Summary Extraction and Relevance Identification from Spotify Podcasts (Project Proposal)

Arijit Ghosh Chowdhury
arijit2@illinois.edu
University of Illinois, Urbana Champaign

Dattatreya Mohapatra (Captain)
dm42@illinois.edu
University of Illinois, Urbana Champaign

Gargi Balasubramaniam
gargib2@illinois.edu
University of Illinois, Urbana Champaign

1 INTRODUCTION

With the vast amount of data circulating in the digital space, there is need to develop machine learning algorithms that can automatically shorten longer texts and create succinct summaries. Applying text summarization reduces reading time, accelerates the process of researching for information, and increases the amount of information that can fit in an area.

Inspired by the need to understand more about spoken-content retrieval - the given scenario is especially relevant in the domain of podcasts, which have become a widely used medium of communication today.

In this project, we take inspiration from the TREC 2020 and 2021 Podcast Summarization Challenges [2]. The TREC 2020 Podcast track features transcript and audio datasets from Spotify - a leading platform for streaming podcasts.

2 PROBLEM DESCRIPTION

Given the automatic transcript of a podcast episode, the goal is to attempt the following tasks:

(1) Generate meaningful summaries using multiple baseline models (see Section 4) and compare them to identify the best performing method
(2) If time permits, identify worst performing summaries and modify baseline models to address their challenges

We plan to attempt task 1 as part of the scope of this project and will undertake task 2 if time permits.

Our approach involves conducting a qualitative and qualitative evaluation of 3 techniques - TextRank [5], Seq2Seq Pointer Networks [6] and BERT [7]. We plan to increase complexity in a step by step manner, and observe improvements in the summarization output. To this end, we hope to come up with ways to improve existing baselines after thorough analysis.

2.1 Motivation

There has been a lot of work on text summarization and snippet ranking that involve both shallow ranking methods, as well as deep learning architectures. However, the development of most of the models was based on a critical assumption - structured text data. Podcasts present a genre of datasets where the style of text can vary from very formal to very casual depending on the speakers. Podcast transcripts also contain text from multiple sources (speakers), which is another aspect that existing models do account for. These two reasons cascade various other consequential challenges to processing podcast transcripts using baseline models[3]. We aim to address these challenges by implementing existing models (see Section 4) and comparing the generated summaries against manual summaries. If time permits, we will also explore ways to modify existing models to work with the unique traits of podcast data.

2.2 Relevance to CS 410

Text summarization is essentially a generative modelling task. It also depends on the sentence and language features. Furthermore, in case of extractive methods, there can be multiple candidate summaries for a podcast transcript which have to be ranked using similarity metrics. CS410 covers all of these topics and has provided us with solid background to be able to tackle these tasks on a real-life noisy dataset. We hope to use the content of the lectures and expertise of course staff to explore a new area of application for these concepts and solve an actual problem.

3 DATA

The Spotify Dataset [1] contains around 105,360 episodes from various podcasts on Spotify. The GCP Text-to-Speech API was used to generate transcriptions. In this work, we limit our scope to the transcriptions of the summaries, and do not work on audio data. The target summaries are scored from Bad (B) to Excellent (E).

4 METHODS

For our case study, we use three baselines with increasing amount of complexity.

- **TextRank [5]** : The algorithm ranks sentences based on their similarity to other sentences. The basic idea implemented by a graph-based ranking model is that when one node links to another one, it is basically voting for that other node. The higher the number of votes that are cast for a node, the higher the relevance of the node.
- **Seq2Seq Pointer-Generator Networks [6]** : This abstractive summarisation technique uses a hybrid pointer-generator network that can copy words from the source text via "pointing", which allows accurate representation of information, while keeping the ability to produce novel words through the generator. It also uses coverage to keep track of what has already been summarized, which discourages repetition.
- **BERT [7]** : This paper uses two-stage decoding process to leverage BERT's context modeling ability. On the first stage,

Text Information Systems, , Fall '21', UIUC

this method generates the summary using a left-context-only-decoder. On the second stage, every word of the summary is masked and the refined word is predicted one-by-one using a refine decoder.

5 IMPLEMENTATION DETAILS AND WORKLOAD

All models will be trained under the same hardware and system configurations. For this project Google Colab will be used and the neural methods will make use of the Nvidia Tesla K80 GPU. All programming will be done in Python. The chosen deep learning framework is Pytorch, and other numerical and machine learning libraries like numpy, pandas and sklearn will also be widely used.

Estimated Workload : Estimated time taken for each method is 15 hours, which includes training across multiple hyperparameters, evaluating and documenting results. Analysing the dataset and preprocessing is estimated to take another 10 hours of effort. Another 5-10 hours are kept for collating all the results and documenting final observations and outlining practical prospects of this case study. Total estimated work hours = 65 hours.

6 EVALUATION

For this project, we will use the commonly used automatic evaluation metric ROUGE [4]. Additionally, we will use five english speaking volunteers to score the summaries into the defined sprectrum of Bad(B) to Excellent(E).

7 AUTHORS AND AFFILIATIONS

Arijit Ghosh Chowdhury, Dattatreya Mohapatra and Gargi Balasubramaniam are first year MS CS students at the University of Illinois at Urbana Champaign. Arijit has a background in NLP Research in the social media and content space. Dattatreya's research focuses on graph mining and search query understanding. Gargi has previous experience with basic tools of Machine Learning and Deep Learning, with a research background in interpretable machine learning. This project will be a first learning experience in the text domain.

REFERENCES

[1] Ann Clifton, Sravana Reddy, Yongze Yu, Aasish Pappu, Rezvaneh Rezapour, Hamed Bonab, Maria Eskevich, Gareth Jones, Jussi Karlgren, Ben Carterette, and Rosie Jones. 2020. 100,000 Podcasts: A Spoken English Document Corpus. In *Proceedings of the 28th International Conference on Computational Linguistics*. International Committee on Computational Linguistics, Barcelona, Spain (Online), 5903–5917. https://www.aclweb.org/anthology/2020.coling-main.519

[2] R. Jones, Ben Carteree, Ann Clion, Maria Eskevich, G. Jones, Jussi Karlgren, Aasish Pappu, S. Reddy, and Yongze Yu. 2020. TREC 2020 Podcasts Track Overview. *ArXiv* abs/2103.15953 (2020).

[3] Rosie Jones, Hamed Zamani, Markus Schedl, Ching-Wei Chen, Sravana Reddy, Ann Clifton, Jussi Karlgren, Helia Hashemi, Aasish Pappu, Zahra Nazari, Longqi Yang, Oguz Semerci, Hugues Bouchard, and Ben Carterette. 2021. Current Challenges and Future Directions in Podcast Information Access. *CoRR* abs/2106.09227 (2021). arXiv:2106.09227 https://arxiv.org/abs/2106.09227

[4] Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*. Association for Computational Linguistics, Barcelona, Spain, 74–81. https://aclanthology.org/W04-1013

[5] Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing Order into Text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Barcelona, Spain, 404–411. https://aclanthology.org/W04-3252

[6] Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get To The Point: Summarization with Pointer-Generator Networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Vancouver, Canada, 1073–1083. https://doi.org/10.18653/v1/P17-1099

[7] Haoyu Zhang, Jingjing Cai, Jianjun Xu, and Ji Wang. 2019. Pretraining-Based Natural Language Generation for Text Summarization. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*. Association for Computational Linguistics, Hong Kong, China, 789–797. https://doi.org/10.18653/v1/K19-1074