Ollscoil Teicneolaíochta an Atlantaigh

Atlantic Technological University

AI Audiobook Creator

Patrick Duffy

4th Year Computing in Software Development

https://github.com/PatrickDuffy1/Y4_Final_Year_Project

Introduction

Project Overview

This project focuses on the development of an AI-powered system for generating single and multi-speaker audiobooks entirely offline. It features both command-line and user interface options, ensuring ease of use, privacy, and affordability.

Innovation

While AI-generated single-speaker audiobooks exist (to a limited extent), this project introduces a novel, automated multi-speaker system. It uses text analysis to identify characters, allows unique synthetic voices to be assigned to each, and produces high-quality, immersive audiobooks without the need for human voice actors.

Motivation

Traditional audiobooks are expensive to produce (and buy), especially multi-speaker versions, which are very rare and costly. This tool allows users to convert ebooks into audiobooks at no cost, increasing accessibility, for traditional book readers, audiobook listeners, and particularly for visually impaired users, making immersive audio storytelling more widely available.

Original Contribution

At the project's start, no comprehensive open-source or closed-source solution existed for multi-speaker audiobook generation. All development and design choices are original, informed by research and built from the ground up.

Objectives

Single-Speaker Audiobooks

Automatically generate audiobooks with a single synthetic narrator from any book or text file.

Multi-Speaker Audiobooks

Detect and assign unique voices to different characters for an immersive, expressive listening experience.

Expressive Tone Rendering

Ensure that generated voices reflect emotional tone and natural variation, enhancing realism.

Custom Voice Support

Allow users to add new voices (such as personal or cloned voices) to expand the voice pool.

Full Offline Functionality

Ensure all features run locally without internet access, while also supporting optional remote/cloud execution.

Technologies

Text-to-Speech (TTS)

Generates expressive, natural-sounding voice output from text.

- XTTSv2 – Multilingual, zero-shot TTS with real-time performance and voice cloning.
- Coqui.ai TTS library – An inference and zero-shot voice cloning model by the same creators as XTTSv2.

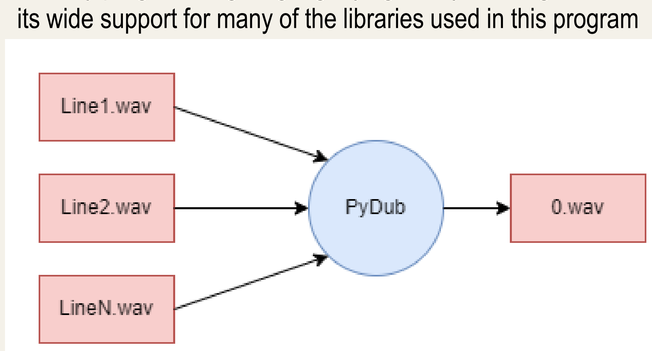
Large Language Models (LLMs)

Used for character identification, dialogue structuring, and contextual text understanding.

- Gemma 3 27B – Smaller LLM optimized for consumer GPUs.
- LLaMA 3.3 70B – Larger LLM, older than Gemma 3.
- llama.cpp & GGUF – Enables efficient local inference of quantized LLMs on CPUs/GPUs.

Python

Primary programming language tying the system together. Chosen for its wide support for many of the libraries used in this program



JSON

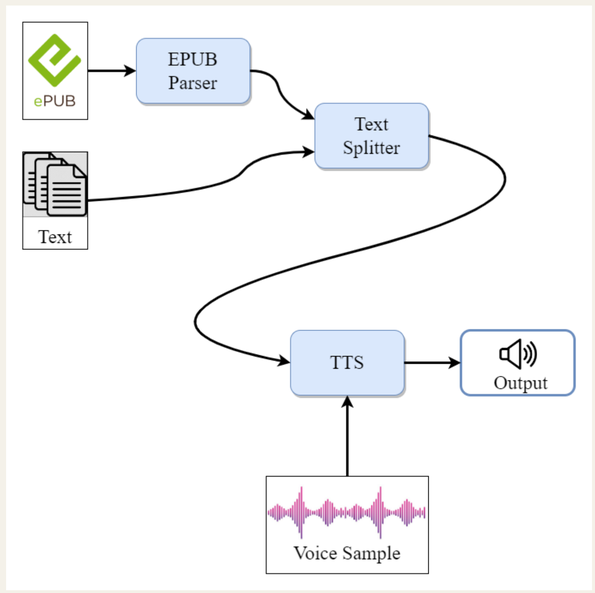
Manages dialogue structure, character line attribution, speaker metadata, and generation outputs.

System Design

This system generates both single-speaker and multi-speaker audiobooks using AI-driven text-to-speech (TTS) and large language models (LLMs), with a focus on modularity, offline functionality, and voice customization.

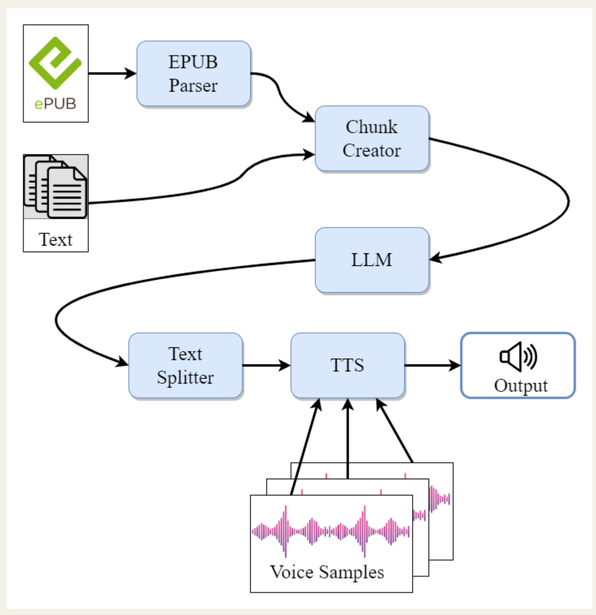
Single-Speaker Audiobook Generation

- Supports plain text and EPUB inputs.
- Uses XTTSv2 for high-quality, cloned voice synthesis.
- Handles preprocessing, chunking, and punctuation cleanup for smooth speech generation.
- Output is structured chapter-wise for easier editing and navigation



Multi-Speaker Audiobook Generation

- Utilizes an LLM (Gemma 3 or LLaMA 3.3 used during testing) to identify who says what.
- Each character can be assigned a unique voice by the user.
- Processes text in smart-sized chunks to avoid memory and coherence issues.
- Outputs structured JSON mappings of dialogue to characters.



Voice Cloning and Assignment

- Users can provide voice samples for custom cloning.
- Flexible character-to-voice mapping supports aliases and eliminates the need to reuse voices.
- Audio is generated for each dialogue line, then stitched into chapters.

Architecture and Extensibility

- Core logic is wrapped in a modular Session class for easy reuse.
- Decoupled from UI, supports CLI, GUI, and future automation tools.
- Designed to work with any LLM or TTS model compatible with existing interfaces.

Output Structure

- Output: WAV - one file per chapter or full book (advised to convert to MP3 or another compressed format).
- Fully compatible with common audiobook players.
- JSON files preserve all character and line mappings for reproducibility or editing.

Acknowledgements

I would like to thank my supervisor Gerard Harrison for his guidance throughout the project.

System Evaluation

TTS Output Quality

- Synthesized voices sound natural and unique.
- Minor issues with American accent bias.
- High-quality voice clones depend on clean, expressive source samples.

LLM Output Quality :

- Strong performance in attributing character lines.
- Some misclassifications due to small input chunks or edge-case texts.
- Better performance at higher temperatures—contrary to initial expectations.

Model	Size	Hardware	Processing Time
Gemma 3 27B (Q4)	16 GB	Desktop (24GB VRAM)	3 hours
Llama 3.3 70B (Q4)	41.5 GB	Desktop (24GB VRAM)	24 hours

Performance Comparison of the two main LLM Models used in testing for Character Line Identification on the book Harry Potter and the Philosophers Stone

Performance Highlights

TTS Performance (XTTSv2):

- High-quality, lifelike speech output.
- VRAM-efficient but slow on CPU; potential memory leaks noted.
- Chunked input gives linear time complexity: O(n).

LLM Performance (Gemma 3 27B & Llama 3.3 70B):

- Accurate dialogue attribution in most cases.
- Llama 70B more robust in edge cases, but 8x slower than Gemma.
- Requires substantial RAM/VRAM for full efficiency.

Limitations & Challenges

- High hardware requirements for large models (e.g., 27B+ LLMs).
- Output variability across different books; edge cases still cause attribution issues.
- Voice cloning is plausible but not perfect in tone/accent accuracy.
- Prompt and chunk optimization remain open areas for refinement.

Conclusion

This project successfully developed a fully offline, AI-powered audiobook generator for both single and multi-speaker formats. Key goals like local execution, voice flexibility, and natural expressiveness were met through modular design and modern TTS and LLM integration.

Key Achievements

- ***Single & Multi-Speaker Generation***: Natural-sounding, expressive audiobooks created from text.
- ***Character Voice Assignment:*** Automated detection and unique synthetic voice assignment for each character.
- ***Offline First***: Fully functional without internet; cloud support optional.
- ***Custom Voice Support:*** Users can clone and add new voices.
- ***Future-Proof Design***: Plug-and-play model loading enables seamless upgrades.

Notable Discoveries

- Smaller models (e.g., Gemma 3 27B) performed better than expected.
- TTS expressiveness emerged even without explicit tone modeling.
- Higher temperature values improved speaker attribution.
- Rapid model evolution validated the need for modular architecture.

Future Directions

- Enhance performance with faster model backends and GPU optimization.
- Improve attribution with smarter prompt strategies and chunking methods.
- Integrate tone control and advanced TTS models like Dia-1.6B.
- Develop automatic voice-character matching via voice classifiers.
- Use new LLMs to improve character line identification.

Impact

This system makes audiobook creation more affordable, accessible, and customizable especially for visually impaired users and underserved genres. By eliminating the cost barrier and supporting multi-speaker narration, it offers a scalable, open alternative to expensive, traditionally produced audiobooks.