

AI Based Voice Cloning System: From Text to Speech

Md. Sadik¹; P. Vijaya²; Y. Revathi³; V. Siva Naga Tanuja⁴;
B. Soudhamini⁵; R. Vaishnavi⁶

¹ Assistant Professor; ^{2,3,4,5,6} Student,
CSE Dept, Sri Vasavi Engineering College, Tadepalligudem, A.P., India

Publication Date: 2025/04/26

Abstract: The rapid advancements in Artificial Intelligence and Deep Learning have significantly improved Text-To-Speech (TTS) technology, enabling more accurate and natural voice conversion. This project presents a Voice Cloning System that leverages a Transformer-based encoder and a GAN-based vocoder to generate high-quality, natural-sounding speech from text. The system supports both Text-to-Speech (TTS), where textual input is converted into a default synthesized voice, and Voice Cloning, which allows the replication of a new voice using a short audio sample. By employing a one-shot learning approach, the system enables speaker adaptation with minimal training data, making it efficient and scalable for real-world applications. The Transformer-based encoder effectively captures linguistic and prosodic features, while the GAN-based vocoder enhances the realism of the generated speech by refining spectral details. The model's ability to generalize across different speakers ensures robustness, even when trained on limited datasets. This project highlights the potential of deep generative models in speech synthesis and their impact on various domains, including assistive technology, where it can help individuals with speech impairments communicate more naturally, personalized virtual assistants that adapt to user preferences, and entertainment industries for voiceovers and character dubbing.

Keywords: Voice Cloning, Text-to-Speech (TTS), Transformer Encoder, GAN-based Vocoder, One-Shot Learning, Speech Synthesis, Deep Learning, Artificial Intelligence, Speaker Adaption, Generative Models, Speech Conversion.

How to Cite: Md. Sadik; P. Vijaya; Y. Revathi; V. Siva Naga Tanuja; B. Soudhamini; R. Vaishnavi (2025), AI Based Voice Cloning System: From Text to Speech. *International Journal of Innovative Science and Research Technology*, 10(4), 1453-1461. <https://doi.org/10.38124/ijisrt/25apr834>

I. INTRODUCTION

Speech synthesis has experienced significant advancements with the emergence of Artificial Intelligence (AI) and Deep Learning, particularly in Text-to-Speech (TTS) and Voice Cloning technologies. Traditional TTS systems required extensive speaker-specific datasets and relied on rule-based synthesis, often resulting in robotic and unnatural speech. However, modern neural network-based approaches, such as Transformer models and Generative Adversarial Networks (GANs), have revolutionized speech generation by enhancing its naturalness, adaptability and efficiency. This project introduces a Voice Cloning System that employs a Transformer-based encoder to process input text and extract linguistic and speaker-specific features. It further utilizes a GAN-based vocoder to refine the synthesized speech, ensuring high-quality and human-like voice output. A major advantage of this system is its one-shot learning capability, which enables speaker adaptation from just a short audio sample, eliminating the need for large-scale data for the training of the model. The ability to generate natural-sounding speech with minimal input has significant implications across various domains. In assistive technology, it can help individuals with speech impairments regain their

voice. Personalized virtual assistants can benefit from adaptive voice synthesis, making AI interactions more engaging. In the media and entertainment industry, this technology can enhance dubbing, audiobook narration, and video game voiceovers. Additionally, it can improve accessibility solutions for the visually impaired by providing more natural and expressive synthesized speech.

By leveraging deep generative models, this system pushes the boundaries of voice synthesis, demonstrating the effectiveness of AI-driven approaches in speech technology.

II. LITERATURE SURVEY

"ReVoice: A Neural Network based Voice Cloning System," authored by Sahil Kadam, Aditya Jikamade, Prabhat Mattoo, and guided by Prof. Varsha Hole, was presented at the 2024 IEEE 9th International Conference for Convergence in Technology (I2CT). This study introduces "ReVoice," a system designed to replicate human voices using advanced neural network architectures. Leveraging the LibriTTS dataset, the authors trained the model to ensure high-quality, natural-sounding speech synthesis. The architecture integrates components like Tacotron2 for converting text to

mel-spectrograms, an encoder and synthesizer for capturing and reproducing vocal nuances, and the WaveRNN vocoder to transform spectrograms into audible waveforms. This combination allows ReVoice to achieve accurate and expressive voice cloning. The research contributes to the field of speech synthesis by offering insights into the integration of these technologies for effective voice replication.

"Voice Cloning in Real Time," authored by Varad Naik, Aaron Mendes, Saili S. Kulkarni, Saish Naik, and Saish Prabhu Verlekar, was published in the International Journal for Research in Applied Science and Engineering Technology in August 2022. The paper explores advancements in voice cloning technologies, emphasizing the transition from early models requiring extensive data to contemporary approaches capable of synthesizing speech with minimal input. Key techniques discussed include speaker adaptation, which involves fine-tuning pre-trained models to new speakers; speaker encoding, which utilizes embeddings to capture unique vocal characteristics; and vector quantization, a method for compressing and reconstructing audio data. These methodologies aim to enhance the efficiency and realism of voice cloning systems, enabling applications such as personalized speech synthesis and real-time voice conversion.

"A Survey on Text Pre-Processing & Feature Extraction Techniques in Natural Language Processing," authored by Ayisha Tabassum and Dr. Rajendra R. Patil in 2020, provides a comprehensive overview of essential techniques in Natural Language Processing (NLP). The paper emphasizes the significance of text pre-processing methods such as sentence segmentation, converting text to lowercase, tokenization, parts-of-speech tagging, stop-word removal, punctuation removal, stemming, and lemmatization. These techniques are crucial for cleaning and structuring raw text data, facilitating more accurate and efficient analysis by machine learning algorithms. Additionally, the survey discusses feature extraction methods like Named Entity Recognition (NER), Bag-of-Words (BoW), and Term Frequency-Inverse Document Frequency (TF-IDF), which are instrumental in representing text data numerically for computational processing. The authors highlight the importance of the sequence in which pre-processing steps are applied, noting that errors in early stages can propagate and adversely affect subsequent analysis. This survey serves as a valuable resource for understanding the foundational techniques in

NLP and their practical applications in tasks such as text classification, information retrieval, and the development of chatbots and virtual assistants.

The paper titled "Voice Converter Using DeepSpeech and Tacotron" authored by Sree Nithy Chandran from 2020. DeepSpeech is an open-source automatic speech recognition (ASR) engine that transcribes spoken language into text. Tacotron, on the other hand, is a sequence-to-sequence model developed by Google for text-to-speech (TTS) synthesis, converting written text into natural-sounding speech. When combined, these models can facilitate a voice conversion system by first transcribing input speech into text using DeepSpeech and then generating speech in a target voice using Tacotron. In this system, the process begins with DeepSpeech transcribing the source speaker's audio into text. This text, along with a reference audio sample of the target speaker, is fed into Tacotron. Tacotron then synthesizes speech that mirrors the target speaker's voice while conveying the content of the original speech. This approach enables applications like personalized speech synthesis and assistance for individuals who have lost their voice. While this general methodology is widely recognized, specific implementations may vary based on the system's design and objectives.

III. METHODOLOGY

➤ Proposed System

This project focuses on developing an AI-based voice cloning system that can generate natural and expressive speech while mimicking a specific speaker's voice. It leverages OutetTS, a transformer-based text-to-speech (TTS) model, which provides high-quality and realistic speech synthesis. The system takes input text and converts it into speech using pre-trained speaker embeddings, allowing for speaker adaptation with minimal training data. The project includes a voice cloning module, where a user can upload an audio sample, and the model will learn the speaker's voice characteristics to generate speech in the same voice. This is achieved using deep learning-based speech synthesis techniques, ensuring fluency, clarity, and expressiveness. The system is designed for applications in virtual assistants, audiobooks, dubbing, and assistive speech technologies while maintaining efficiency and scalability.

➤ System Architecture

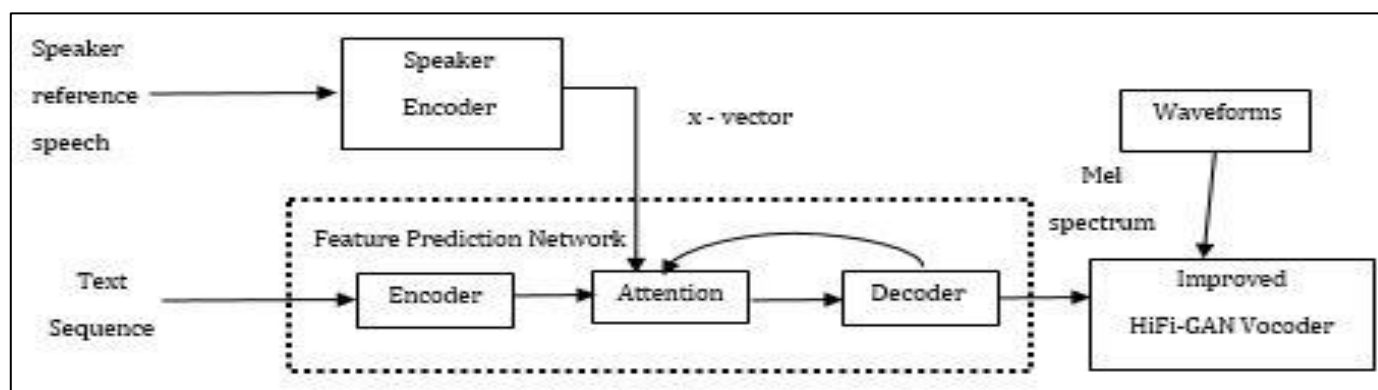


Fig 1 System Architecture

- *Speaker Encoder*

The speaker encoder extracts speaker-specific features from a given reference speech sample. It generates an x-vector, a numerical representation of the speaker's characteristics, such as tone, pitch, and speaking style.

- *Feature Prediction Network*

This module takes the text sequence and the extracted x-vector as inputs and processes them through the following components:

- ✓ *Encoder*

It converts the input text sequence into a feature representation. It learns the contextual relationships between words and converts the text into a form that can be used for speech synthesis.

- ✓ *Attention Mechanism*

The attention mechanism helps the model to align the features of the text with the features of the speaker. It ensures that the right parts of the text sequence are attended to when generating the speech output.

- ✓ *Decoder*

The decoder takes the attended feature representation and transforms it into an intermediate mel-spectrogram representation, which represents the frequency and amplitude of the speech signal over time.

- *Improved HiFi-GAN Vocoder*

The HiFi-GAN vocoder converts the generated mel-spectrogram into waveform. This step synthesizes the final speech audio, making it more natural and realistic.

- *Final Output: Waveform*

The generated waveform is the final speech output, which sounds similar to the speaker's voice used in the reference sample.

- *Saving the Audio File*

The generated speech is saved as an .wav file, allowing users to store and reuse the audio.

➤ *Process Flow*

The Voice Cloning System follows a step-by-step structured pipeline, starting with data preprocessing, where input text is tokenized, normalized, and converted into phonetic representations, while audio samples undergo feature extraction using Mel-spectrograms to capture the speaker-specific characteristics. In the feature extraction stage, a Transformer-based encoder processes the text input, extracting linguistic features such as phoneme duration, intonation, and rhythm. Simultaneously, a speaker embedding module analyses the short audio sample, extracting essential voice attributes like pitch, tone, and timbre. These embeddings are then combined using a speaker adaptation mechanism that employs one-shot learning to create a unique speaker representation, allowing the system to mimic the target voice without extensive training data.

Once the feature embeddings are generated, they are passed to the speech synthesis module, where a sequence-to-sequence model aligns the text with the cloned speaker's voice characteristics. The processed output is then sent to a GAN-based vocoder, which converts the intermediate feature representation into a high-fidelity waveform by enhancing spectral details, reducing distortions, and making the synthesized voice sound more natural. The generated speech undergoes post-processing, including noise reduction, volume normalization, and prosody adjustment, ensuring clarity and smoothness.

Finally, the system supports two main functionalities: Text-to-Speech (TTS), where any text is converted into a default synthesized voice, and Voice Cloning, where a short audio sample enables speaker adaptation, replicating the target voice with high accuracy. The entire process is optimized for real-time performance, making it efficient for applications in assistive technology, virtual assistants, media production, and accessibility solutions. Future improvements could include multilingual support, emotion-based synthesis, and fine-grained prosody control, further enhancing the system's adaptability and realism.

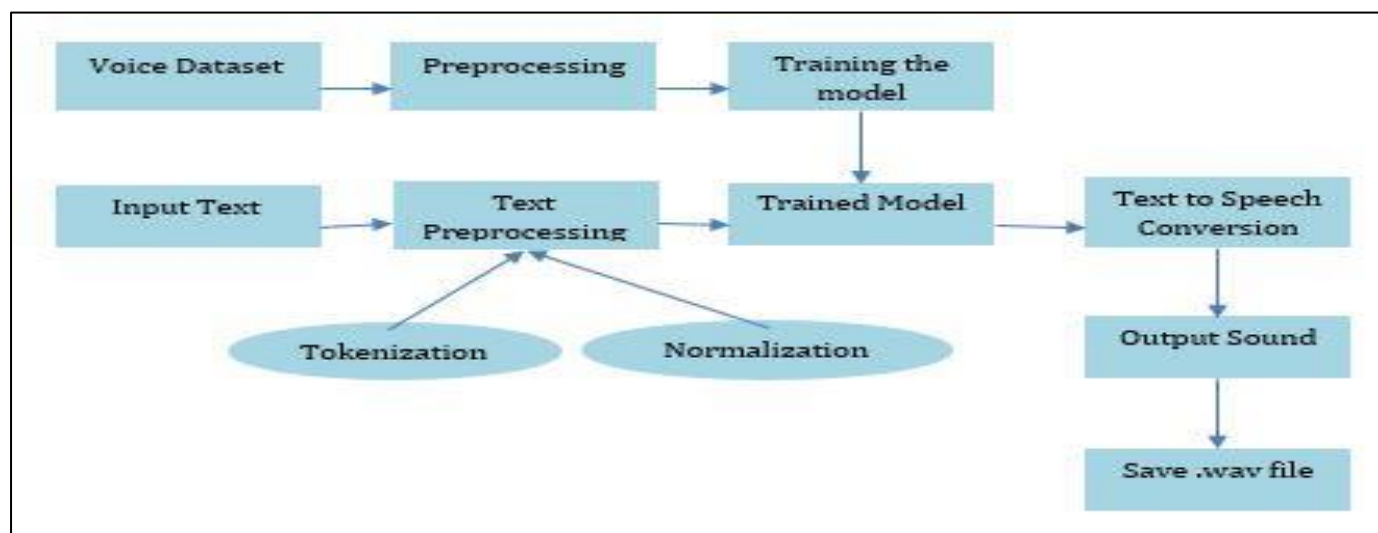


Fig 2 Proposed System

- *Voice Dataset*

A set of audio recordings containing various speech samples used to train and evaluate voice cloning model.

- *Preprocessing*

The process of cleaning, trimming, and formatting raw audio data to ensure uniformity and compatibility with the model training pipeline.

- *Training the Model*

Utilizing deep learning algorithms to analyse voice data and learn the mappings required for accurate text-to-speech generation.

- *Trained Model*

A fully optimized model that has learned voice characteristics and can generate realistic, human-like speech from textual input.

- *Input Text*

The user-provided text which the system will convert into synthetic speech using the trained model.

- *Text Preprocessing*

Breaking down the input text, removing noise, and applying linguistic rules to prepare it for further speech synthesis steps.

- *Tokenization*

Dividing the cleaned text into smaller linguistic units such as phonemes, syllables, or words for easier model processing.

- *Normalization*

Standardizing the text input by expanding abbreviations, converting numerals to words, and correcting grammar for consistency.

- *Text to Speech Conversion*

The trained model interprets the processed text and generates corresponding speech using learned acoustic and linguistic patterns.

- *Output Sound*

The final audio output generated by the system, representing the spoken version of the input text provided by the user.

- *Save .wav file*

The generated speech is exported and stored as a high-quality .wav audio file for playback or further analysis.

IV. IMPLEMENTATION

➤ *Training Datasets*

- *Emilia-Dataset (CC BY-NC 4.0)*

The Emilia-Dataset is a high-quality speech dataset created for voice cloning, speech synthesis, and text-to-speech (TTS) applications. It contains recorded speech data from a single speaker, often with detailed phonetic

transcriptions. Since it is licensed under CC BY-NC 4.0, it allows non-commercial use, meaning it can be used for research and academic purposes but not for commercial applications. This dataset is particularly useful for training speaker-adaptive TTS models and one-shot voice cloning tasks.

- *LibriTTS-R (CC BY 4.0)*

LibriTTS-R is an improved and refined version of the original LibriTTS dataset, derived from LibriSpeech, a well-known speech corpus. It is designed specifically for text-to-speech (TTS) synthesis and includes high-quality recordings with better alignment between text and speech. The dataset is freely available under the CC BY 4.0 license, meaning it can be used for both commercial and non-commercial projects. It is widely used in deep learning-based TTS systems, voice cloning, and speaker adaptation.

- *Multilingual LibriSpeech (MLS) (CC BY 4.0)*

The Multilingual LibriSpeech (MLS) dataset is a large-scale multilingual speech dataset derived from audiobooks in the LibriVox project. It contains speech recordings in multiple languages, making it a valuable resource for multilingual text-to-speech (TTS), automatic speech recognition (ASR), and speech translation tasks. The dataset is openly available under the CC BY 4.0 license, allowing both commercial and non-commercial use. It is particularly beneficial for training speech synthesis models that support multiple languages and improving the generalization of TTS systems.

➤ *Model Configuration*

To begin the voice cloning process, the Outetts TTS system is initialized with the required parameters and model setup. The steps involved are as follows:

- *Loading the OuteTTS Model*

The core component of the voice cloning system is the OuteTTS-0.2-500M model, a high-performance transformer-based TTS model offered by the OutetAI framework. This model is loaded using the outetts library, which abstracts the complexities of model initialization, hardware selection, and interface handling.

- *Specifying the Model Path and Language*

The model path parameter indicates the pre-trained checkpoint used for text-to-speech synthesis. In this implementation, the publicly available OuteAI/OuteTTS-0.2-500M checkpoint is utilized, which supports multilingual capabilities, though English ("en") is selected as the target language. The configuration object HFModelConfig_v1 binds the model architecture, language, and tokenizer together to ensure correct phoneme conversion and pronunciation handling.

- *Defining Speaker Profiles*

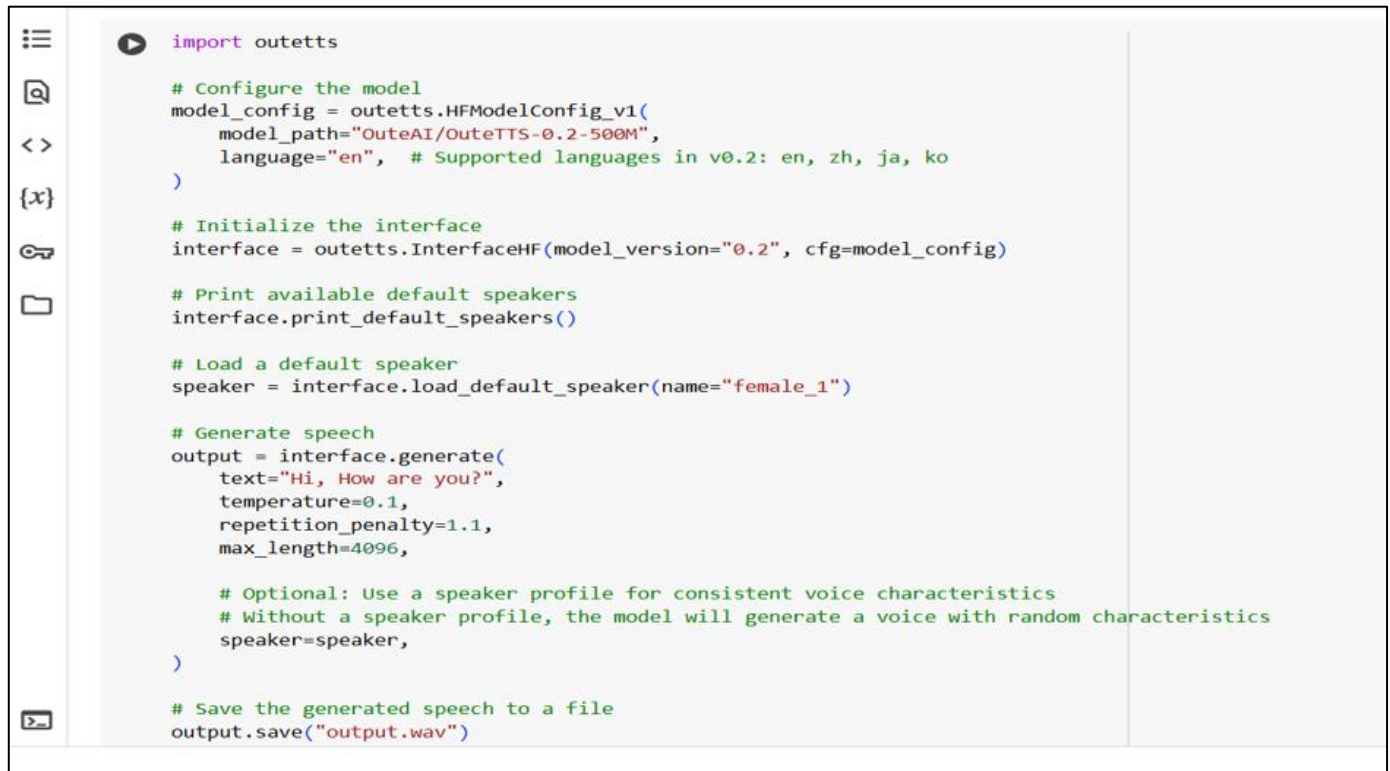
Speaker profiles play a critical role in determining the voice characteristics used during synthesis. The OuteTTS system supports two primary modes of speaker definition:

✓ *Default Speaker Profiles:*

These are pre-trained voice embeddings provided by the OutetAI model. Users can choose from voices such as "female_1", "male_3", etc.

✓ *Custom Voice Cloning (One-shot):*

In this mode, the system creates a speaker profile from a user-provided audio sample. The audio is processed using an internal speaker encoder, and optionally transcribed using Whisper ASR. This enables cloning a voice from just a few seconds of speech.



```
import outetts

# Configure the model
model_config = outetts.HFModelConfig_v1(
    model_path="OuteAI/OuteTTS-0.2-500M",
    language="en", # Supported languages in v0.2: en, zh, ja, ko
)

# Initialize the interface
interface = outetts.InterfaceHF(model_version="0.2", cfg=model_config)

# Print available default speakers
interface.print_default_speakers()

# Load a default speaker
speaker = interface.load_default_speaker(name="female_1")

# Generate speech
output = interface.generate(
    text="Hi, How are you?",
    temperature=0.1,
    repetition_penalty=1.1,
    max_length=4096,

    # Optional: Use a speaker profile for consistent voice characteristics
    # Without a speaker profile, the model will generate a voice with random characteristics
    speaker=speaker,
)

# Save the generated speech to a file
output.save("output.wav")
```

Fig 3 Model Configuration



```

v Voice Cloning

[ ] speaker = interface.create_speaker(
    audio_path="/content/OSR_us_000_0010_0k.wav",

    # If transcript is not provided, it will be automatically transcribed using whisper
    transcript=None, # Set to None to use whisper for transcription

    whisper_model="turbo", # Optional: specify Whisper model (default: "turbo")
    whisper_device=None, # Optional: specify device for Whisper (default: None)
)

2025-03-23 14:10:01.471 | INFO | outetts.version.v1.interface:create_speaker:122 - Transcription not provided, transcribing audio with whisper.
2025-03-23 14:10:01.472 | INFO | outetts.whisper.transcribe:transcribe_once:5 - Loading model turbo
2025-03-23 14:10:16.734 | INFO | outetts.whisper.transcribe:transcribe_once:7 - Transcribing /content/OSR_us_000_0010_0k.wav
2025-03-23 14:10:19.141 | SUCCESS | outetts.whisper.transcribe:transcribe_once:9 - Transcription: The birch canoe slid on the smooth planks. Glue the sheet to the dark blue backgrou

[ ] # Save speaker profile
interface.save_speaker(speaker, "speaker.json")

# Load speaker profile
speaker = interface.load_speaker("speaker.json")

```

Fig 4 Voice Cloning

```

# Generate speech
output = interface.generate(
    text="Wow!She is amazing!! said by rohit",
    temperature=0.5,
    repetition_penalty=1.1,
    max_length=4096,

    # Optional: Use a speaker profile for consistent voice characteristics
    # Without a speaker profile, the model will generate a voice with random characteristics
    speaker=speaker,
)

# Save the generated speech to a file
output.save("output_cloned1.wav")

```

Fig 5 Speech Generation

```

[ ] from IPython.display import Audio
Audio("/content/OSR_us_000_0010_8k.wav", autoplay=True) #original

[ ] from IPython.display import Audio
Audio("output_cloned1.wav", autoplay=True) # converted

```

Fig 6 Synthesized Output Voice

V. EXPERIMENTAL RESULTS

➤ Performance Analysis

The below screenshot represents the similarity analysis between the actual and generated voice. The system evaluates the effectiveness of the voice cloning process using spectral analysis and similarity scores.

• Key Functionalities Shown

Cosine Similarity Score (99.66%): Indicates a high similarity between the generated and actual voice.

DTW (Dynamic Time Warping) Similarity Score (0.75%): A lower score suggests minimal deviation between the original and cloned voices.

Spectrogram Comparison: Visual representation of frequency and amplitude over time for both the actual and generated voices, showing structural alignment.

• Observations

The cloned voice closely matches the actual voice in terms of frequency distribution.

The high cosine similarity score validates the efficiency of the voice cloning model.

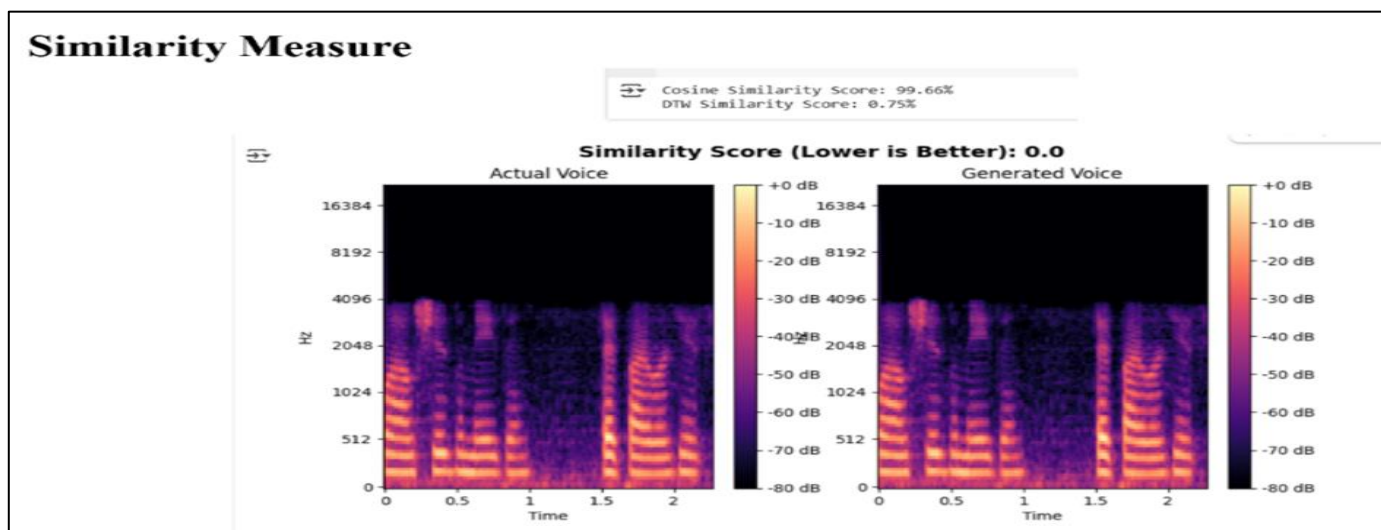


Fig 7 Visualization of Similarity Measure

➤ *Output*• *Input Page*

The below screenshot represents the input page of the AI-based Voice Cloning System.

✓ *Interface consists of two primary sections*▪ *Text-to-Speech (TTS) Module:*

Users can select a default speaker, enter text, and generate speech using the default voice model.

▪ *Voice Cloning Module:*

Users can upload an audio sample and input text, which will be converted into speech using the cloned voice.

✓ *Key Functionalities Shown*

- Speaker selection for default TTS
- Text input for speech synthesis
- Audio file upload for voice cloning
- Buttons for generating TTS and cloned speech

Fig 8 Input Page

• *Output Page*

The below screenshot shows the output generated after processing the user inputs. It displays the converted speech as audio waveforms.

✓ *Key Functionalities Shown*

- The Default Speaker Sample playback, generated using the text entered in the TTS module.
- The Uploaded Voice Sample, representing the original audio sample provided for cloning.

- The TTS Output, which plays the generated speech from the TTS module.
- The Cloned Speech Output, representing the synthesized speech that mimics the uploaded voice sample.

✓ *Observations*

The generated speech waveforms indicate successful text-to-speech and voice cloning processes.

The cloned voice output closely resembles the uploaded voice sample in waveform structure.

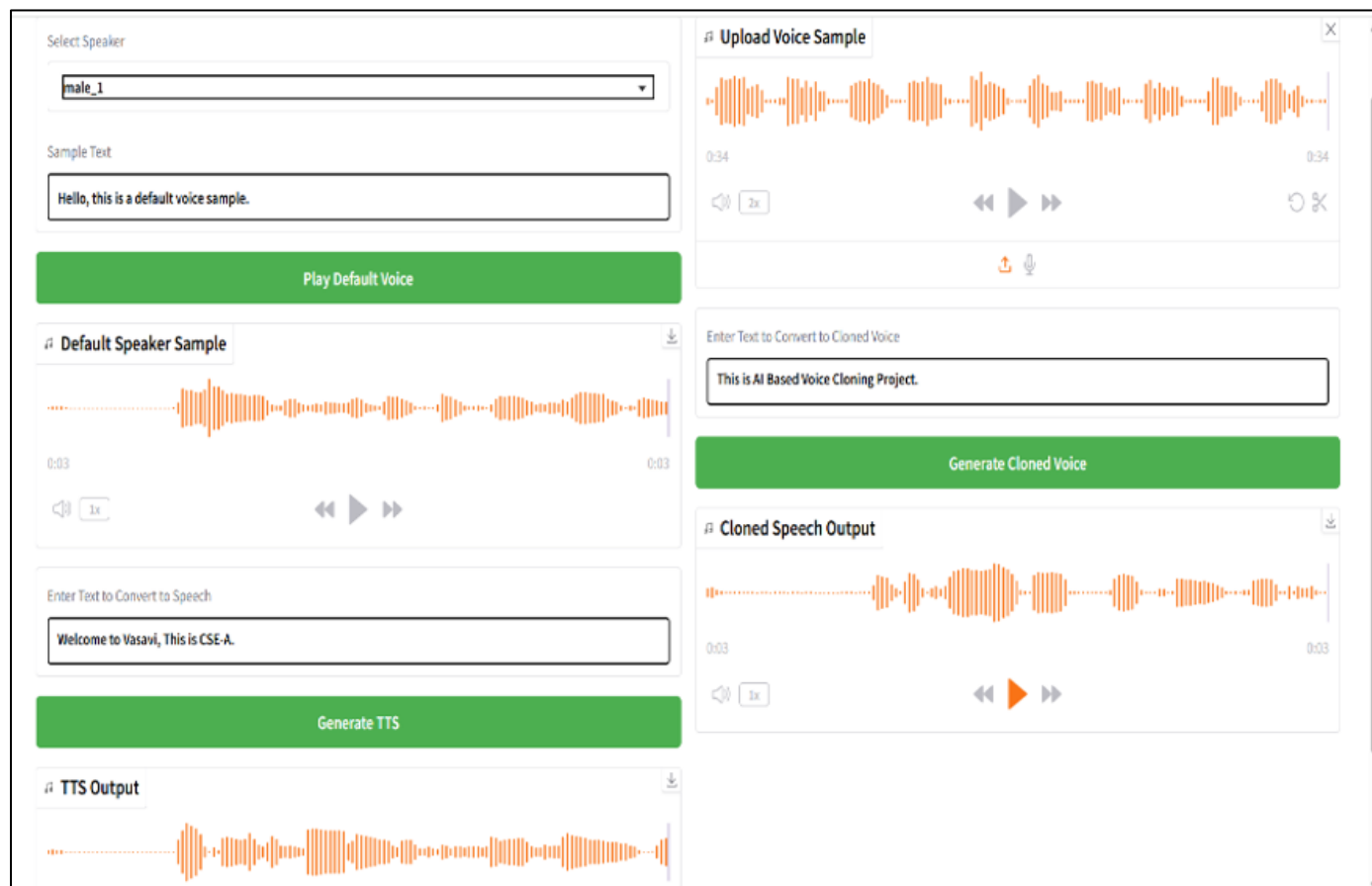


Fig 9 Output Page

➤ Functional Testing

• Input Validation Tests

Check how the model behaves with edge-case inputs (very long text, special characters, non-standard spelling, etc.).

• Speaker Profile Robustness

Test cloning quality using different lengths and quality of input voice samples.

VI. CONCLUSIONS

This project introduces a deep learning-based voice cloning system that generates realistic speech from limited voice samples. Using one-shot learning, it adapts quickly to new voices without requiring large datasets. The model captures speaker-specific traits like tone, pitch, and pronunciation via deep neural networks, enabling natural voice synthesis.

Generative Adversarial Networks (GANs) power the system, with a generator producing speech and a discriminator refining realism. Speaker adaptation further personalizes the output to match individual voice characteristics. To evaluate performance, the system employs Mel-Frequency Cepstral Coefficients (MFCCs) for vocal similarity and Dynamic Time Warping (DTW) for rhythm and timing accuracy, ensuring high-fidelity, speaker-consistent results.

Applications include personalized virtual assistants, content creation, gaming, audiobook narration, e-learning, and accessibility tools—offering lifelike, adaptive audio experiences. This system signifies a major advancement in AI-powered speech synthesis.

REFERENCES

- [1] Wang Y, Skerry-Ryan R, Stanton D, et al. Tacotron: Towards End-to-End Speech Synthesis. Proceedings of InterSpeech 2017; 2017:4006–4010.
- [2] Shen J, Pang R, Weiss RJ, et al. Natural TTS Synthesis by Conditioning WaveNet on Mel Spectrogram Predictions. ICASSP 2018 - IEEE International Conference on Acoustics, Speech, and Signal Processing; 2018:4779–4783.
- [3] Jia Y, Zhang Y, Weiss RJ, et al. Transfer Learning from Speaker Verification to Multi-Speaker Text-To-Speech Synthesis. Advances in Neural Information Processing Systems (NeurIPS); 2018:4480–4490.
- [4] Ren Y, Hu C, Tan X, et al. FastSpeech: Fast, Robust and Controllable Text to Speech. Neural Information Processing Systems (NeurIPS); 2019.
- [5] Kumar KR, Gulati T, Zen H, et al. High-Fidelity Speech Synthesis with Improved Variational Autoencoders. Proceedings of InterSpeech 2019; 2019:2918–2922.
- [6] Arik et al. (2017): "Deep Voice: Real-time Neural Text-to-Speech". Proceedings of the 34th International

- Conference on Machine Learning (ICML), pp. 195-204.
- [7] Ping et al. (2018): "Cloning a Voice in 5 Seconds to Generate Arbitrary Speech in Real-time". Proceedings of the 19th Annual Conference of the International Speech Communication Association (Interspeech), pp. 392-396.
- [8] Chen et al. (2019): "Transformers for Speech Synthesis". Proceedings of the 20th Annual Conference of the International Speech Communication Association (Interspeech), pp. 270-274.
- [9] Liu et al. (2020): "Wavetone: A High-Quality, Flexible, and Efficient Text-to-Speech System". Proceedings of the 21st Annual Conference of the International Speech Communication Association (Interspeech), pp. 330-334.
- [10] Huang et al. (2020): "Voice Cloning using Generative Adversarial Networks". IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 28, pp. 1455-1465.
- [11] Kong et al. (2020): "HiFi-GAN: Generative Adversarial Networks for High-Fidelity Speech Synthesis". Proceedings of the 21st Annual Conference of the International Speech Communication Association (Interspeech), pp. 335-339.