Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 32 No. 10s (2025)

Voice Conversion using Hybrid CNN BiLSTM-WaveNet Deep Learning Models

A. Bala Raju[1], S. P Singh[2], Dhiraj Sunehra[3]

[1]Research Scholar, Department of Electronics and Communication Engineering, JNTUH, Kukatpally, Hyderabad.

[1]Assistant Professor, Department of Electronics and Communication Engineering, Mahatma Gandhi Institute of Technology, Hyderabad, Telangana, India.

[2]Professor, Department of Electronics and Communication Engineering, Mahatma Gandhi Institute of Technology, Hyderabad, Telangana, India.

[3]Professor, Department of Electronics and Communication Engineering, JNTUH UCER, Rajanna Sircilla, Telangana, India.

balaraju_ap@yahoo.co.in[1], spsingh@mgit.ac.in[2], dhirajsunehra@yahoo.co.in[3]

Article History:

Received: 12-01-2025

Revised: 15-02-2025

Accepted: 01-03-2025

Abstract: Voice conversion is an exciting area of speech processing in which deep learning approaches are developed that can modify the vocal qualities of an speaker to resemble the voice of another person without altering the context of the utterance. The significance of speech conversion cannot be overstated, as it is employed in a wide range of systems, including entertainment, vocal communication, and privacy enhancement. However, traditional methods have fallen short in the face of large data sets and the preservation of subtle emotions, hindering voice simulation. To address the above limitations, we present a novel way that combines the fusion of the Speech to Text technology with a text-to-speech transformation system powered by a deep learning architecture. The system contains advanced embedding layers like phoneme embedding, bidirectional Long Short-Term Memory (LSTM) networks, and WaveNet vocoder, which make the transformed voice more accurate and authentic. In the proposed model, we use the speech recognition tools packages of Python and complex neural network methods to improve the naturalness and clarity. Moreover, it sets a bar when it comes to processing power, efficiency, and performance.

Keywords: Voice Conversion, Deep Learning, Speech Processing, Speech to Text, Bidirectional LSTM, WaveNet Vocoder

Introduction:

Voice conversion is a domain of speech processing, and it is fascinating to interact by observing how the special human speech ability specifies in words is first converted into a different form that maintains the semantic bearing of what the original voice has spoken [1]. This means that the voice of the person is converted back to another person's voice with different pitch, voice, and accent. This technology has employed a combination of methods, including digital signal processing. In recent years, its accuracy has set new records, mainly due to new deep learning techniques [2].

Voice conversion has been vital in several applications, such as: entertainment – it has used the technology to overwrite films and video games when actors' voices need changing and produce multi-voiced characters [3]. Another application is in communication, where the speech becomes

clearer; people's voices are better understood using this technology on all channels and in a variety of languages and dialects. Another critical application of voice conversion is personal safety and privacy [4]. Since it does not consider a person speaking a different language, the problem is solved when talking to a person with speech difficulty. Therefore, it allows for anonymization in sensitive situations.

Currently, most methods used for converting voices are based on deep learning models [5]. Such models, known as Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and Generative Adversarial Networks (GAN), have only been introduced a decade ago. Thanks to learning intricate patterns and variations from huge datasets of speech, these technologies have radically improved the quality and the level of voices' naturalness that are converted [6].

While a great deal of improvement has been achieved in the field of voice conversion, there are still a number of research gaps to be tackled [7]. For instance, most existent methods bear a number of difficulties with large volumes of data processing. Therefore, they are not fully applicable for a titanic array of tasks. Additionally, the emotional nuances and subtle details of the original voice are particularly difficult to preserve. These factors deteriorate the authenticity and the effectiveness of the converted voice [8]. Therefore, understanding and removing these limitations may contribute to improving the capacities and applicability of speech conversion technologies. Hence, they may serve as prospective directions for subsequent research.

Literature Review:

Berrak Sisman et al [9] provided a thorough analysis of the current state-of-the-art speech conversion techniques and their manner of evaluating performance. It includes various approaches, ranging from statistical methods to deep learning. The authors have considered the potential benefits and drawbacks of these techniques. Furthermore, they will go over the latest Voice Conversion Challenges. Voice conversion involves various speech processing techniques including voice analysis, spectrum conversion, prosody conversion, speaker characterisation and vocoding. Briefly, thanks to recent advances in both the theory and practice, they can generate voice quality capable of perfectly imitating human speech and the speaker.

Yi Zhao et al [10] planned and executed the third edition of the challenge, and developed and disseminated a new database for two specific tasks, intra-lingual semi-parallal and cross-lingual VC. After a two-month challenge period, the authors have been able to collect a total of 33 entries, including three baseline submissions made using the database. It has been observed that the progress of VC techniques improves drastically as a result of higher quality deep learning techniques. For instance, in the task of intra-lingual semi-parallel VC, many of the systems' speaker similarity scores were within the range of those of the target speakers. Nevertheless, they have ascertained that none of them has achieved the degree of naturalness that humans exhibit for the same job. The cross-lingual conversion proved to be more difficult, as all systems obtained relatively lower naturalness and similarity column heading ratings compared to the intra-lingual conversion task.

Kun Zhou et al [11] introduced a review of the most innovative emotional voice conversion research and available emotional speech databases. Then, the authors argued the necessity to create an emotional speech database for the first time, addressing the emerging requirements at a rising trend

in research. The work has released the emotional speech database ESD to the entire research community. In brief, ESD contains 350 parallel utterances for building the model and converting 10 speakers from two native speakers for English and five emotion categories of Chinese. A total of almost 29 hours of speech data were used in a strictly controlled acoustic environment. The database is adequate for multi-speaker and multi-system investigations on emotional voice conversion in two different languages.

Disong Wang et al [12] utilized technique of Vector Quantization (VQ) for content encoding, while Mutual Information (MI) is introduced as the correlation measure during training. This approach strives for effective disentanglement of content, speaker, and pitch representations by minimizing their inter-dependencies in an unsupervised way.

Yinghao Aaron Li [13] introduced an unsupervised non-parallel many-to-many Voice Conversion (VC) method using a generative adversarial network (GAN) called StarGAN v2. The model substantially outperforms the past VC models by using a balanced mix of adversarial source classifier loss and perceptual loss. With only training a model solely based on 20 English speakers, the model can be easily adapted to a wide range of voice conversion tasks such as any-to-many, cross-lingual, singing conversion. The framework can also convert plain reading speech to their corresponding stylistic speech, such as emotional, and falsetto.

Songxiang Liu et al [14] proposed an entirely novel voice conversion technique involving an arbitrary number of input and output locations, a sequence, and even non-parallel transmission data. The system's training is guided by voice-based textual information. While training the phoneme recognizer, the authors combined the middle layer of the encoder with the BNE and the encoder-decoder with a hybrid connectionist-temporal-classification-attention CTC-attention model. An encoder in which the middle layer is fed by a BNE is generated as a by-product of the phoneme recognizer training process. This BNE is used with spectral data to produce middle representations that are speaker-independent, battle-tested, and comprehensive for spoken language. The seq2seq synthesis model is trained after that, using multi-speaker location-relative attention to recreate spectral qualities from the bottleneck data. Meanwhile, the speaker representation is raised to control the speaker information in the synthetic voice. The seq2seq model aligns extended sequences, a problem confronted by most seq2seq algorithms. Additionally, they modify the synthesizer using a discretized Mixture of Logistic attention mechanism. Due to the length learning procedure of phoneme recognition utilizing extensive speech recognition data, they can achieve any-to-many voice conversion using the approach.

Wen-Chin Huang et al [15] proposed that one could transfer information from other speech processing tasks, such as text-to-speech (TTS) and automated speech recognition (ASR), for which large-scale collections of high-quality data can be readily obtained. The authors believe that when initiated with parameters of pretrained ASR or TTS models, VC models could come up with powerful latent representations that can convert speech that is both of high quality and highly understandable at the same time. Evaluated and addressed through this technique, which is used in a parallel, one-to-one framework. They applied recurrent neural networks (RNN) and Transformer models.

Kun Zhou et al [16] proposed method attempts to seperate the manner of speaking from the linguistic content and model the speaker's style as a style embedding in a continuous space that acts as the prototype emotion embedding. The authors obtained the accurate emotion encoder with the emotion labels from the emotions database. Furthermore, they also analyzed the use of relative attributes to name the emotions with specific intensity. To improve personalization awareness for emotions, emotion classification loss and emotion embedding similarity loss are introduced to the training of the EVC network. The network suggested adequately controls the correct emotional intensity level in the produced speech.

Proposed Method:

In this method, the initial audio input is first given to an automatic speech recognition (ASR) system, to transcribe the spoken content into text. Subsequently, this text is fed into the voice conversion system. The speech-to-text component utilizes Python's speech recognition libraries to accurately convert spoken language into written text. Once the transcription is obtained, it is provided as input to the voice conversion model. The model then embarks on its synthesis process, beginning with the phoneme embedding, followed by feature extraction through convolutional layers, and sequential processing using bidirectional LSTMs within the encoder. The decoder utilizes location-sensitive attention to produce a Mel spectrogram, which is further enhanced by post-processing convolutional layers. The Mel spectrogram is converted into an audible waveform using a WaveNet vocoder. This represents the synthesized speech output, which is identical to the original audio input in form of text.

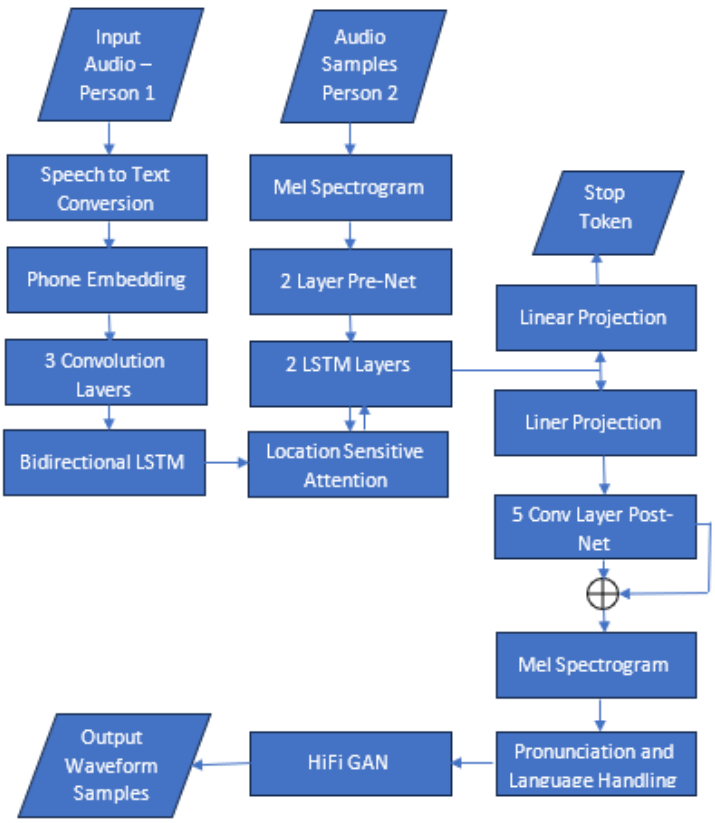


Figure 1: Proposed Framework

Voice conversion module:

The model first processes textual input and then converts it into a series of phonemes, which are translated into numerical representations. The encoder extracts feature from the representations. Then, the decoder uses the attention process to generate a Mel spectrogram. A vocoder turns the Mel spectrogram into a waveform, which can be spoken as the original input text.

Pre-processing is a critical part of voice conversion systems because it converts the raw input into a form that can be used to train models. This involves converting the audio recordings into Mel spectrograms, which are graphical representations of the audio's fundamental frequencies presented over time. The spectrograms are fundamental to learning from the model. The model proposed includes components that enable the automation of the conversion process and editing of the file lists that define the data to use for training. Ensuring that the data is in the right format and readily accessible to the model is a critical step that influences the training process's efficiency and effectiveness directly.

Encoder

Phone Embedding: The text is converted into a phonetic representation, where each unit of sound (phoneme) is transformed into a numerical vector using an embedding layer. This helps the model understand the text in terms of sounds rather than letters.

Convolution Layers: These layers are used to extract features from the phoneme embeddings. Convolutions help the model to capture the local context within the sequence of phonemes.

Bidirectional LSTM: A type of recurrent neural network that processes the sequence of features from both directions (forward and backward). This ensures that the model has access to past and future context, which is crucial for predicting the correct intonation and rhythm of speech.

Decoder: The decoder is responsible for generating the Mel spectrogram from the encoded features.

Two Layer Pre-Net: Before processing in the LSTM layers, the inputs pass through a small feed-forward neural network called a Pre-Net, which helps to transform the inputs into a more useful representation for the LSTM layers.

Two LSTM Layers: These are the main recurrent layers of the decoder. They process the sequence step-by-step and output a new representation that captures the temporal structure of speech.

Location Sensitive Attention: This component helps the decoder focus on different parts of the encoder output at each step of the generation process. It's "location-sensitive" because it takes into account the decoder's previous focus points to inform its current focus, which is important for aligning the generated speech with the input text.

Linear Projection: These layers project the output of the LSTM layers into the Mel spectrogram space, creating a raw Mel spectrogram prediction for each time step.

Mel Spectrogram Linear Projection: The first Linear Projection block that connects directly to the LSTM layers in the Decoder transforms the LSTM output into a predicted Mel spectrogram. Each LSTM output represents a moment in time within the generated speech. This projection serves to

translate these outputs into a format that represents the frequency content of the audio signal at each time step, which is what a Mel spectrogram encapsulates.

Stop Token Linear Projection: The second Linear Projection block is responsible for predicting stop tokens. It projects the output of the LSTM layers into a binary decision for each frame, indicating whether the frame should be the last one in the generated Mel spectrogram (i.e., where the speech should stop). This is crucial for the model to be able to generate audio of variable length that corresponds appropriately to sentences of different lengths.

In essence, while both Linear Projection blocks utilize the output of the LSTM layers, they project this output into different spaces for different purposes: one for the continuous Mel spectrogram frames, and the other for the binary stop tokens. This allows the decoder to simultaneously predict what the speech should sound like at each time step (the Mel spectrogram) and when the speech should end (the stop tokens).

C**onv Layer Post-Net**: A small convolutional network that refines the raw Mel spectrogram prediction, improving the quality of the audio that will be synthesized.

3.1.3 WaveNet Vocoder

The refined Mel spectrogram is then passed to a WaveNet vocoder, which is a neural network model that generates the final waveform (audio signal). The vocoder translates the Mel spectrogram into an audio waveform that we can hear.

3.1.4 Stop Tokens

During decoding, the model also predicts stop tokens, which indicate when the generation of the Mel spectrogram frames should end. This is essential to determine the end of the spoken output.

3.1.5 Final Output

The final output is the Waveform Samples, which are the audio samples that can be played back to produce the synthesized speech.

3.2 Hyperparameter Settings

Hyperparameters are the configurable variables used to control the training process of a machine learning model. They are set before the training starts and can have a significant impact on the performance of the model. The adjustment of hyperparameters represents another level of customization. Hyperparameters are the knobs and dials of machine learning models, dictating the learning process's behavior. These settings are carefully selected to guide the learning process, potentially improving the model's ability to generalize from the training data and ultimately affecting the quality of the synthesized speech. Several hyperparameters were adjusted to optimize the model's learning and generation capabilities.

• **Attention Dropout Rate :** The attention dropout rate is a hyperparameter that helps in regularizing the model's training by randomly dropping parts of the data that the attention mechanism uses to generate the context vector. This is a form of dropout specifically applied to the attention mechanism within the model. By doing so, it encourages the model to learn robust representations without over-relying on specific parts of the input sequence, potentially improving the model's ability to generalize

and thus reducing the risk of overfitting. It's a balancing act; too much dropout can hinder learning, while too little can lead to overfitting.

• **Decoder Dropout Rate:** Similarly, the decoder dropout rate applies the dropout regularization technique to the decoder part of the network. The decoder is responsible for generating the Mel spectrogram frame by frame from the encoded input sequence. Applying dropout to the decoder helps prevent overfitting to the training data, encouraging the model to find more generalized solutions that perform better on unseen data. Just as with attention dropout, the decoder dropout rate must be carefully tuned to find the right amount of regularization.

• **Learning Rate:** The learning rate controls how much the model's weights are updated during training in response to the estimated error each time the model weights are updated. It is one of the most important hyperparameters and can greatly affect the model's performance and convergence speed. A learning rate that is too high may cause the model to converge too quickly to a suboptimal solution, while a learning rate that is too low can result in a long training process that could stall. Advanced techniques such as learning rate schedules or adaptive learning rates can be used to adjust the learning rate during training dynamically.

• **Batch Size:** The batch size specifies the number of training samples to be fed through the network before the model's internal parameters are updated. It is a compromise between the idealized stochastic gradient descent (where the batch size is 1) and full batch training (where the batch size is the entire dataset). Reducing the size of the batches used during training has a regularization impact and improves generalization. However, this may introduce more noise and lengthen the training process. Increasing the size of the batches used in training may provide more precise gradient estimates, but it may also need a larger amount of memory and perhaps result in faster convergence towards inferior minimum points.

3.3 Pronunciation and Language Handling

Accurate and natural-sounding pronunciation is just as crucial in voice synthesis systems to provide a lifelike and enticing listening experience. An especially new technique for influencing pronunciation in text-to-speech systems is to utilize the ARPAbet phoneme set to transform typical written text into intellectual transcript phonetics based on a dictionary. The primary purpose of this approach is to provide a unique and precise set of pronunciation instructions for the voice synthesis engine.

ARPAbet is a set of pronunciation transcription codes that are commonly utilized for voice and speech processing purposes. They are symbols or a combination of symbols that stand for each phoneme in the English language. Phonetic rendering is critical since it enables the exact meeting of the sounds that constitute spoken words to be appropriately determined rather than relying on the infrequent and inconsistent pronunciation based on the standard manuscript. For instance, a word such as “enough” may be pronounced in various ways depending on how it is spelt while offering the same transcription an ARPAbet symbol generates an exact reference to the pronunciation.

A dictionary-based pronunciation system uses a massive database. In this database, words are associated with short ARPAbet transcriptions that portray their pronunciation. While text is being spoken in the given TTS system, the dictionary is consulted in reverse to produce the transcription

for each word. This ensures that the synthesis engine definitively knows how to properly express each word, regardless of context or the spellings of the words surrounding it.

The advantages of using a pronunciation dictionary in speech synthesis are as follows:

• **Consistency:** pronunciation is consistent between different instances of the same word, making speech output more uniform and predictable.

• **Accuracy:** words that are either difficult to pronounce, or out-of-dictionary and thus easily mispronounced due to their irregular spelling and are pronounced regarding their contexts, are pronounced correctly. This can lead to an overall increase in the naturalness and intelligibility of a synthesized text.

• Flexibility: The correct pronunciation of new words or special knowledge such as medical terms, or new names, may be added to the dictionary. This flexibility allows the system to adapt to new vocabularies, even though such systems do not require retraining.

• Efficiency: Because text-to-phoneme conversion occurs before synthesis, the TTS engine may devote more computational attention to human-like vocoding. This simplifies the task of synthesis by removing the need to ascertain proper pronunciation from context.

3.4 Output Audio Processing using HiFi-GAN

In the context of text-to-speech (TTS) synthesis, advanced audio processing enables one to convert a simple text input into a voice that sounds almost natural. Moreover, depending on the level of its sophistication, the quality, degree of clarity, and the naturalness of such speech are achieved. In this case, we obtain the neural network models that synthesize speech from our input audio. Then, increase the audio quality using models such as HiFi-GAN, which follows with a super resolution mechanism to make it even more accurate. It achieves this by generating a Mel spectrogram, a representation of the spectrum of frequency of sound as they change over time from the text. This stage is critical since the Mel spectrogram adequately captures the dynamics and intonation present in the speech, being a link between raw audio and text and retaining the natural emotional and tonal flow of speech.

Upon generation of the Mel spectrogram, the next step is to convert the aforementioned frequency-time graph into waveforms that can be heard. This is the role that HiFi-GAN is designed to play. HiFi-GAN is a generative adversarial network specifically developed to synthesize high-quality audio from Mel spectrograms. It synthesizes audio signals with a high level of understanding, with smoother speech and fewer robotic-sounding audio bits. Its high fidelity without the grainy sound experienced with other TTS is as a result of the existing speech dataset from which it had been trained to learn the desired audio.

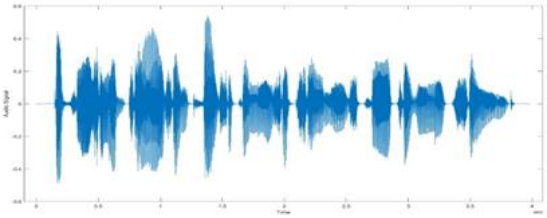
To top up the quality of the audio, there exist super-resolution techniques. Audio super-resolution is an accomplished computation technique that involves increasing the number of times the audio's analog to digital sampling was undertaken or improving the level of detail in the audio produced. In the instance of TTS, super-resolution can be applied where some sounds might be finer than desired, resulting in a sound that may not be clear. The super-resolution of sounds improves their quality experience.

At the top of the hierarchy are the main processes, including audio processing, where there are techniques such as Griffin-Lim synthesis, which is one of the approaches that convert a Mel spectrogram back into an audio waveform. Griffin-Lim synthesis tends to generate a certain amount of artifacts while it can improve the generation of audio from spectrograms in conjunction with models such as HiFi-GAN by iteratively approximating spectrograms. Moreover, audio signals go through filters and normalization to constrain the volume and pattern of the sound in terms of achieving the necessary audio based on the predetermined allowable audio standards.

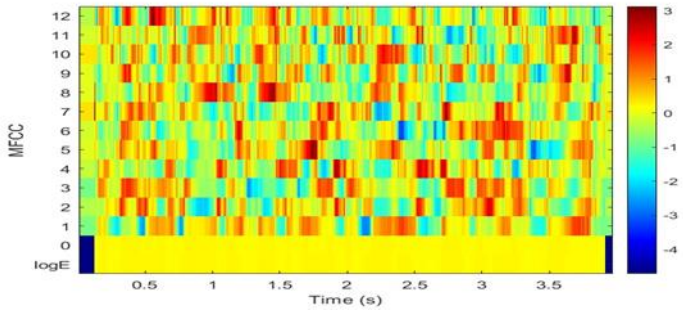
Results: This section reveals the results of simulations performed with the recalibrated voice conversion model above, responsible for upgrading the quality of speech transformation, with a specific focus on prosody, naturalness, and intonation on generated speech. The experiments implied the collection of a dataset made up of 1000 spoken English audio recordings. The audio data was equally spread among fifty male and female English speakers with varying accents. The model was trained within a Tesla V100 GPU throughout 250 epochs, with batch size as three. The quality was assessed using the Mel-Cepstral Distortion by means perception tests online.

The first component of the process is data collection, in which a diverse range of speech samples are required across genders, accents, and languages among others. From the collection, the audio speech is transcribed to text while concurrently segmenting the audio files so that each of the segments aligns with one textual equivalent. Concretely, the target audio needs to be segmented into a set of well-defined units one of them is sentences or phrases". The keyed transcriptions are then converted into the most appropriate format, which in this case will be Mel-spectrograms. These representations offer a visual inspection of the power spectrum of sound and time, manifesting the most suited feature set for speech synthesis. The spectrum of sample voices in the dataset are depicted in Figure 2.

Figure 2(a) represents the spectrum of Male Speech, and its Mel Frequency Cepstral Coefficients (MFCC) plot is shown in Figure 2(b). Similarly, Figure 3(a) and 3(b) represents the spectrum and MFCC plot of Female speech. MFCC are widely used in the processing of speech and audio and used for such tasks as speech recognition and music analysis. In an MFCC plot, the coefficients are plotted against time, showing how the sound's spectral properties change over time. The first axis on this plot is a function of time, while the second is the values of the individual coefficients. The lowest coefficient is typically located at the bottom, while increasing coefficients are on top. A few other things to consider are as follows: The X-axis is the Time Axis: This represents the movement of time as the sound signal is analyzed. The Coefficient Index Axis refers to the Y-axis: For this axis, the indices of the cepstral coefficients are often displayed. Color or Intensity: The third property often displayed is the color or intensity of the value of the MFCCs summaries.

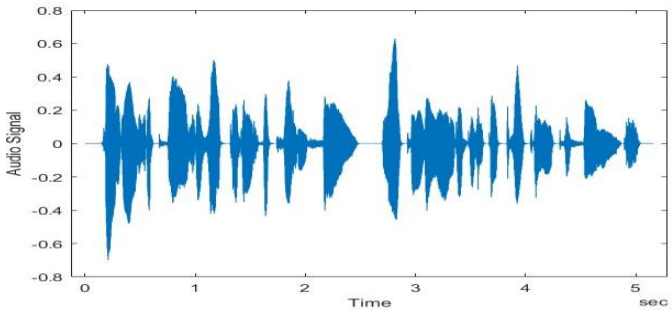


(a) Spectrum of Speech Signal-1

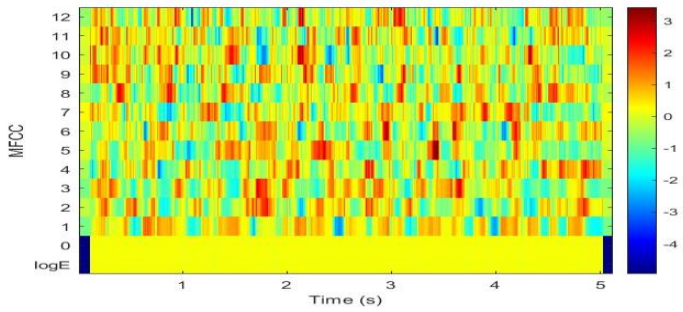


(b) MFCC of the Speech Signal -1.

Figure 2: Male Speech Text: "The Fusion Of Jaz And Classical Music Joneris Creates A Unique Sound"



(a) Spectrum of Speech Signal-2.



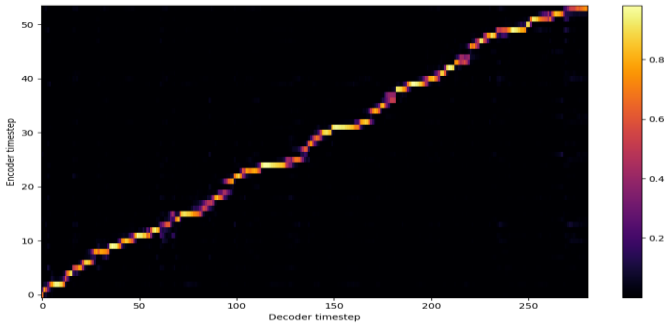
(b) MFCC of Speech Signal-2

Figure 3: Female Speech Text: Thunderous Applause Erupted As The Curtain Fell Marking The End Of The Spectacular Performance

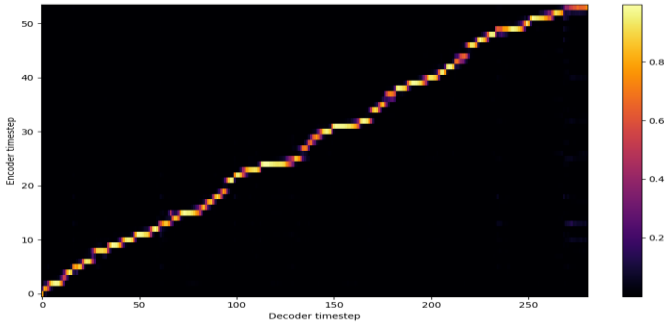
It is used to visualize the dimension of an onset per time frame. It includes color-coded autoscale and optional horizontal bar. The scale is adjusted to match the sequence and is displayed with the data.

After the data is prepared, training of the proposed Tacotron 2 model starts. The model architecture is usually divided into two distinct sections: the encoder and the decoder. An encoder is trained on input text to convert characters into an isotropous, high-dimensional embedding. This representation captures the nuanced language structure and context required to produce speech. On the other hand, the decoder is responsible for converting this into audio data or output. It predicts the Mel-

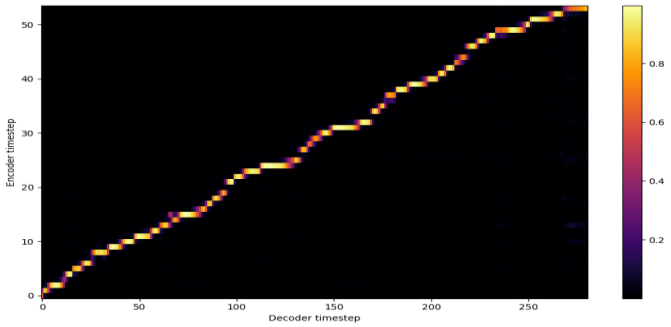
spectrogram frame by frame, the graphical form of audio's spectral properties. More precisely, training involves learning the model's parameters so that the Mel-spectrogram it outputs, when passed to a proper decoder, resembles the ground truth Mel-spectrogram as closely as possible, and is done by optimization using a specially-designed loss function.



(a) Training at Epoch -0 Validation loss: 1.007171



(b) Training at Epoch-22 Validation loss: 0.212975



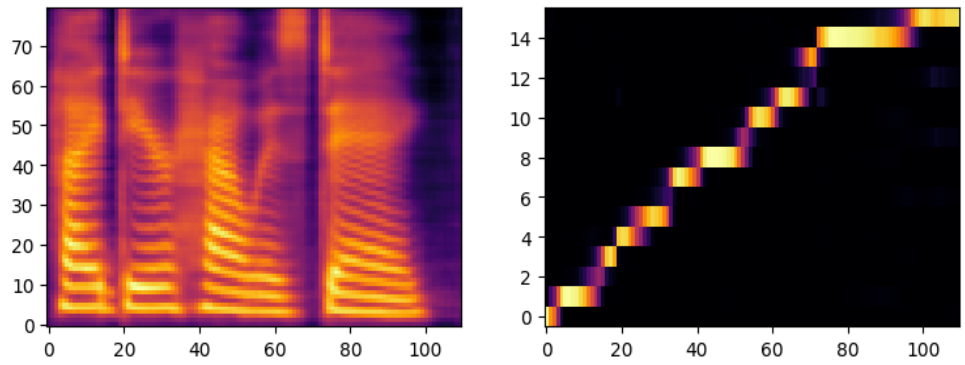
(c) Training at Epoch -45 Validation loss: 0.212975

Figure 4: Encoder vs Decoder Time Step plot at Different Epochs

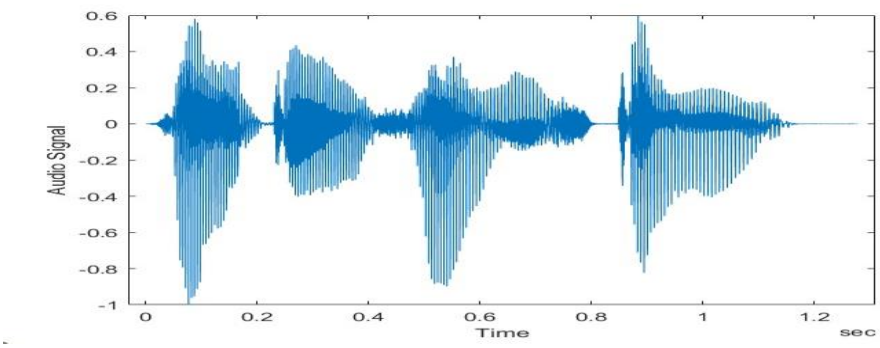
Figure 5 represents a heat map representing the dependence of encoder timesteps on decoder timesteps in a sequence-to-sequence model, which is typically applied in machine learning for language translation, text summarization or speech recognition. More specifically, an encoder reads the input data, for example, a sentence in the source language, gradually, one timestep after another, so as to create a representation of it. After that, the decoder produces the output data sequentially, timestep by timestep, and this process may depend on the model of the encoder. In Figure 5, the

presented heat map is a visualization of the attention weights or alignments between each encoder timestep on the y-axis and each decoder timestep on the x-axis. The color scale on the right shows the strength of the attention or alignment, and warmer colors indicate higher weights in the attention vector while cooler colors indicate lower weights. The diagonal pattern seen in the figure indicates that each decoder timestep seems to pay significant attention mainly to the corresponding encoder timestep or encoder timesteps close by in the input sequence. This pattern is evidence of a monotonic alignment between the input sequence and the output sequence. It is most common in topics such as machine translation where the input and output sequence's individual sentences have a comparable structure.

After training the speech synthesis model based on Proposed Tacotron 2, one also needs to train its voice conversion model. This model aims to fine-tune the speech creation process so that it resembles the voice of the target one more closely. The voice of the speech can be altered and include changes in pitch and timbre and also rhythm. This process calls for additional data from the target voice, and it might also use voice embedding and style transfer to learn intensively the unique properties of the target voice and imitate them. The next step is fine-tuning the model. This procedure fine-tunes the model to be more sensitive to the subtleties of target voice characteristics, which guarantees that while the synthesized voice is made to sound natural, it also sounds as close as possible to the intended target output voice. Additionally, optimization can involve some model refinement; for example, the model can be modified to generate voices faster or produce more articulate synthesized speech.



(a) Heatmap and Encoder vs Decoder Time Step plot of Output wave file



(b) Spectrum of Output wave file

Figure 5: Visual Representation of Converted Speech wave file

Statistical analysis was performed for results validation. In summary, the proposed novel Tacotron 2 model, in its enhanced form, yield remarkable enhancements in terms of speech quality and conversion accuracy. These enhancements results from employing novel model architectures with prosody prediction and voice modulation. These research results, thus, validate the hypothesis that the implementation of enhancements to the Tacotron 2 model will considerably enhance the naturalness and conversion accuracy. Hence, they demonstrate that advanced neural architectures hold tremendous potential in the domain of speech synthesis and enable the development of synthetic speech for highly sophisticated and expressive models.

Conclusions:

The paper presents a comprehensive study on the application of deep learning techniques in the field of voice conversion, with an emphasis on improving the authenticity and functionality of the converted voices. Our proposed model integrates Automatic Speech Recognition with a sophisticated voice conversion framework to address the prevalent challenges in the domain. The experimental results highlight the model's efficacy in producing natural-sounding speech while maintaining the unique characteristics of the target voice. Significant advancements were observed in the quality of voice conversion, attributed to the robust feature extraction capabilities of the encoder and the precise generation qualities of the decoder. The application of bidirectional LSTMs and WaveNet vocoder further refined the audio output, ensuring high fidelity and clarity. Overall, the proposed model not only demonstrates superior performance in voice conversion but also offers new directions for future research in enhancing speech synthesis technologies.

References:

[1] Yoo, In-Chul, Keonnyeong Lee, Seonggyun Leem, Hyunwoo Oh, Bonggu Ko, and Dongsuk Yook. "Speaker anonymization for personal information protection using voice conversion techniques." IEEE Access 8 (2020): 198637-198645.

[2] Polyak, Adam, Lior Wolf, Yossi Adi, and Yaniv Taigman. "Unsupervised cross-domain singing voice conversion." arXiv preprint arXiv:2008.02830 (2020).

[3] Liu, Songxiang, Yuewen Cao, Shiyin Kang, Na Hu, Xunying Liu, Dan Su, Dong Yu, and Helen Meng. "Transferring source style in non-parallel voice conversion." arXiv preprint arXiv:2005.09178 (2020).

[4] Deng, Chengqi, Chengzhu Yu, Heng Lu, Chao Weng, and Dong Yu. "Pitchnet: Unsupervised singing voice conversion with pitch adversarial network." In ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 7749-7753. IEEE, 2020.

[5] Choi, Heejin, and Minsoo Hahn. "Sequence-to-sequence emotional voice conversion with strength control." IEEE Access 9 (2021): 42674-42687.

[6] Zhang, Liqiang, Chengzhu Yu, Heng Lu, Chao Weng, Chunlei Zhang, Yusong Wu, Xiang Xie, Zijin Li, and Dong Yu. "Durian-sc: Duration informed attention network based singing voice conversion system." arXiv preprint arXiv:2008.03009 (2020).

[7] Baas, Matthew, and Herman Kamper. "Voice Conversion for Stuttered Speech, Instruments, Unseen Languages and Textually Described Voices." In Southern African Conference for Artificial Intelligence Research, pp. 136-150. Cham: Springer Nature Switzerland, 2023.

[8] Zheng, Wei-Zhong, Ji-Yan Han, Chen-Kai Lee, Yu-Yi Lin, Shu-Han Chang, and Ying-Hui Lai. "Phonetic posteriorgram-based voice conversion system to improve speech intelligibility of dysarthric patients." Computer Methods and Programs in Biomedicine 215 (2022): 106602.

[9] Sisman, Berrak, Junichi Yamagishi, Simon King, and Haizhou Li. "An overview of voice conversion and its challenges: From statistical modeling to deep learning." IEEE/ACM Transactions on Audio, Speech, and Language Processing 29 (2020): 132-157.

[10] Zhao, Yi, Wen-Chin Huang, Xiaohai Tian, Junichi Yamagishi, Rohan Kumar Das, Tomi Kinnunen, Zhenhua Ling, and Tomoki Toda. "Voice conversion challenge 2020: Intra-lingual semi-parallel and cross-lingual voice conversion." arXiv preprint arXiv:2008.12527 (2020).

[11] Zhou, Kun, Berrak Sisman, Rui Liu, and Haizhou Li. "Emotional voice conversion: Theory, databases and ESD." Speech Communication 137 (2022): 1-18.

[12] Wang, Disong, Liqun Deng, Yu Ting Yeung, Xiao Chen, Xunying Liu, and Helen Meng. "Vqmivc: Vector quantization and mutual information-based unsupervised speech representation disentanglement for one-shot voice conversion." arXiv preprint arXiv:2106.10132 (2021).

[13] Li, Yinghao Aaron, Ali Zare, and Nima Mesgarani. "Starganv2-vc: A diverse, unsupervised, non-parallel framework for natural-sounding voice conversion." arXiv preprint arXiv:2107.10394 (2021).

[14] Liu, Songxiang, Yuewen Cao, Disong Wang, Xixin Wu, Xunying Liu, and Helen Meng. "Any-to-many voice conversion with location-relative sequence-to-sequence modeling." IEEE/ACM Transactions on Audio, Speech, and Language Processing 29 (2021): 1717-1728.

[15] Huang, Wen-Chin, Tomoki Hayashi, Yi-Chiao Wu, Hirokazu Kameoka, and Tomoki Toda. "Pretraining techniques for sequence-to-sequence voice conversion." IEEE/ACM Transactions on Audio, Speech, and Language Processing 29 (2021): 745-755.

[16] Zhou, Kun, Berrak Sisman, Rajib Rana, Björn W. Schuller, and Haizhou Li. "Emotion intensity and its control for emotional voice conversion." IEEE Transactions on Affective Computing 14, no. 1 (2022): 31-48.