Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X
IMPACT FACTOR: 7.056

IJCSMC, Vol. 14, Issue. 9, September 2025, pg.15 – 20

TonePick: An AI-Based Personalized Audiobook Application Using Voice Cloning and Real-Time TTS

Seeun Kim[1]; Jinyoung Kim[2]; Junwoo Park[3]; Sungsu Ahn[4]; Seungjae Lee[5*]

[1,2,3,4,5]Computer Engineering Department, Sunmoon University, Korea
[1] kimseeun6442@naver.com; [2] kimjin11444@gmail.com; [3] wnsdn0006@gmail.com; [4] tjdtn9824@naver.com;
[5*](corresponding author) leeko@sunmoon.ac.kr

DOI: https://doi.org/10.47760/ijcsmc.2025.v14i09.003

Abstract: This paper presents TonePick, a mobile application that enables the generation of personalized audiobooks by leveraging voice cloning and real-time Text-to-Speech (TTS) synthesis. The system allows users to create a custom AI voice model using short speech samples and applies it to read PDF documents with emotional expressiveness. Built with React Native, Firebase, and the ElevenLabs API, the architecture supports scalable deployment and seamless client-server interaction. A user study involving 27 participants demonstrated high levels of satisfaction across usability, naturalness, and emotional engagement, confirming the system's effectiveness and applicability. The proposed approach highlights the potential of voice cloning as a socially impactful extension of TTS technology.

Keywords: Voice Cloning, Text-to-Speech, Personalized Audiobook, Mobile Application, ElevenLabs API

I. INTRODUCTION

Recent advancements in Text-to-Speech (TTS) technology have enabled its widespread application across various domains, including smartphone navigation, audiobooks, and AI-powered voice assistants. In particular, end-to-end deep learning-based TTS systems—comprising Text2Mel modules for spectrogram generation and vocoder architectures for waveform synthesis—have significantly improved the intelligibility and naturalness of synthetic speech [6], with quality approaching that of human speech.

End-to-end speech synthesis models such as Tacotron [1] have significantly improved the quality of synthetic speech by learning a direct mapping from text to spectrogram, eliminating the need for manual feature engineering.

Nevertheless, conventional TTS systems are typically trained on standardized datasets using a limited number of speakers, resulting in monotonic, read-style output with limited expressive capability. Such limitations reduce user immersion, fail to reflect individual vocal characteristics, and impair emotional communication—factors that are especially critical in emotionally rich content environments.

To address these issues, voice cloning has emerged as a promising alternative. Voice cloning refers to the process of synthesizing speech that replicates the voice characteristics of a target speaker using only a short sample [7]. It is considered an extension of speaker adaptation techniques in TTS. Recent developments demonstrate that high-fidelity, natural-sounding speech can be generated from voice samples as short as 10 to 20

seconds, highlighting the technology's potential for personalized audio content and emotionally aware applications. Transfer learning approaches, as demonstrated by Jia et al. [2], enable the synthesis of novel speaker voices using only a few seconds of reference audio, making personalized TTS feasible with minimal data.

This project proposes a personalized AI audiobook application based on voice cloning. By submitting a 15-second voice sample, users can generate a unique AI voice model that reads aloud PDF documents with emotional intonation. The system was built using the React Native (EXPO) framework, with Firebase for user data management and the ElevenLabs API for high-quality real-time TTS processing, enabling seamless integration between frontend and backend components.

During the initial development phase, we experimented with open-source TTS models such as Tacotron2 and GlowTTS. However, constraints related to limited training data and high computational requirements resulted in suboptimal speech quality and implementation complexity. To overcome these limitations, we adopted the ElevenLabs API, which provides stable, high-quality, emotionally expressive voice synthesis.

The primary contribution of this project lies in its integration of accessible voice cloning technology with real-time TTS applications, enabling a high degree of personalization and immediacy in user interaction. Unlike prior studies that focused primarily on system performance or static voice models, our approach emphasizes inclusivity and emotional connectivity. Moreover, recent findings suggest that people often struggle to distinguish short AI-generated voice samples from real voices, which supports the immersive nature of personalized synthetic speech in this context [11]. The target users include individuals with visual impairments, the elderly, and those seeking to preserve or reconnect with the voice of a loved one. As such, voice cloning is positioned not merely as a technical enhancement to TTS, but as a socially impactful technology that "can also be utilized to preserve or restore the unique voice of individuals who have difficulty communicating".

II. System Design & Implementation

This study presents the design and implementation of TonePick, a mobile application that generates personalized audiobooks using the user's voice. The system architecture was designed with scalability, usability, and real-time performance in mind.

A. Overall System Architecture

The system is composed of a client-based mobile application, a cloud-based backend infrastructure, and an external AI TTS API. The client application is developed using React Native (Expo) to support both iOS and Android platforms. Firebase is responsible for user authentication, file storage, and real-time database operations [3]. High-quality voice cloning and speech synthesis are performed via the external ElevenLabs API [4].

This modular architecture enables seamless interaction between components. It also provides a flexible foundation for future expansion and service enhancement. The overall architecture and data flow are illustrated in Fig. 1.

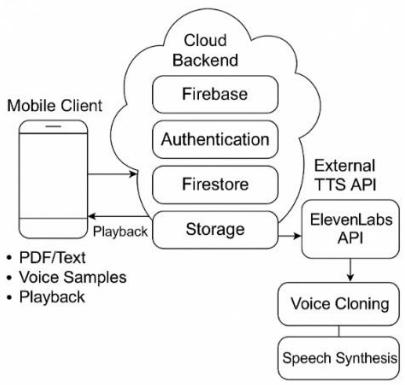

Fig. 1 Overall system architecture of TonePick integrating mobile client, Firebase backend, and ElevenLabs TTS API

B. Frontend Implementation

The frontend is built using React Native, leveraging libraries such as React Navigation, AsyncStorage, and Expo Modules (e.g., Audio, FileSystem, ImagePicker) to enhance functionality and user experience [8]. The UI is designed with both Styled Components and StyleSheet to ensure responsive layouts and consistent visual design.

Each screen—such as MainPage, FilePage, VoicePage, and FileDetailPage—was implemented as an independent functional unit. Users can upload PDF files, play back voices, bookmark content, and synthesize TTS audio through an intuitive interface.

User-specific data such as Voice IDs were stored locally using AsyncStorage, ensuring persistence even in offline scenarios.

C. *Backend and Database Integration*

The backend is built on Firebase Authentication, Firestore, and Firebase Storage. Authentication was implemented via email-based login and registration. Firestore manages metadata in real time, including PDF documents, audio files, and Voice IDs.

Files are stored in Firebase Storage and are accessible via unique URLs. All operations are performed via RESTful APIs, ensuring real-time synchronization between the frontend and backend and maintaining data consistency.

D. *Voice Cloning and TTS Pipeline*

To enable personalized voice synthesis, the system integrates the ElevenLabs Voice Cloning API. Users are prompted to record seven short voice samples, which are sent as a multipart/form-data POST request to the voice registration endpoint.

Upon successful submission, a unique Voice ID is generated. This Voice ID is then used to synthesize speech by sending the extracted PDF text to the corresponding TTS endpoint. The server responds with a high-quality MP3 audio stream containing emotional expression.

The resulting audio is stored in Firebase Storage and linked with the original document in Firestore, enabling persistent and contextual playback. The endpoints used in this pipeline are summarized in Table I.

The overall pipeline workflow, including voice collection, generation, and playback, is described in detail in Section F and visualized in Fig. 3.

TABLE I
DESCRIPTION OF ELEVENLABS API ENDPOINTS

Method	Endpoint	Purpose
POST	/v1/voices/add	Upload user voice samples and create Voice ID
POST	/v1/text-to-speech/{voice_id}	Generate TTS audio using the custom voice

E. *PDF Processing and Text Extraction*

When a user uploads a PDF file through the mobile application, the file is first transmitted to Cloudinary, which generates a publicly accessible file URL [9]. This URL is then forwarded to a Flask-based server, where the core text extraction process takes place [10].

The Flask server utilizes Python libraries such as pdfminer.six to parse and extract the raw textual content from the uploaded PDF [5] . The extracted text is subsequently passed through a preprocessing pipeline to improve its compatibility with the TTS (Text-to-Speech) engine. This includes sentence segmentation, removal of special characters, whitespace normalization, and paragraph structuring.

Once preprocessing is complete, the resulting clean text is returned to the mobile application and stored in Firebase Firestore, where it is linked with the corresponding document and user metadata. This allows the text to be reused in future TTS synthesis without the need for repeated processing.

The end-to-end process is illustrated in Fig. 2, and a summary of the components and their roles is provided in Table II.

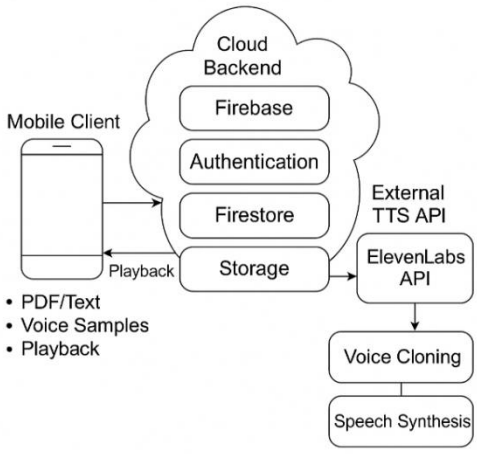

Fig. 1 Overall system architecture of TonePick integrating mobile client, Firebase backend, and ElevenLabs TTS API

TABLE II
COMPONENTS OF THE PDF PROCESSING AND TEXT EXTRACTION PIPELINE

Component	Role
Cloudinary	Stores the uploaded PDF and returns a public file URL
Flask Server	Extracts and pre-processes text from the PDF
Python Parser(pdfminer.six)	Parses and converts PDF content to raw text
Pre-processing Logic	Cleans and segments text for TTS compatibility
Firestore	Stores the final text and links it to the corresponding document

F. Voice Synthesis Pipeline using ElevenLabs API

To enable the generation of personalized audiobooks based on the user's voice, the system implements a structured voice synthesis pipeline using the ElevenLabs API. This process allows users to create a custom voice model from a limited set of recordings and generate TTS (text-to-speech) audio content with emotional expression. The pipeline consists of the following stages.

1) Voice Sample Collection: Users are prompted to record seven short sentences through the RecordPage in the mobile application. Each recording can be replayed for quality assurance before final submission. These audio samples serve as training data for the voice cloning process. Multi-modal learning has recently been shown to improve the effectiveness of few-shot voice cloning, especially in low-resource conditions like ours [12].

2) Voice Model Generation (Voice ID Creation): After completing the recordings, the user assigns a custom voice name (voiceName). The seven recorded files and the name are packaged as multipart/form-data and submitted to the ElevenLabs endpoint https://api.elevenlabs.io/v1/voices/add. Upon successful processing, the server generates a unique Voice ID, which identifies the synthesized voice in subsequent API calls.

3) Text-to-Speech Synthesis: When the user selects text extracted from a PDF, the application sends a request to https://api.elevenlabs.io/v1/text-to-speech/{voice_id} along with the following parameters:

- voice_id: The previously generated identifier
- model_id: "eleven_multilingual_v2" (supports multilingual synthesis and emotional tone)
- text: The input text string
- output_format: "mp3_44100_128"

The API returns a base64-encoded MP3 file containing the generated speech in the user's cloned voice.

4) Audio Processing and Playback: The received audio is decoded from its base64 format and saved to the device's local storage. The application provides real-time playback through screens such as VoicePage and FileDetailPage, allowing users to immediately verify and reuse the generated speech.

5) Metadata Storage and Reusability: All relevant metadata—including the Voice ID, voiceName, and audio file path—is stored in Firebase Firestore. This allows the personalized voice to be reused across various documents, particularly in the audiobook playback feature.

6) Process Visualization: The entire voice synthesis workflow is illustrated in the following diagram:

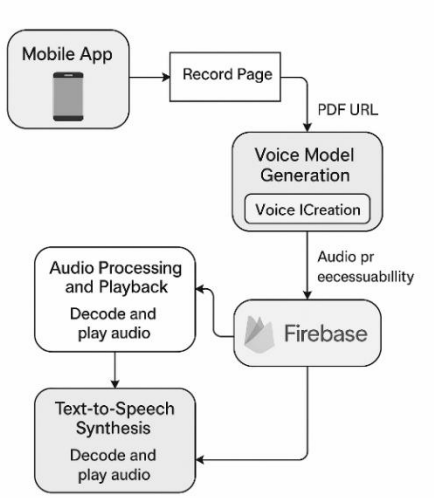

Fig. 3 Voice Synthesis Pipeline using ElevenLabs API

III. Result and Discussion

To evaluate the effectiveness and user experience of the proposed TonePick application, we conducted a user study with 27 participants, primarily in their 20s. A structured questionnaire was distributed via Naver Form after each participant completed a full session, including voice recording, voice synthesis, and playback. The questionnaire included nine items rated on a 5-point Likert scale and one open-ended question to capture qualitative impressions.

A. *Quantitative Analysis*

Table III presents the average scores collected from the user survey. The results demonstrate consistently positive feedback across all categories, particularly in terms of emotional realism and satisfaction.

TABLE III
AVERAGE EVALUATION SCORES (N=27) BASED ON 5-POINT LIKERT SCALE ITEMS

Evaluation Item	Avg. Score (out of 5)
Ease of Use	4.2
UI Intuitiveness	4.2
TTS Speed Satisfaction	4.2
Overall Satisfaction	4.6
Naturalness of AI Voice	4.4
Similarity to Real Voice	4.3
Human-like Narration Perception	4.3
Emotion Conveyance (tone, pitch)	4.3
Emotional Impact of Using Own Voice	4.5

B. *Qualitative Feedback*

In the open-ended responses, users described the experience as “immersive,” “like narrating to myself,” and “different from commercial TTS systems.” This suggests that personalized TTS not only enhances comprehension but also deepens emotional involvement.

C. *Application Potentials*

Based on both the system performance and user responses, the following practical use cases are identified:

- **Education:** Parents or teachers can record audiobooks or lessons in their own voice to foster familiarity and trust.
- **Accessibility:** Visually impaired or elderly individuals can listen to content using familiar voices for improved understanding and comfort.
- **Memory Preservation:** Users can archive meaningful voices (e.g., those of deceased loved ones) for long-term emotional storage.
- **Author Narration:** Book authors can narrate their own writing, expressing emphasis and emotion as originally intended, thereby enhancing the authenticity and interpretative fidelity of the audiobook.

IV. Conclusions

This study presented the design, implementation, and evaluation of TonePick, a mobile application that enables users to generate personalized audiobooks using their own voice. By integrating modern voice cloning techniques with real-time Text-to-Speech (TTS) synthesis, the system provides a scalable and emotionally engaging platform for voice-driven content consumption.

Through the use of the ElevenLabs API, the application successfully overcame the limitations of conventional TTS systems—such as lack of personalization and emotional expression—by allowing users to generate natural, expressive audio from just a few voice samples. The modular architecture based on React Native, Firebase, and Flask further ensured cross-platform compatibility and seamless backend communication.

The results from a user study involving 27 participants indicate strong satisfaction in terms of usability, voice naturalness, and emotional realism. Both the quantitative survey responses and qualitative feedback underscore

the application's potential not only as a technical solution but also as a socially meaningful tool for accessibility, education, and memory preservation.

Future work may explore multi-language support, further reduction of latency, and integration of more flexible voice editing options. In sum, TonePick demonstrates the potential of voice cloning as a transformative technology that bridges technical innovation with emotional resonance and personal identity. However, future implementations should also consider ethical implications, as synthetic voice services have been shown to reinforce accent biases and contribute to digital exclusion for underrepresented speech communities [13].

ACKNOWLEDGEMENT

This research was supported by the MSIT (Ministry of Science ICT), Korea, under the National Program for Excellence in SW, supervised by the IITP (Institute of Information & Communications Technology Planning & Evaluation) in 2025"(No. 2024-0-00023).

REFERENCES

[1]. Y. Wang, R. Skerry-Ryan, D. Stanton, et al., “Tacotron: Towards end-to-end speech synthesis,” in *Proc. Interspeech*, 2017, pp. 4006–4010.
[2]. Y. Jia, Y. Zhang, R. Weiss, et al., “Transfer learning from speaker verification to multispeaker text-to-speech synthesis,” in *Proc. NeurIPS*, 2018.
[3]. Firebase Documentation. Accessed on: Jun. 13, 2025. [Online]. Available: https://firebase.google.com/docs
[4]. ElevenLabs API Reference. Accessed on: Jun. 13, 2025. [Online]. Available: https://docs.elevenlabs.io
[5]. pdfminer.six Documentation. Accessed on: Jun. 13, 2025. [Online]. Available: https://pdfminersix.readthedocs.io/
[6]. S. Ping, et al., “Deep Voice: Real-Time Neural Text-to-Speech,” Proc. ICML, pp. 195–204, 2020.
[7]. S. Arik et al., “Neural Voice Cloning with Few Samples,” *Advances in Neural Information Processing Systems*, vol. 31, 2018.
[8]. React Native Documentation. Accessed on: Jun. 13, 2025. [Online]. Available: https://reactnative.dev/docs
[9]. Cloudinary Documentation. Accessed on: Jun. 13, 2025. [Online]. Available: https://cloudinary.com/documentation
[10]. Flask Documentation. Accessed on: Jun. 13, 2025. [Online]. Available: https://flask.palletsprojects.com/
[11]. J. Ma et al., "People are Poorly Equipped to Detect AI-Powered Voice Clones," in Proc. USENIX Security Symposium, 2023.
[12]. S. Liu et al., "Improve Few-shot Voice Cloning using Multi-Modal Learning," in Proc. NeurIPS 2021, pp. 1–10, 2021.
[13]. R. Hutchinson et al., “It's not a representation of me: Examining Accent Bias and Digital Exclusion in Synthetic AI Voice Services,” in Proc. CHI 2023, pp. 1–12.

