Generating Personalized News Podcasts from Print Media for those On the Go and the Blind

Daniel Phillips and Patrick Hosein
The University of the West Indies, St. Augustine, Trinidad
daniel.phillips@my.uwi.edu, patrick.hosein@sta.uwi.edu

***Abstract*—Print media continues to be more trusted by the public than online sources. Unfortunately many people have little time to read such media and would prefer audio versions while driving or exercising and generally while busy with other activities. This also holds true for the visually impaired. We introduce an automated, personalized solution for emerging markets. This system follows the current shift to digital media which allows for more personalized solutions. Many can benefit from these implementations as they seek alternative revenue sources due to the decline in the revenue sources from print media. This system leverages freely available locally hosted Large Language Models (LLMS) and Text-To-Speech (TTS) synthesis tools to provide an accessible and sustainable alternative to traditional media. We compare two methods, one using Google's cloud-based NotebookLM for summarization and personalization and another entirely locally hosted solution using the LLaMA model and a quantized version of the Orpheus Text-To-Speech model. Our implementation allows for both GPU-Based solutions as well as CPU-only solutions thus reducing the need for external cloud-based systems. We show that the local pipeline also provides comparable performance to external proprietary systems. It produces a personalized experience for users while maintaining a similar level of clarity and relevance to the proprietary system. A dual-layer evaluation approach was conducted to evaluate the technical performance of the model while also evaluating the potential to monetize an Artificial Intelligence (AI) news podcasting solution. The results showed a demand for an ad-free, subscription-based model, allowing users to have more variety in how they consume news. The proposed solution underscores the transformative potential of open AI tools and technology when used in the media space.**

I. Introduction

In many developing countries, traditional media houses are currently experiencing a clear decline in profitability. For example, in the Caribbean countries such as Barbados, Jamaica, TT and Guyana are seeing profits decline by 54%, 40%, and 26.5% respectively [19]. This decline is due to increased competition with online pay-per-click advertising [19] along with changes in advertising spend by key organizations [20]. There has also been a decline in readership among the general public in developing counties such as Trinidad and Tobago and developed countries such as the United States [6] [7]. Around the world, many media houses are currently experiencing a decreased interest in print media as viewership trends shift more toward modern digital media. At present, users are more inclined to browse the summaries of articles online or consume content on social media rather than purchase newspapers and engage with traditional print media. According to a recent survey, only about 9% of Americans rely on print media for their news compared to 56% that preferred to get their news digitally [6].

Across the globe, there has been an increased adoption of podcasting as a medium for content consumption. Podcast adoption has grown as a popular form of content due to its ease of consumption and strong appeal to digital media audiences. Due to their nature, podcasts offer more convenience and flexibility when compared to other media forms as they can be consumed while multitasking or performing daily tasks such as exercise or chores. Unlike other media forms, it can also be consumed while driving and even be paused and continued at a later time [15]. There is also the opportunity for it to be uniquely customized to the users' preferences as can be delivered individually to the consumer. It is predicted that in 2025 there will be an estimated 584 million podcast listeners worldwide [5]. Also, it is estimated that 4.46 billion dollars would be spent on podcast advertising worldwide. This is compared to the 3.46 billion spent in 2023 [9]. Current trends indicate that podcast-related spending is projected to exceed 5 billion in 2027. Given the Caribbean's tendency to align with global trends, it is expected that podcast adoption and spending in the Caribbean will increase significantly in the near future. During the COVID-19 pandemic, Caribbean-based podcasts experienced a notable increase in viewership, which demonstrated increased interest in this form of media [21]. It is also expected that this trend will allow for podcast advertising to emerge as a viable avenue for monetization.

Visual impairment currently affects approximately 2.2 billion people worldwide. With the global population aging, its prevalence is expected to only increase within the coming years [8]. Visual impairments often make it difficult for individuals to stay informed by limiting their ability to access information such as the news. Social activities such as keeping up with current events have been shown to boost cognitive function, especially in those who are elderly [17]. Keeping up to date with current events through traditional media presents challenges, even for older individuals without visual impairments. Many of the print media outlets are now leaning towards digital mediums which require a subscription along with a basic level of technological proficiency. As podcasting continues to grow as a media format, we believe that its integration with traditional newspapers holds the potential to significantly enhance news accessibility, particularly for elderly individuals and those with

visual impairments.

Research by Rajput et al further shows how podcasting can be an effective advertising medium. Their research showed how podcasting allows the presenters and viewers to foster genuine connections which improves the success rate of this type of advertising. Podcasting can also facilitate interactions between the hosts and the audience fostering greater trust and a stronger feeling of authenticity. This leads to the audience being more confident in the goods or services promoted by the podcast host. Influencer advertising via podcasts can be an effective method when combined with a consistent presence. Content trustworthiness was a major factor in determining the success of the advertising strategy [16]. Traditional media houses now have a great opportunity to monetize through podcasts given the public's reliance on the news and the high level of trust placed in its content. Surveys have shown that the contents of the news usually falls in the category of "very trustworthy" or "somewhat trustworthy" [19] .

A. Problem Statement

Currently in developing countries newspapers are primarily delivered as a physical copy or digitally in a text format. This limits accessibility for both the elderly and visually impaired community while limiting the profitability for the traditional media houses. An alternative would be to generate personalized podcasts based on the subscriber preferences which can be delivered directly using email or internet-based messaging. Currently, tools like NotebookLM by Google can generate a podcast from any uploaded PDF document. This however can produce an output that can be highly generalized and require a level of technological proficiency to upload files and craft the prompts required to generate the desired output. These tools also typically require a premium subscription to generate multiple podcasts per day. This method also raises privacy concerns for sensitive data as the files must uploaded to the cloud for processing due to the lack of open source methods to achieve similar results locally.

B. Objectives and Contributions

In this paper we propose a method for generating news podcasts locally using a custom pipeline. This local method is similar to NotebookLM but it only uses locally hosted LLMs and TTS models to produce the output. These models will be used to produce a personalized podcast based on the user's news article preferences. These personalized daily podcast can then be monetized providing an alternative revenue source for local newspapers. The performance of our method will also be compared to the performance of NotebookLM which is widely used.

II. Related Literature

A. News Summarization and LLMs

There are many options observed in literature to provide summarizations of text. Most methods can be categorized as either extractive or abstractive [14] . The extractive approach first identifies key phrases that are then extracted by identifying the most crucial information. These phrases are concatenated to produce the final summary. The other method, the abstractive technique, generates a concise summary of the text using new novel sentences and phrases based on an understanding of the original content. Kumaran et al explored using extractive NLP techniques to produce text summaries. In their paper, the NLTK (Natural Language Toolkit) is used to tokenize the text analyzing the structure and phrases of the text. This method was chosen as it was seen as more efficient than using deep learning techniques which require more resources. The author also stated that these NLP models tend to be more accurate and context-aware [14]. Mathialagan et al explored the use of a combination of the summarization techniques. In their research, a model was produced by combining the extractive and abstractive methods. This method performed better than the individual models alone for the news summarization tasks. The authors used the GRUEN scores to evaluate the model performance when summarizing the newspaper articles. The models ran in a Google Colab notebook which had 12 GB of RAM and 100GB of disks. This implementation used the BERT, XLNet, and GPT-2 models for the text summarizations.

One of the challenges of Large Language Models (LLM) is that they can sometimes lack accuracy when generating summaries. They have been known to hallucinate adding inaccuracies into the final summary. The inclusion of made-up information can cause misinterpretation and be confusing for users [16]. This is typically caused by training the model on outdated data or data that lacks proper context. The quality of the input and training data is important to prevent this from occurring [18].

B. AI-Generated Podcasts

The creation and advancement of Text to Speech (TTS) and Large Language Models (LLMs) have allowed for information to be digested in new ways. The prevalence of these automated systems has allowed for the generation of podcasts which would allow a user to interact with information in an auditory way [13] . Approaches such as Kadam et al's method, combined TTS, LLMs, and Retrieval-Augmented Generation (RAG) to effectively generate podcasts based on text content. This can also make the content more interesting and engaging when compared with the use of a traditional TTS model alone. Arya et al developed a method for creating podcasts based on local models. For their implementation, the prebuilt model library Ollama was used to host the Llama3.2:3b model. The bark TTS model was chosen to handle the audio generation based on the model's output. A RAG-based chatbot was also developed to allow users to ask specific questions related to the content provided [13].

Another implementation by Tiffany et al explored the benefits of custom-tailored podcasts to aid with the students learning. A 180-student study was conducted to compare the effectiveness of reading compared to converting textbooks into generalized and personalized podcasts. Based on the findings,

the AI-generated podcasts were more enjoyable than reading alone. The podcasts were generated to be tailored to the learning styles of each student involved in the study. For the generation of the tailored podcast scripts, Google's Gemini pro model was used. The Gemini model was able to generate personalized podcasts, that incorporated the students interests where possible. For example, the podcast script incorporated examples referencing the specific student's hobbies and preferences to maintain their interest. The students were surveyed using a Likert Scale questionnaire to gather feedback on the podcasts. The survey results showed favorable responses regarding the use of AI-generated podcasts. Their research demonstrated that AI-generated podcasts can be a great supplement to traditional textbooks [11].

In the study conducted by José Manuel Martins Ferreira, Google's NotebookLM was used to simplify the complex content of a university-level Computer Science course [12]. The author and students were impressed by its ability to explain complex concepts simply for the students to understand. During this research, NotebookLM took the role of both teaching and grading courses. The teaching aspect was seen as more favorable by the students as they were able to better understand the material when compared to reading the lecture notes alone. It was noted, however, that sometimes the podcasts contain erroneous content. This limitation was mitigated by supplementing NotebookLM with information from multiple sources [12].

C. TTS Systems.

There are several TTS systems that have recently gained popularity such as Coqui TTS [2], Bark TTS [4], Piper TTS [3], Hola-TTS [10], and Orpheus-TTS [1]. Many of these models support multiple languages and can be used for a wide array of purposes. These advanced models can replicate human speech patterns and rhythm producing natural-sounding speech. The resource requirements of these models are typically small enough to allow for local deployment while producing good results. Models such Bark are rapidly improving quality making them suitable to produce high-quality podcasts [13]. Some of these models can also incorporate emphasis and non-speech elements, such as laughter, resulting in a more realistic and engaging experience. To improve the quality of the output, a preprocessing stage was added to remove any artifacts, such as parenthesis and incompatible punctuation before TTS processing [11].

III. Methodology

A. System Overview

A prototype was first developed using NotebookLM as a benchmark to compare the performance of our approach. We then implemented our custom pipeline to generate podcasts using only locally hosted models. NotebookLM is a proprietary, closed-source model that operates on custom GPUs within Google's cloud infrastructure, as it requires substantial computational resources. However, this provides us with a useful baseline for what users will expect from an automated podcast implementation.

1) Prototype: For both implementations, volunteers were asked to select their preferences for the news articles from the following categories: NEWS, OPINION, WORLD NEWS, BUSINESS AND STOCKS, TOBAGO DIGEST, WHAT'S ON, CLASSIFIEDS, DEATH NOTICES, and SPORT. A notebook was then created on NotebookLM using a digital PDF copy of the newspaper as a source. A prompt was then used to specify which topics should be covered based on the selected preferences. The audio overview "deep dive conversation" feature was then used to generate the podcast. The audio of the generated podcast was then distributed to the volunteers to review.

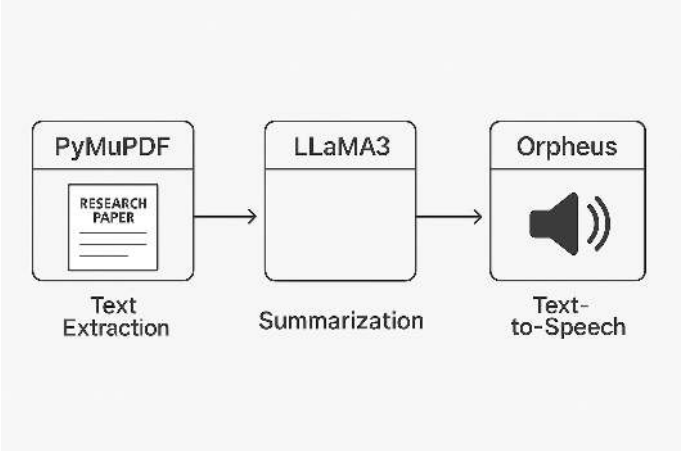

Fig. 1. Local Implementation Pipeline

B. Local System

Figure 1 shows the pipeline of the local approach.

1) Script Generation: For our implementation the PDF digital copy of the newspapers was parsed and the text was extracted using the PyMuPDF library. The newspaper was then split into sections based on the section headings to ensure that only the sections of interest were processed. For each section, the extracted text was split into 5000 character chunks to prevent the lengthy text from exceeding the context window of the model, reducing it's performance. Each chunk was then processed separately and appended together. This processing included the summarization and podcast script generation. Once all of the chunks were processed, the entire script for the section was processed as a unit to ensure that it was cohesive for that particular section. Once the scripts for the sections were completed they were appended together producing the final script. To improve the script flow, intros, and outros were added between the sections. At the beginning and end of the final script, the main intro and outro were appended creating the final script. The llama3.2 3B model hosted locally via Ollama was used for script generation.

2) Audio Generation: For the audio generation the Orpheus model was chosen. This model was chosen because it had the best consistency and tone suitable for the podcast format. The other models had a more robotic-sounding tone, which would not be suitable with a high-quality podcast. The Orpheus quantized 3b GGUF model was used via LM Studio as this was the smallest version of the model available. Firstly, the NLTK sentence tokenizer was used to split the text into sentences. This was done to improve the TTS performance as long sections of text often produced poor results. Each sentence was processed separately and stitched together using the pydub library.

3) Podcast Audio File Delivery: The podcast audio files were delivered via WhatsApp and Telegram to the volunteers. The generated podcast was then shared with the volunteers. This process was repeated daily for one week to ensure that participants were exposed to multiple podcast episodes.

C. Podcast Evaluation

The model performance was evaluated by the volunteers and the responses were captured in two surveys. The volunteers were asked to first compare the perceived quality and usability of the podcasts produced by the local and cloud-based instances. Another survey was used to also evaluate their willingness to listen to and subscribe to AI-generated news podcasts. In the surveys they were asked to give feedback on the performance of the podcasts and if they would like anything additional to be included to make it more appealing. Both surveys employed questions where the users rated responses on a Likert scale alongside open-ended questions for more detailed responses.

IV. Results

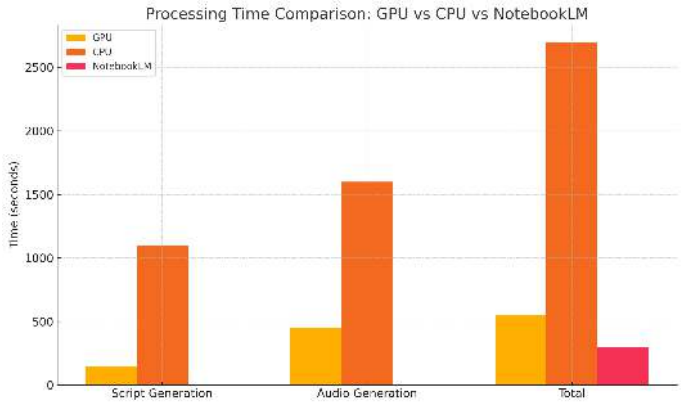

Fig. 2. Processing Time Comparison

A. Processing Efficiency

Figure 2 shows the results obtained when producing the podcasts. For evaluation of the local pipeline, the processing of the script was split into two parts, the generation of the script and the generation of the audio file. On the GPU Instance, the article extraction and script generation took on average about 150 seconds while the generation of the audio took about 450 seconds. So combined, it took on average about 550 seconds (9.17 minutes) to generate the final output. The script generation consumed 5.3GB of GPU VRAM and between 81% to 100% of GPU utilization. The audio generation consumed 4.6GB of VRAM and used between 69% to 90% of GPU utilization. The hardware used for testing was a midrange consumer grade 4060 NVIDIA GPU with a total of 8GB of VRAM. The CPU-only implementation took around 1100 seconds for the article extraction and generation of the script. The CPU-only audio generation took about 1600 seconds. Combined, CPU only implementation took 2700 seconds (45 minutes) to generate the final podcast.

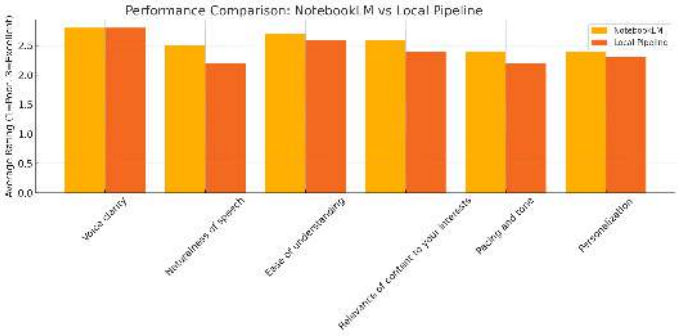

Fig. 3. Processing Time Comparison

B. Podcast User Evaluation

The first survey completed by the volunteers assessed the podcast host's voice clarity, naturalness of speech, ease of understanding, relevance, pacing, tone, and personalization. Figure 3 shows the performance comparison of the approaches.

1) Voice and Clarity and Naturalness: Around 80% of respondents scored both NotebookLM and the local model as excellent with voice clarity. This suggests that both instances handled the majority of words effectively. This was impressive as the local instance does not have access to Google's advanced voice synthesis technology and infrastructure. Respondents who did not rate the model's performance as excellent cited word mispronunciations, particularly of regional terms, as the main reason for their rating. The naturalness of speech was more of a challenge for the local instance as the NotebookLM instance received a slightly more favorable score in this regard. The NotebookLM instance was seen as more expressive while the local instance based on Orpheus TTS was seen as more monotone and robotic. These findings are consistent with other papers such as [10] where local TTS models can exhibit poor timbre and naturalness. These results indicate that the locally implemented model is successful but less able to handle some of the nuances encountered in the scripts generated based on the news articles.

2) Comprehension and Ease of Understanding: Both systems were rated highly for ease of understanding as the majority of users rated this section as excellent. This indicates that users

perceived the content itself to be clear, well-structured, and easily digestible in both versions. This would have demonstrated that the Gemini model used by NotebookLM version and the Llama 3.2 model used by the local version were able to produce high-quality scripts even from a local unstructured newspaper text. One of the negative responses to the local model came from deficiencies when processing the opinion section of the newspapers. This section was particularly challenging, as understanding the full context often required familiarity with current events. Additionally, in one instance, an opinion piece included references to religion that were not directly related to the news content, causing confusion among some users due to a lack of contextual information in the script.

3) Relevance and Personalization of Content: Both approaches received comparable results for the personalization of the content. This near parity was notable as they employed different approaches to filter the content to the user's preferences. NotebookLM used an approach where the content was filtered based on the content of the articles while the local version employed a page filtering technique to only process information on the pages containing the topics of interest. This showed that the page filtering approach was a success as it yielded similar results while being less resource-intensive. The NotebookLM was viewed as clearer while the local version felt more tailor-made by some users especially how it made better references to specific local events extracted from the articles. This is most likely due to the filtering producing a script closer to what is expected when browsing the physical copy of the newspaper.

4) Pacing and Tone: Feedback indicated that while NotebookLM's voice had more polished pacing, the local version was slightly easier to understand—particularly when presenting complex news topics. There was a split where 50% of people preferred the NotebookLM version with 40% preferring the local version and 10% having no preference. The ones who preferred the NotebookLM version cited that it has more "logical flow" and "natural speech" while the respondents who preferred the local version said that it was "easier to digest" had "better relevance" and it "sounded more appropriate for the region". These results were noteworthy as the processing power of cloud systems is not comparable to local systems. However, these results indicate that with optimization and refinement, they can deliver a similar level of user satisfaction.

5) Broader Listener Insights: The second survey captured the listener's insights including the listener's expectations, monetization preferences, and accessibility considerations for AI-generated podcasts.

A large proportion of the respondents rated access to up-to-date audio news as highly valuable. It received an average score of 4.3/5, which confirms that up-to-date news audio is important to users and provides a compelling case for the use of podcasts as a medium to deliver the news. Respondents indicated that they valued an ad-free listening experience as this was also highly rated with over 90% of users giving it a score of 4 or 5 on the Likert scale. This was also seen in literature where listener retention improves with reduced advertising interruptions [9, 13]. Interestingly, the preferred monetization method by the respondents was a subscription-based model. Over 70% preferred the subscription model over the pay-per-episode model. About 54% of the respondents indicated that they would pay between $7 and $30 USD per month for a news podcast. This suggests that consumers prefer paying for access over encountering advertisements. This also validates that this approach can provide a reoccurring revenue stream for the media outlets. Lastly, users also indicated that they would be interested in a video podcast version. This shows that the users are interested in an expansion to a multimodal news delivery medium where headlines can be combined with audio and video to increase engagement.

V. Discussion

Based on the results there is a clear opportunity to transform the delivery of news in developing countries through the use of AI-based podcast automation. The implementation of the NotebookLM version provided a valuable baseline for the LLM-based approach for podcast generation. The newspapers presented a unique challenge for these LLMs approaches as the formatting is unstructured and ambiguous as it is designed to be read by a human, and not an AI model. With prompting the NotebookLM model was able to ingest the entire newspaper and produce a podcast based on the user's interest. Users appreciated the engaging content created and the ability to have a personalized experience.

NotebookLM however had several limitations with the automated generation of these systems. The first is that the system is entirely cloud-based, relying on proprietary models and infrastructure. There also isn't an API readily available for the automation of podcast generation. The closed source models also would have a user "locked-in" raising cost and privacy concerns. Also, there is less control to tailor or optimize the output to a specific use case which makes it unpredictable and leads to inconsistencies at times.

Our local pipeline provided a method for reproducing a similar experience to NotebookLM using locally hosted open-source components. The generation time of the local model was comparable to the NotebookLM model and so was the quality of the audio produced. The limitations of the local implementation include the need for additional configuration, as the newspaper content must be filtered by user-selected sections and is restricted to a single voice option. Despite the different approaches, the quality of the podcast produced was comparable to the NotebookLM implementation.

The local implementation can be delivered directly to the user without users needing to be computer literate and understand how to prompt the model to get their personalized results. While NotebookLM can export generated podcasts our solution is automated and can be easily adapted to automatically deliver podcasts directly to users on WhatsApp, Email, or Telegram which would provide easier access for the older and visually impaired users (50+).

Even though in some aspects the local implementation achieved slightly lower scores it shows that currently a locally hosted solution can approach the performance of the proprietary platforms while offering additional privacy and cost savings. These findings validate the research goal to design a fully open-source, low-cost solution that supports the end users. By combining the LLM summarization and script generation with the local TTS models this project has laid the foundation for ethical and equitable automated podcast generation.

VI. Conclusion

This study introduced and evaluated a locally hosted pipeline for transforming digital newspapers in developing countries into personalized audio podcasts. This approach leveraged large language models (LLMs) and text-to-speech (TTS) synthesis to automate the process based on user preferences. This research demonstrates that this can be accomplished using open source tools that require only modest computational resources, while still retaining several features found in commercial solutions such as Google's NotebookLM. Our solution offered key advantages such as GPU and CPU-only versions which translates to reduced cost, and full control of the data used. Participants rated this approach comparably to the cloud-based method, reinforcing its usefulness in this domain. This automated solution is expected to benefit from future advancements in AI allowing for improved content delivery and consistency. This approach also presents a potential revenue stream for traditional news outlets facing declining income, while simultaneously enhancing accessibility for those on the go, older adults and individuals with visual impairments.

VII. Future work

Many advancements can be made to improve the performance of the local approach. One would be to add multilingual support based on the local context. Based on user feedback, a valuable improvement would be to incorporate accurate pronunciations of local slangs, along with a more regionally appropriate accent for the presenter.

Another proposed addition would also be a feedback loop where users can rate and provide real-time feedback on the podcast. This will allow customers to give an in-depth review of the relevance and composition of the news report allowing for more refined personalized content based on the customer's evolving preferences.

In the future, we would seek to have access to the articles potentially through a licensing agreement without the need to parse the complete newspaper. This would allow for increased accuracy and consistency with what is generated. Also, we can incorporate the articles from multiple sources giving a more balanced view on the content. This would allow for it to become a more comprehensive news aggregation tool that can potentially remove some bias from the reporting.

References

[1] GitHub - canopyai/Orpheus-TTS: Towards Human-Sounding Speech — github.com. https://github.com/canopyai/Orpheus-TTS. [Accessed 30-06-2025].

[2] GitHub - coqui-ai/TTS: - a deep learning toolkit for Text-to-Speech, battle-tested in research and production — github.com. https://github.com/coqui-ai/TTS. [Accessed 30-06-2025].

[3] GitHub - rhasspy/piper: A fast, local neural text to speech system — github.com. https://github.com/rhasspy/piper. [Accessed 30-06-2025].

[4] GitHub - suno-ai/bark: Text-Prompted Generative Audio Model — github.com. https://github.com/suno-ai/bark. [Accessed 30-06-2025].

[5] Podcast Statistics You Need To Know in 2025 — backlinko.com. https://backlinko.com/podcast-stats. [Accessed 30-06-2025].

[6] The people in print — trinidadexpress.com. https://trinidadexpress.com/opinion/columnists/the-people-in-print/article_3415e0a0-253d-11ef-b2ce-3370a2ba4b85.html. [Accessed 30-06-2025].

[7] Ashleigh Barber. Reasons for the Decline of Local Newspapers — Close Up — closeup.org. https://www.closeup.org/the-decline-of-local-newspapers/. [Accessed 30-06-2025].

[8] Laura N. Cushley, Benedict Leonard-Hawkhead, Andrew Jonathan Jackson, and Tunde Peto. Global certification of visual impairment registries: A scoping review. *Acta Ophthalmologica*, 103(1):7–15, September 2024.

[9] Maitrayee Dey. Podcast Advertising Statistics By Market Revenue And Country-Wise — electroiq.com. https://electroiq.com/stats/podcast-advertising-statistics/. [Accessed 30-06-2025].

[10] Hongwu Ding, Yiquan Zhou, Wenyu Wang, JiaCheng Xu, and Jiaqi Mei. Hola-tts: A cross-lingual zero-shot text-to-speech system for chinese, english, japanese, and korean. In *2024 IEEE 14th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, page 601–605. IEEE, November 2024.

[11] Tiffany D. Do, Usama Bin Shafqat, Elsie Ling, and Nikhil Sarda. Paige: Examining learning outcomes and experiences with personalized ai-generated educational podcasts, 2024.

[12] José Manuel Martins Ferreira. Ai-assisted assessment: A dual perspective on effective usage plans for students and teachers. In *2025 IEEE Global Engineering Education Conference (EDUCON)*, page 1–10. IEEE, April 2025.

[13] Arya Jalindar Kadam, Prasad Rajaram Kute, Chinmay Ashok Kale, Kushal Bhoraji Pathave, Chaitali Shewale, and Pawan Wawage. Enhancing comprehension with llm, tts, and rag: Transcription of text into podcasts and chatbots. In *2025 International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE)*, page 1–6. IEEE, January 2025.

[14] B. Logesh Kumaran and N. V. Ravindran. Nlp based text summarization techniques for news articles. In *2024 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI)*, page 1–6. IEEE, March 2024.

[15] Laura Minooka. The rise of podcasting: Evolution, impact, and future directions, 2024.

[16] Aditi Rajput and Aradhana Gandhi. Behind the mic: Exploring the impact of influencer podcasts on consumer perception. In *2024 International Conference on Automation and Computation (AUTOCOM)*, page 593–599. IEEE, March 2024.

[17] Nur Riviati and Bima Indra. Reading holy quran associated with better cognitive function in older adults: A systematic review. *Gerontology and Geriatric Medicine*, 10, January 2024.

[18] Bediga Sharan, Umar Mubarak Mulani, R. Naveenkumar, Sujit Kumar Badodia, Hemant Makawana, and Joshuva Arockia Dhanraj. Natural language processing for generating synthetic news detection. In *2024 7th International Conference on Contemporary Computing and Informatics (IC3I)*, page 431–435. IEEE, September 2024.

[19] Janelle De Souza. Changing fortunes of media in TT - Trinidad and Tobago Newsday — newsday.co.tt. https://newsday.co.tt/2020/09/27/changing-fortunes-of-media-in-tt/. [Accessed 29-06-2025].

[20] Janelle De Souza. Government's cost-cutting extends to media advertising - Trinidad and Tobago Newsday — newsday.co.tt. https://newsday.co.tt/2025/05/13/governments-cost-cutting-extends-to-media-advertising/. [Accessed 30-06-2025].

[21] Rebecca Walcott. How podcasts are helping Caribbean people during the pandemic — podnews.net. https://podnews.net/article/caribbean-podcasting. [Accessed 30-06-2025].