasts generally have background music and multiple speakers.

ACKNOWLEDGEMENT

The implementation and documentation of this project would not be succeeded without the kind support from individuals. First of all, we would like to express our special gratitude to Dr. P.V. Kashid who always gives us valuable advice and kind assistance to complete this project. Last but not least, we would like to thank the Faculty of Information Technology, Savitribai Phule Pune University for giving us the great knowledge. Finally, we would like to give our appreciation to our parents who support us since the beginning till the end of this project.

REFERENCES

[1] A. Vartakavi, A. Garg and Z. Rafii, "AudioSummarization for Podcasts," 2021 29th European Signal Processing Conference (EUSIPCO), 2021.

[2] U. Hahn and I. Mani, “of Automatic Researchers are investigating summarization tools and methods that,” in IEEE Computer 33.11, no. November, pp. 29–36, IEEE, 2000.

[3] D.H. Klatt “Review of text-to-speech conversion for English,” Journal of the Acoustical Society of America, vol. 82(3), 1987.

[4] A. Khan and N. Salim, “A review on abstractive summarization methods,” Journal of Theoretical and Applied Information Technology, vol. 59, no. 1, pp. 64–72, 2014.

[5] R. Ferreira, L. De Souza Cabral, R. D. Lins, G. Pereira E Silva, F. Freitas, G. D. C. Cavalcanti, R. Lima, S. J. Simske, and L. Favaro, “Assessing sentence scoring techniques for extractive text summarization,” Expert Systems with Applications, vol. 40, no. 14, pp. 5755–5764, 2013.

[6] R. A. Garc´ıa-Hernandez and Y. Ledeneva, “Word sequence models for ´ single text summarization,” in Proceedings of the 2nd International Conferences on Advances in Computer-Human Interactions, ACHI 2009, pp. 44–48, IEEE, 2009.

[7] K. Sarkar, “An Approach to Summarizing Bengali News Documents,” in International Conference on Advances in Computing, Communications and Informatics, pp. 857–862, ACM, 2012.

[8] Ms. Pratibha V. Waje, Dr. R. Jain, A Recommendation System for Execution Plans, Journal of Shanghai Jiaotong University, Volume 16, Issue 11, November - 2020,107-113.