Podcast Summarization System

Nisha Vanjari
Computer Engineering Department
K.J. Somaiya Institute of Technology
Mumbai, India
nvanjari@somaiya.edu

Sameer Pinjari
Computer Engineering Department
K.J. Somaiya Institute of Technology
Mumbai, India
nvanjari@somaiya.edu

Shubhada Labde
Computer Engineering Department
K.J. Somaiya Institute of Technology
Mumbai, India
nvanjari@somaiya.edu

Pallavi Patil
Computer Engineering Department
K.J. Somaiya Institute of Technology
Mumbai, India
nvanjari@somaiya.edu

Pradnya Patil
Computer Engineering Department
K.J. Somaiya Institute of Technology
Mumbai, India
nvanjari@somaiya.edu

Aarti Sahitya
Computer Engineering Department
K.J. Somaiya Institute of Technology
Mumbai, India
nvanjari@somaiya.edu

2022 5th International Conference on Advances in Science and Technology (ICAST) | 978-1-6654-9263-8/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICAST55766.2022.10039668

Abstract— **Podcasts are widely used by content creators to convey their thoughts,ideas and events in the form of audio. Podcasts are widely used in the streaming industry like spotify and it is showing continuous growth over the years. The whole podcasts are much more lengthy and noisy and to reduce the length of podcasts, there are many fine-tuned novel approaches to summarize it which are tested on professional datasets like Dai- lyMail,CNN and many more. In this paper, we will analyze such existing state-of-the-art approaches to summarize the podcast which laid the foundation of our idea to create a summarization platform based on that.**

***Keywords*— NLP, Podcast, speech summarization, neural networks**

I. Introduction (*Heading 1*)

In today's digital age, the usage of mobile phones has increased in number thereby creating a big deficiency in content which is quick and light to consume. Podcasts is one such medium of entertainment. But, they too can sometimes prove to be lengthy and time-consuming. But with the rise of this popularity comes several problems. Does the average working/studying professional today have the time to listen to these lengthy pieces of recordings in one go? The answer is No.

Instead, they can go through the summary of the podcast and decide if it is of their interest. Enables efficient Content Discovery and is time-saving. Helps search engines to sort podcasts according to keywords from the text sum- mary/description. Summaries will play a major role in Content Promotion. Podcast summarization will allow users to identify the content of their actual need. We have seen a lot of users who often rely on textual podcasts to discover the content of their needs. By summarizing podcasts, they can read and even listen to the important information conveyed by the content creator which reflects their ideas,events and preferences. We have proposed an idea to make such a platform where users can invest their time listening to segments of podcasts of their need.

II. LITURTURE SURVEY

We have studied over 15 research papers related to various Natural Language Processing (NLP) approaches to extract the summary text-based data from the whole text data. There may be a diverse range of text-based data including data containing many formal and informal words and sentences, data including linguistic words that may be used frequently in day-to-day life like data of day-to-day conversations of Whatsapp, Messenger, etc. There are methodologies include divide and conquer, automatic speech recognition, ROUGE score method, Pegasus, Big bird for audio and text-based summarization on the datasets including CNN, Daily mail, Pubmed, Arxiv.

Divide-and-Conquer Approach [1]

The divide and conquer approach can be applied to long texts such as financial reports and can be very effective if we combine this approach with the sequence to sequence RCNN model. This method breaks down long texts into smaller pieces that can be used to train the model, reduce the noise. Then these small pieces of data can be summarized easily as the model can be more trained with this approach. This method works fine with PubMed and arxiv datasets. ROUGE F1 scores can be improved, if the RUM units are used with a decoder of the model, unlike LSTM units. This method combined with a more powerful transformer based model such as the PEGASUS model gives far better results in terms of text summarization. There is a limitation with this approach as it is used with sequence to sequence and the transformer-based model which cannot be much effective with a very large input sequence as there is a limit for that.

Hierarchical Learning Approach for Large Input Sequences [2]

Podcasts generally contain larger input sequences which are one of the limitations in sequence to sequence RCNN model as discussed in the divide and conquer method. To solve this issue, Hierarchical Architecture based model is used. Hierarchical learning combined with transformer models based on sentence-level and token-level representations is used to get a better result and to eliminate the problem of long input sequences in plain sequence to sequence model. Here, Hyper parameter tuning is not extensive which is one of the limitation in this approach.

BERT [3]

978-1-6654-9263-8/22/$31.00 ©2022 IEEE

BERT stands for ”Bidirectional Encoder Representations from Transformers”. This approach uses only one output layer for fine-tuning as BERT is the pre-train bidirectional architecture that jointly conditions both left and right contexts of all layers. It is a powerful approach with GLUE and MultiNLI score above 80 percent. BERT can handle many NLP-related tasks with great accuracy.

A. Big Bird [4]

A transformer-based model such as BERT has the limitation that is its quadratic dependency. The Big Bird approach is a sparse mechanism that makes this quadratic dependency to linear along with the preservation of the functionalities of the quadratic dependency mechanism. This approach is a universal approximator of sequence to sequence functions. This approach increases the capacity of the input sequence up to 8 times its original capacity.

B. Deep Reinforcement Learning Approach [5]

A technique for empirical outcomes show that the proposed RNES model can adjust between the cross-sentence lucidness and significance of the sentences successfully, and accomplish cutting edge execution on the benchmark dataset. Improve- ment in the presentation of the brain intelligence model and presentation of human information into the RNES is required.

C. SummaRuNNer [6]

“SummaRuNNer” is a RCNN based model for summarization of documents which give better results compared to state-of-the-art. Being an interpreted model, it visualizes the prediction in three terms : information content, salience and novelty. It is a novel abstractive training mechanism which eliminates the need for extractive labels at training time. Here one limitation is that further exploration combining extractive and abstractive approaches is required.

D. NEWSROOM [7]

NEWSROOM is the largest known dataset containing data in the form of news. It is even more challenging to identify the keywords from such a dataset and summarize it. Here such a dataset can be used for training the existing state-of-the-art models to get the more efficient results. For future scope this dataset can also be used with different language processing strategies to obtain desired results.

E. Summarization of Financial Disclosures: MultiLing 2019 [8]

It describes the challenges faced while summarizing the financial terms or narrative disclosures mainly in English Language. The data mainly focuses on the business related data published in various sections of financial reports mainly UK reports.Participants were asked to generate a summary of such data in a pdf format. Summaries were then evaluated on the basis of ROUGE scores.

F. PEGASUS [9]

“PEGASUS stands for Pre-training with Extracted Gap-sentences for Abstractive Summarization”. In PEGASUS, im- portant parts of the sentences are removed from input and are generated as one output sequence. PEGASUS model on dataset containing news, scientific words, emails and a similar type of data achieves state-of-the-art performance measuredby ROUGE scores.

III. Existing System

From the survey, we have found that PEGASUS, Big bird, Divide and conquer are some of the great approaches that provide significantly good results over more complex datasets. These are NLP-based text summarization models that give great results with appropriate training.

Traditional audio and text summary methods need to improve with some new techniques for significantly better results. Although methods like Divide and conquer and ROGUE score methods were rigorously tested with some other techniques for more complex summaries. They together give far better results. Hierarchical Attention Transformer provides good results but hyper parameter tuning was not extensive. Neural coherence models provide great results for cross-sentence semantic and syntactic coherence patterns.

IV. Analysis and discussion

We will first run Automatic Speech Recognition (ASR) on the podcast to generate a transcript. Next, we will run Abstractive Text Summarization on the transcript. Lastly, this summary will be converted from text to speech. All of this data will be available to the user from the web app.

Our application focuses on creating a web application in which a content creator can upload its full podcast in the form of audio. Then the audio and text-based summary can be generated using a machine learning model which helps people to listen to the summarized podcast and can avoid listening to the whole length the of the podcast.

We conducted an extensive research of all the pre-trained summarization models currently being used in the industry. We calculated their accuracy by a metric names ROGUE scores whose comparison we have tabulated in the Table 1. After this comparison, we concluded that the BART pre-trained model works best for summarizing conversations which is the baseof communication in Podcasts.

TABLE I(a)(b)(c)

(a)TRAINING WITH ROUGE SCORES

Algorithm	Model Dataset	Fine-Tuned Dataset
Bart-Large	Xsum	Samsum
T5-large	Samsum	Deepspeed
Bart-Large	CNN Daily Mail	Samsum
Pegasus	Xsum	C4
Pegasus	CNN Daily Mail	C4

(b)VALIDATION WITH ROUGE SCORES

Algorithm	Validation		
	ROGUE-1	ROGUE-2	ROGUE-L
Bart-Large	54.392	29.808	45.154
T5-large	53.0823	28.7097	43.939
Bart-Large	42.621	21.983	33.034
Pegasus	-	-	-

(c)Testing WITH ROUGE SCORES

Algorithm	Testing		
	ROGUE-1	ROGUE-2	ROGUE-L
Bart-Large	53.306	28.355	44.095
T5-large	51.672	26.537	42.968
Bart-Large	41.317	20.872	32.134
Pegasus	45.20	22.06	36.99
Pegasus	43.90	21.20	40.76

Rouge scores (recall-oriented understudy for gisting evaluation) is basically a measurement for assessing programmed synopsis of texts as well as machine interpretations of texts. It works by comparing a naturally delivered interpretation against a bunch of reference summaries (typically human-created)

A. Proposed Idea

We will develop a web app using ReactJS where a content creator will upload the full length of the audio podcast. We will use MongoDB as database and GCP Storage Bucket to store audio files

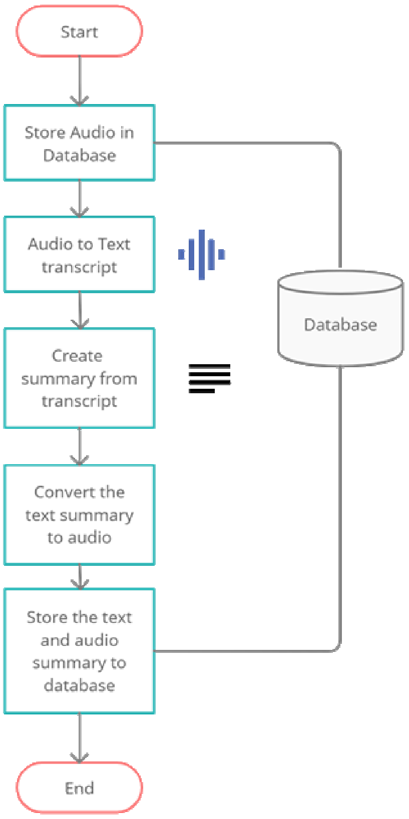
Start
Store Audio in Database
Audio to Text transcript
Create summary from transcript
Convert the text summary to audio
Store the text and audio summary to database
End
Database

Fig. 1. State Diagram

B. Software and Hardware Requirements

Software requirements:

- Language used: React.js, Python, Tensorflow.
- Database: MongoDB.
- To store audio files: GCP Storage

Bucket.Hardware requirements:

i7 CPU or above, 16GB RAM, CUDA-enabled GPU.

V. Fine-Tuning BART for Summarization

Despite the fact that BART has been utilized calibrate various Natural Language Processing assignments, its application to outline isn't as forward and simplified as other models. Since BART is prepared as a covered language model, the result vectors are grounded in tokens rather than sentences, while in an extractive rundown, most models manipulate sentence-level portrayals. Although, the fact that the division implants addresses different sentences in BART algorithm, they just apply sentences to match the inputs, while in a rundown one should encode and control multiple sentencedata-sources.

To address an individual sentence, we embed outer tokens toward the beginning of the sentences, and each image gathers highlights for the sentence going before it. We additionally use the intervals section implants to recognize multiple sentences inside a report. For sentiments we relegate portion inserting layers depending on whether one is odd or even. For instance, for the report, we would appoint implants.

Along these lines, report portrayals are learned progressively where lowering transformer layers address adjoining sentences, while the upper layers, in blend with self consideration, address multiple sentence talk. Position enclosings in the first BART model have the greatest length of 512; we overcome these restrictions by adding more positions that will be instated haphazardly and fine-tuned with different boundaries in encoder.

VI. IMPLEMENTATION

In this section, we portray the outline datasets utilized in our experiments and talk about various implementation details.

A. Summarization Datasets

Our model was evaluated using two benchmark datasets, more specifically XSum and SAMSUM. These datasets support a variety of summarization styles, from features to very short one-sentence summaries. The outlines also change in terms of the type of post-processing task they embody (for example, some show more cut-and-paste operations, others are very abstract). The Table II shows the measurements for these datasets (test sets). An overview of the model (highest quality level) is included in the supplementary material.

B. Summarization Models

We applied a dropout (probability 0.1) before every linear layer in every summary model. The Transformer decoder has 768 hidden units and all feedforward layers have a hidden size of 2,048. All models were trained in 1,000 steps on one GPU (NVIDIA Tesla K80). The model's checkpoints were saved and scored at each step in the validation set. We selected the top three control points based on the score loss in the validation set and reported the average score in the test set. During decoding, We used a ray search (size 5) and adjusted the α so that the length penalty for the validation set went from

0.6 to 1. Abstract Despite its popularity with abstracts, it is worth noting that our decoders do not employ a copy or cover mechanism. This is primarily because the focus is on building the model with minimal requirements, and these mechanisms may introduce additional tuning hyper-parameters.

VII. CONCLUSION

The purpose of creating this application is to provide text and audio based summary that can be extracted from the whole length of the audio podcast so that any working professional/student or any other person who is busy in day to day working tasks but still wants to listen to the podcast can use our application. Because summaries provide all the highlighted points necessary to listen avoiding non-essential sentences/words.

REFERENCES

[1] PodSumm: Podcast Audio Summarization. Aneesh Vartakavi and Aman-meet Garg, Gracenote Inc.

[2] A Divide-and-Conquer Approach to the Summarization of Long Doc- uments. Alexios Gidiotis and Grigorios Tsoumakas. IEEE/ACM Trans- actions on audio, speech, and language processing.

[3] Hierarchical Learning for Generation with Long Source Sequences. Tobias Rohde, Xiaoxia Wu , Yinhan Liu. Birch AI, Seattle, WA. Paul G. Allen School of Computer Science & Engineering, University of Washington, Seattle, WA.

[4] Big Bird: Transformers for Longer Sequences. Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, Amr Ahmed. Google Research.

[5] BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova. Google AI Language.

[6] Deep Reinforcement Learning for Sequence-to-Sequence Models. Yaser Keneshloo, Tian Shi, Naren Ramakrishnan, Chandan K. Reddy, Senior Member, IEEE.

[7] Learning to Extract Coherent Summary via Deep Reinforcement Learn- ing. Yuxiang Wu, Hong Kong University of Science and Technology. Baotian Hu, University of Massachusetts Medical School MA, USA.

[8] SummaRuNNer: A Recurrent Neural Network-based Sequence Model for Extractive Summarization of Documents. Ramesh Nallapati, Feifei Zhai, Bowen Zhou. 1011 Kitchawan Road, Yorktown Heights, NY 10598.

[9] Get To The Point: Summarization with Pointer-Generator Networks. Abigail See, Stanford University. Peter J. Liu, Google Brain. Christopher D. Manning, Stanford University.

[10] NEWSROOM: A Dataset of 1.3 Million Summaries with Diverse Extractive Strategies. Max Grusky, Mor Naaman, Yoav Artzi Department of Computer Science of Cornell Tech Cornell University, New York, NY10044.

[11] MultiLing 2019: Financial Narrative Summarisation. Mahmoud El- Haj, School of Computing and Communications, Lancaster University, United Kingdom.

[12] PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization. Jingqing Zhang, Yao Zhao, Mohammad Saleh, Peter J. Liu.