Transforming Podcast Preview Generation: From Expert Models to LLM-Based Systems

Winstead Zhu and **Ann Clifton** and **Azin Ghazimatin** and **Edgar Tanaka** and **Ward Ronan**

Spotify

{winsteadx,aclifton,azing,edgart,edwardr}@spotify.com

Abstract

Discovering and evaluating long-form talk content such as videos and podcasts poses a significant challenge for users, as it requires a considerable time investment. Previews offer a practical solution by providing concise snippets that showcase key moments of the content, enabling users to make more informed and confident choices. We propose an LLM-based approach for generating podcast episode previews and deploy the solution at scale, serving hundreds of thousands of podcast previews in a real-world application. Comprehensive offline evaluations and online A/B testing demonstrate that LLM-generated previews consistently outperform a strong baseline built on top of various ML expert models, showcasing a significant reduction in the need for meticulous feature engineering. The offline results indicate notable enhancements in understandability, contextual clarity, and interest level, and the online A/B test shows a 4.6% increase in user engagement with preview content, along with a 5x boost in processing efficiency, offering a more streamlined and performant solution compared to the strong baseline of feature-engineered expert models.

1 Introduction

Podcasts, videos, and other long-form talk content, have become flourishing media, offering diverse content that caters to a wide range of audiences. Discovering new content, however, remains challenging, as the long-form nature of episodes demands significant time investment to assess their relevance (Jones et al., 2021). Previews, which are short and representative segments of an episode, provide a solution by capturing engaging, self-contained moments that are easy to understand without additional context (Barua et al., 2025).

Generating effective previews from episodes that can exceed an hour is a challenging task and requires robust content understanding. For example, to locate self-contained segments, previous work suggests using segmentation methods to detect topic transitions (Lukasik et al., 2020; Liu et al., 2022a; Retkowski and Waibel, 2024; Ghazimatin et al., 2024). These methods, however, may miss many interesting moments depending on the granularity of segmentation. Furthermore, they typically fail to distinguish between segments containing commercial content such as ads or self-promotions which do not represent the whole content.

Traditional preview extraction approaches often rely on sophisticated feature engineering to derive aggregations of expert models such as sentiment analysis, topic modeling, speech classification, and ad detection, which can be resource-intensive and time-consuming (Rui et al., 2000; Dabholkar et al., 2016). In the meantime, the advent of large language models (LLMs) have transformed the landscape of content understanding and curation (Salemi et al., 2023; Kirstein et al., 2024; Manatkar et al., 2024).

In this paper, we propose leveraging large language models (LLMs) to extract short, compelling and self-contained episode previews. Using only text-based inputs, including episode metadata (title and description) and transcript, we employ few-shot learning with curated examples of high-quality previews to guide LLMs in identifying the characteristics of an effective preview. To extract the preview segment, we prompt the LLMs to provide structured outputs, specifying start and end timestamps to define precise boundaries.

Our contributions are threefold. First, we successfully integrate an LLM into a large-scale, real-world application for podcast preview extraction. Secondly, we propose to use sentence indexing and sentencization to effectively analyze and index lengthy podcast transcripts and accurately retrieve LLM-selected previews. Finally, we demonstrate significant performance improvements over strong baseline expert models through offline human eval-

uations and online A/B testing, while achieving a 5x improvement in processing efficiency. By showcasing the successful productionization of this novel application of LLMs, we aim to advance the discourse on leveraging language models for complex content processing tasks, highlighting their potential to simplify workflows and enhance performance in real-world applications.

2 Previous Work

In this section, we highlight previous related work on highlight extraction, document summarization, and podcast preview extraction.

2.1 Highlight Extraction

Generating previews for podcast episodes closely parallels the tasks of highlight extraction (Sun et al., 2014; Badamdorj et al., 2021; Liu et al., 2022b; Jie et al., 2024). Highlights are typically annotated by human experts (Collins et al., 2017; Lei et al., 2021) or inferred through weakly supervised signals, such as identifying frequently edited segments in videos (Sun et al., 2014).

Given the domain dependency of labeled data and the cost of gathering them for long-form content, unsupervised approaches for highlight detection have also been explored. These include leveraging aesthetic features (Song et al., 2016) (e.g., selecting visually pleasing thumbnails), detecting recurring audio-visual patterns (Islam et al., 2024) (e.g., cheering or clapping in sports videos), or employing methods like k-means clustering (Song et al., 2016) or graph-based techniques (Erkan and Radev, 2004) to identify representative parts of the text or video. While podcast preview generation is similar to highlight extraction, it introduces an additional challenge: the previews must serve as standalone content, providing listeners with a self-contained piece of the content that can be understood on its own.

2.2 Document Summarization

Previous studies on document summarization highlight LLMs' power to identify and retrieve *key information* from lengthy documents using both extractive summarization (Zhang et al., 2023; Chhikara et al., 2025) and abstractive summarization (Tanaka, 2022; Chang et al., 2024) approaches. Building on this foundation, we utilize LLMs to extract previews, focusing specifically on extracting contiguous segments of text. However, effective

methods are essential for locating and extracting information from *long* texts, and Ghazimatin et al. 2024 illustrate successful indexing mechanisms for LLM-selected chapters, essential for accurate retrieval from long texts which inspires our work.

2.3 Traditional vs. LLM-Powered Podcast Preview Extraction

Previously, preview extraction relied on sophisticated feature-engineered systems, requiring the aggregation of one or more expert models such as sentiment analysis model and emotion recognition model (Zhu, 2021; Smith et al., 2017; Irie et al., 2010; Rehusevych and Firman, 2020). Our work has been inspired by such traditional methods to focus on human perception-related aspects like sentiment and attention in initial prompts. However, we are able to outperform these traditional approaches by utilizing LLMs, which automate and improve the extraction and retrieval process through implicit prompt iteration for a more nuanced understanding of transcript content.

Overall, by focusing on LLM-powered methodologies, our work advances beyond conventional systems, offering a more streamlined and contextually aware approach for podcast preview extraction and information retrieval.

3 System Design

In this section, we describe two systems for podcast preview generation: the sophisticated, feature-engineered legacy machine learning (ML) preview extraction system, and the newly developed LLM preview extraction system.

3.1 Language Filtering

Currently, both the legacy ML and LLM preview systems have been primarily developed on English podcast data, therefore language filtering is applied in both systems to only process English-language episodes. The legacy ML preview system employs an audio-based in-house model to perform spoken language identification similar to Zhu et al. 2023. The results are then combined with metadata language annotations (which can be noisy) to co-determine the episode language and filter for only English-language episodes. In contrast, the LLM preview system relies solely on existing noisy metadata language annotations for filtering English episodes, without needing extra language detection techniques.

3.2 Legacy ML Preview System

The legacy ML preview system (Figure 1 left) is a sophisticated system that utilizes advanced feature engineering and a series of expert models to generate podcast previews. This system involves multiple stages of data processing and signal analysis to select previews. The main components are as follows:

Topic Analysis, Sentiment Analysis and Primary Signal Aggregation The podcast episode transcript is first analyzed using an in-house model to identify key topics, which are then processed by another in-house model to assess the sentiment intensity related to each topic. This creates the so-called *primary signals*, which are aggregated to compute topic trends and identify dominant topics. Speaker boundaries and question-answer segments are also analyzed based on the transcript to ensure smooth transition of topics.

Ad Detection and Sound Event Detection: In parallel to primary signal extraction, the system uses an in-house ad detection model to identify ad content from transcript and an in-house sound event detection model to detect non-speech regions from episode audio. These create the so-called *secondary signals*, which are scaled and aggregated based on predetermined adjustment scores for different types of non-core speech elements such as ads or music.

Signal Merging and Peak Selection: By processing primary and secondary signals, the system derives overall *selectivity scores*. These scores are analyzed to locate peak regions approximately 60 seconds long, from which previews may be extracted.

Sentence Break Detection, Trimming and Ranking: An in-house technique is applied on the episode audio to identify sentence starts and ends, which are considered suitable candidates for preview starts and ends. The detected sentence breaks are then combined with the *selectivity scores* and fed into an in-house trimmer model to adjust the start and end of each preview candidate to improve coherence and context while adhering to the 1-minute duration requirement. Lastly, all preview candidates are ranked by an in-house ranking model to assign a score for each preview candidate. The candidate with the highest ranking score is used as the final preview for the episode.

This sophisticated legacy ML preview system showcases the extensive feature engineering and model integration necessary to produce high-quality podcast previews.

3.3 LLM Preview System

While we use a variety of models at Spotify, for this particular use case we use Gemini 1.5 Pro[1] in the LLM preview system (Figure 1 right) to generate podcast previews. Below are the key steps of the system:

Pre-processing: We first sentencize the podcast transcript using simple heuristics such as punctuation markers and annotate each sentence with start and end timestamps in seconds. This step is crucial for enabling the LLM to accurately identify and retrieve the desired preview offset. These timestamped sentences, along with episode metadata such as title and description, form part of the input prompt to the LLM. Appendix A provides a mock example of a pre-processed, sentencized, and timestamped transcript.

Preview Offset Selection and Preview Metadata Generation: We then apply the LLM to perform preview selection and metadata generation. The LLM prompt for preview offset selection incorporates three key elements:

1. **Structured reasoning process:** The prompt guides the LLM through a step-by-step structured reasoning process. It begins by examining the episode's title, description, and transcript to identify the main topic. The LLM then evaluates preview segments for relevance and engagement. As part of this structured reasoning, the LLM also generates preview metadata, including a concise explanation of the preview's engagement value and a list of topic tags. This structured reasoning approach not only enhances preview relevancy but also makes LLM's decision-making process more transparent and informed.

2. **Preview requirements:** A list of requirements are included in the prompt to ensure that the preview begins with an engaging introduction, progresses logically from foundational concepts to detailed insights, excludes ad content, and starts and concludes with complete thoughts, while aligning with the episode's central theme, evoking emotional resonance, and providing valuable insights. The preview is also required to be approximately one minute long, maximizing audience

---

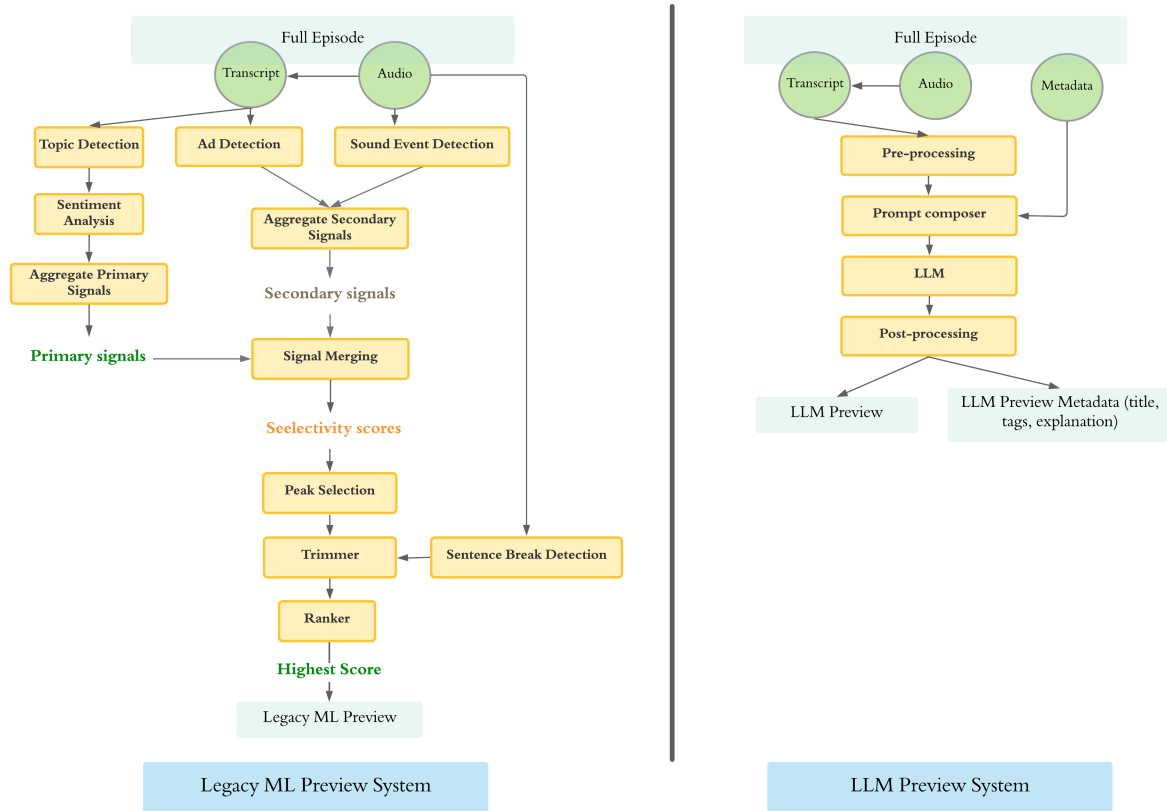[1]Gemini 1.5 Pro: https://ai.google.dev/gemini-api/docs/models/gemini#gemini-1.5-pro

Figure 1: Legacy ML preview system vs. LLM preview system.

engagement during normal attention span (Simon et al. 2023).

3. **Few-shot learning:** A series of manually curated preview examples are included in the prompt to guide the LLM in learning what constitutes a good preview through few-shot learning (Brown et al., 2020).

Prompt Iteration Process: We manually optimized the prompt to achieve strong alignment with human judgment on a small evaluation dataset of episodes from diverse categories, ensuring broad applicability and generalization across different content types. During the prompt iteration process, feedback was gathered directly from the product and design teams as it was challenging to use automated prompt engineering to replace human input in this case, particularly product and design experts. The process involved iteratively adding and deleting preview requirements and few-shot examples, with human experts re-evaluating the prompt on the small evaluation dataset after each major change to ensure improvement and alignment with human judgment. This manual prompt iteration process, despite not being automated and requiring human oversight, effectively replaced the feature

engineering process of the legacy ML system, as it allows for easier incorporation of human feedback, significantly enhancing flexibility and speed when adapting to new preview requirements.

Post-processing: To maintain a concise and coherent preview duration, the preview selected by the LLM is trimmed to the last complete sentence that starts within one minute. While both the LLM-selected preview and the legacy ML system's preview don't always guarantee a one-minute length, the average LLM preview is around 62 seconds long, and the legacy ML preview is around 56 seconds. This means their average durations are not too far apart. Additionally, the need for post-processing trimming is primarily driven by product requirements.

3.4 Comparison of LLM and Legacy ML Preview Systems

In comparison, the LLM system offers several advantages over the legacy ML system:

1. **Streamlined Iterations and Adaptations**: Prompt engineering with LLMs is significantly faster and more streamlined than manual feature engineering and expert model ag-

gregation, as LLMs allow for iterative refinement and quick adaptations to changing requirements.

2. **Lower Maintenance Complexity**: The legacy ML system involves multiple models and dependencies, making maintenance more complex. In contrast, the LLM system utilizes a single LLM framework, reducing complexity and maintenance effort.

3. **Faster Processing Speed**: The legacy ML system requires processing audio directly, which is inherently slower compared to processing text. In comparison, the LLM system works primarily with text data and benefits from faster processing times. Both systems have been deployed on the Dataflow streaming platform[2]: The legacy ML system takes an average of 100 seconds to process an episode, and the LLM system takes an average of less than 20 seconds; even though both systems are already very fast, the LLM system processes episodes faster, resulting in a 5x improvement in processing time and significantly enhancing scalability.

These improvements highlight the LLM system's advantages in terms of simplicity and scalability, making it a more streamlined solution for generating podcast previews.

4 Experiments

In this section, we describe two experiments that we have conducted to evaluate the proposed LLM previews against the legacy ML previews: an offline human evaluation and an online A/B test.

4.1 Offline Human Evaluation

We recruited around 20 evaluators internally to evaluate LLM previews against legacy ML previews, and we used Label Studio[3] platform for data collection and human annotation.

Evaluation Setup: Each evaluator was asked to evaluate around 20 episodes (the actual number of episodes per evaluator was determined based on the time they were able to commit). For each episode, the evaluator was provided with episode metadata including episode title, episode description, and show name, as well as a legacy ML preview and an LLM preview, which were randomly shuffled to

prevent position bias favoring one variant over the other. The evaluator then listened to both previews with subtitles, and was asked to choose the better one or indicate a tie.

Specific Assessment Questions: After selecting a preferred preview for a given episode, the evaluator was asked to rate both previews based on three specific questions to understand the relative performance of both systems:

1. *Understandability:* Whether the preview helps determine the episode's relevance for the listener.

2. *Contextual Clarity:* Whether the preview, along with metadata, provides sufficient context to grasp what is being discussed.

3. *Interest Level:* Whether the preview highlights an interesting segment of the episode.

In Table 1 we present the actual questionnaire that we created for each evaluator on the Label Studio platform (metadata such as episode name, audio file, and subtitles are excluded from the table for simplicity).

4.2 Online A/B Test

The online A/B test was designed to evaluate the impact of LLM previews on user engagement and content discovery compared to legacy ML previews.

A/B Test Context: In the realm of digital media consumption, enhancing user engagement by facilitating content discovery is a critical focus for many platforms. We tested our LLM previews against legacy ML previews in a product that provides an interface where users can navigate through a series of podcast previews. The primary function of previews in this product is to aid users in evaluating unfamiliar podcast content, thereby enhancing podcast discovery. Improvements to these previews are expected to enhance user evaluation experience.

A/B Test Hypothesis: We hypothesized that LLMs would select more compelling episode previews compared to the legacy ML expert models, thus enhancing the value of each preview content and increasing the likelihood that users would have a better and more effective evaluation experience with the preview.

A/B Test Setup: The online A/B test was conducted over 6 weeks across 67 English-speaking countries and was available to all users in those countries. Users were evenly split between treatment and control, with users in the treatment group receiving LLM previews in the product. In order

---

[2]Dataflow streaming pipelines: https://cloud.google.com/dataflow/docs/concepts/streaming-pipelines

[3]Label Studio: https://labelstud.io/

Questions per preview	Response
Does the preview help you decide if this episode is relevant for you?	Yes | No
Does the preview plus metadata contain enough context to understand what is being talked about from the preview?	Yes | No
Does the preview show an interesting part of the episode?	Yes | No
Question per episode	**Response**
Which preview is better?	Preview 1 | Preview 2 | A tie

Table 1: Offline human evaluation Label Studio questionnaire

to validate the usefulness of LLM previews, we generated LLM previews for a subset of recently published English episodes, resulting in LLM previews for 34% of episodes seen in the product during test period (remaining episodes used the same previews as control group). Users in the control group received legacy ML previews but never LLM previews. This test ran for 6 weeks, allowing sufficient time to gather meaningful data on user interactions. By implementing this test setup, we aimed to observe measurable differences in user engagement, specifically focusing on whether LLM previews could help with content discovery compared to legacy ML previews.

5 Results

In this section, we present the results of our offline human evaluation and online A/B test, both of which demonstrate that LLM previews outperform legacy ML previews. These findings showcase the power of LLMs in extracting more engaging and contextually rich podcast previews that improve podcast evaluation and discovery.

5.1 Offline Human Evaluation Results

We gathered 238 valid episode annotations to compare the performance of LLM previews against legacy ML previews.

Overall Comparison Results: The results indicated that LLM previews were better than or non-inferior to legacy ML previews 81.09% of the time, when considering both wins and ties, and better than legacy ML previews 54.2% of the time, when considering only wins. This implies that LLM previews were either preferred over or performed equivalently to legacy ML previews in the majority of cases (Figure 2). A binomial test[4] conducted on these results yielded a p-value of 1.37e-10, allowing us to reject the null hypothesis with confidence (at a significance level of 0.001) and conclude that LLM previews' better performance is statistically

Figure 2: Offline human evaluation: Overall comparison results between LLM previews and legacy ML previews

	Z-Test statistic	P-value	LLM previews better statistically significant?
Q1: Understandability	-4.05	5.09e-05	Yes
Q2: Contextual clarity	-3.40	0.00067	Yes
Q3: Interest level	-4.32	1.59e-05	Yes

Table 2: Offline human evaluation: Question-specific results with Proportion Z-Test

significant and not due to random variation.

Question-Specific Results: Further analysis using a Proportion Z-Test[5] on the three specific questions confirmed that LLM previews statistically significantly outperformed legacy ML previews in terms of *understandability*, *contextual clarity*, and *interest level* (Table 2).

5.2 Online A/B Test Results

The A/B test results indicate a marked improvement in user podcast discovery and evaluation with LLM previews over legacy ML previews. The following metrics were used to evaluate this impact:

Podcast Evaluation Time per User: A statistically significant improvement of 4.6% was observed in the time users spent evaluating podcast previews during their second week in the experiment, indicating that more engaging LLM previews led to enhanced user evaluation experience.

Evaluation Time per Preview: LLM previews resulted in a statistically significant 4% increase in the average time users evaluated each preview during their second week in the experiment. This improvement indicates the capability of LLMs to produce more compelling and effective preview segments, enhancing user interest and providing more value out of user evaluation period.

---

[4]Binomial test: `https://docs.scipy.org/doc/scipy-1.11.1/reference/generated/scipy.stats.binom_test.html`

[5]Proportion Z-Test: `https://www.statsmodels.org/stable/generated/statsmodels.stats.proportion.proportions_ztest.html`

6 Conclusion

In this work, we explored leveraging an LLM for podcast preview generation in a real-world application. Offline human evaluation and online A/B test results demonstrate that the LLM preview system outperforms the legacy ML system, which relies on a sophisticated aggregation of expert models. Transitioning to an LLM-based system streamlines preview generation by eliminating extensive feature engineering. This change speeds up iterations, accelerates adaptations to changing requirements, enhances scalability, and improves preview quality, highlighting the transformative power of LLMs in practical applications to simplify complex content processing tasks.

7 Ethics Statement

Our system prioritizes creator autonomy through opt-out options. Podcast creators maintain full control over their content by having the option to opt out of machine-generated podcast previews (including both LLM previews and legacy ML previews). They can also generate their own previews which will replace any machine-generated previews, ensuring that their content is represented according to their preferences.

We also prioritize accessibility and inclusivity by exploring systems that can more easily adapt to diverse contexts. For instance, unlike traditional ML systems requiring extensive retraining for each language, LLMs offer better flexibility to adapt across different languages. This adaptability allows for the creation of high-quality previews that cater to diverse audiences, enhancing user experience and expanding the reach of engaging content.

References

Taivanbat Badamdorj, Mrigank Rochan, Yang Wang, and Li Cheng. 2021. Joint visual and audio learning for video highlight detection. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8107–8117. IEEE.

Aadit Barua, Karim Benharrak, Meng Chen, Mina Huh, and Amy Pavel. 2025. Lotus: Creating short videos from long videos with abstractive and extractive summarization. *arXiv preprint arXiv:2502.07096*.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *Preprint*, arXiv:2005.14165.

Yapei Chang, Kyle Lo, Tanya Goyal, and Mohit Iyyer. 2024. Booookscore: A systematic exploration of book-length summarization in the era of LLMs. In *The Twelfth International Conference on Learning Representations*.

Garima Chhikara, Anurag Sharma, V. Gurucharan, Kripabandhu Ghosh, and Abhijnan Chakraborty. 2025. Lamsum: Amplifying voices against harassment through llm guided extractive summarization of user incident reports. *Preprint*, arXiv:2406.15809.

Edward Collins, Isabelle Augenstein, and Sebastian Riedel. 2017. A supervised approach to extractive summarisation of scientific papers. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 195–205.

Salil Dabholkar, Yuvraj Patadia, and Prajyoti Dsilva. 2016. Automatic document summarization using sentiment analysis. In *Proceedings of the International Conference on Informatics and Analytics*, pages 1–6.

Günes Erkan and Dragomir R. Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *J. Artif. Intell. Res.*, 22:457–479.

Azin Ghazimatin, Ekaterina Garmash, Gustavo Penha, Kristen Sheets, Martin Achenbach, Oguz Semerci, Remi Galvez, Marcus Tannenberg, Sahitya Mantravadi, Divya Narayanan, Ofeliya Kalaydzhyan, Douglas Cole, Ben Carterette, Ann Clifton, Paul N. Bennett, Claudia Hauff, and Mounia Lalmas. 2024. Podtile: Facilitating podcast episode browsing with auto-generated chapters. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, CIKM '24, page 4487–4495, New York, NY, USA. Association for Computing Machinery.

Go Irie, Takashi Satou, Akira Kojima, Toshihiko Yamasaki, and Kiyoharu Aizawa. 2010. Automatic trailer generation. In *Proceedings of the 18th ACM International Conference on Multimedia*, MM '10, page 839–842, New York, NY, USA. Association for Computing Machinery.

Zahidul Islam, Sujoy Paul, and Mrigank Rochan. 2024. Unsupervised video highlight detection by learning from audio and visual recurrence. *arXiv preprint arXiv:2407.13933*.

Renlong Jie, Xiaojun Meng, Xin Jiang, and Qun Liu. 2024. Unsupervised extractive summarization with learnable length control strategies. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18372–18380.

Rosie Jones, Hamed Zamani, Markus Schedl, Ching-Wei Chen, Sravana Reddy, Ann Clifton, Jussi Karlgren, Helia Hashemi, Aasish Pappu, Zahra Nazari, et al. 2021. Current challenges and future directions in podcast information access. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1554–1565.

Frederic Kirstein, Terry Ruas, Robert Kratel, and Bela Gipp. 2024. Tell me what i need to know: Exploring llm-based (personalized) abstractive multi-source meeting summarization. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 920–939.

Jie Lei, Tamara L Berg, and Mohit Bansal. 2021. Detecting moments and highlights in videos via natural language queries. *Advances in Neural Information Processing Systems*, 34:11846–11858.

Yang Liu, Chenguang Zhu, and Michael Zeng. 2022a. End-to-end segmentation-based news summarization. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 544–554.

Ye Liu, Siyuan Li, Yang Wu, Chang-Wen Chen, Ying Shan, and Xiaohu Qie. 2022b. Umt: Unified multimodal transformers for joint video moment retrieval and highlight detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3042–3051.

Michal Lukasik, Boris Dadachev, Kishore Papineni, and Gonçalo Simões. 2020. Text segmentation by cross segment attention. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4707–4716.

Abhijit Manatkar, Ashlesha Akella, Parthivi Gupta, and Krishnasuri Narayanam. 2024. Quis: Question-guided insights generation for automated exploratory data analysis. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 1523–1535.

Orest Rehusevych and Taras Firman. 2020. movie2trailer: Unsupervised trailer generation using anomaly detection.

Fabian Retkowski and Alex Waibel. 2024. From text segmentation to smart chaptering: A novel benchmark for structuring video transcriptions. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 406–419.

Yong Rui, Anoop Gupta, and Alex Acero. 2000. Automatically extracting highlights for tv baseball programs. In *Proceedings of the eighth ACM international conference on Multimedia*, pages 105–115.

Alireza Salemi, Sheshera Mysore, Michael Bendersky, and Hamed Zamani. 2023. Lamp: When large language models meet personalization. *arXiv preprint arXiv:2304.11406*.

Alexander J. Simon, Courtney L. Gallen, David A. Ziegler, Jyoti Mishra, Elysa J. Marco, Joaquin A. Anguera, and Adam Gazzaley. 2023. Quantifying attention span across the lifespan. *Frontiers in Cognition*, 2.

John R. Smith, Dhiraj Joshi, Benoit Huet, Winston Hsu, and Jozef Cota. 2017. Harnessing a.i. for augmenting creativity: Application to movie trailer creation. In *Proceedings of the 25th ACM International Conference on Multimedia*, MM '17, page 1799–1808, New York, NY, USA. Association for Computing Machinery.

Yale Song, Miriam Redi, Jordi Vallmitjana, and Alejandro Jaimes. 2016. To click or not to click: Automatic selection of beautiful thumbnails from videos. In *Proceedings of the 25th ACM international on conference on information and knowledge management*, pages 659–668.

Min Sun, Ali Farhadi, and Steve Seitz. 2014. Ranking domain-specific highlights by analyzing edited videos. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13*, pages 787–802. Springer.

Edgar Tanaka. 2022. Multilingual abstractive summarization of podcasts with longformers. Master's thesis, State University of Campinas, Institute of Computing.

Haopeng Zhang, Xiao Liu, and Jiawei Zhang. 2023. Extractive summarization via chatGPT for faithful summary generation. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Winstead Zhu, Md Iftekhar Tanveer, Yang Janet Liu, Seye Ojumu, and Rosie Jones. 2023. Lightweight and efficient spoken language identification of longform audio. In *Interspeech 2023*, pages 496–500.

Winstead Xingran Zhu. 2021. Hotspot detection for automatic podcast trailer generation. Master's thesis, Uppsala University, Department of Linguistics and Philology.

A Mock Example of Pre-Processed Transcript with Sentencization and Timestamps

Below is a mock example showcasing the format of a pre-processed transcript with sentencization and timestamps. The transcript is first divided into individual sentences, each placed on its own line. Each sentence is accompanied by square brackets indicating the start and end timestamps in seconds. This process aids the LLM in accurately selecting previews from the episode.

...

[01.00 - 02.50] Here is a mock sentence indicating the start of the transcript.

[03.00 - 05.25] This is another mock sentence serving as a placeholder.

[05.50 - 06.75] Yet another example of a mock sentence.

[07.00 - 09.00] This sentence is mock data for illustrative purposes.

[09.50 - 11.25] Final mock sentence to demonstrate the format.

...