Style-conditioned Text-to-Audio Model
Project Report

Team 3

Sreenidhi Hayagreevan - 018195489
Bhavana Meravanige Veerappa -017852458
Shivani Beri - 018205018
Laxmi Thrishitha Kalvakota - 017605640
Janani Kripa Manoharan - 016721159

Masters in Data Analytics, San Jose State University
DATA266 - Generative Models
Prof. Dr. Simon Shim
December 08, 2025

Abstract

On the current paper, a fully customized Two-stage Text-to-Speech (TTS) was developed based on PyTorch and various supporting libraries, such as Accelerate, Weights and Biases, NumPy/SciPy, and Librosa/torchaudio. It uses a sequence-to-sequence acoustic model based on Tacotron 2, in the first stage. The encoder converts textual input to a linguistic representation and an autoregressive decoder creates mel-spectrogram frames. In order to produce expressive speech, a discrete style-conditioning module, which is executed as a Reference Encoder or Global Style Tokens (GST), is added. The module permits the model to train and regulate the expressive speaking styles to include angry, whisper and excited as in the Expresso dataset but utilize LJSpeech to maintain a natural acoustic quality. The resulting style embedding is combined with the encoder output, which allows a fine-grained control of prosody and emotional expression by using style tags.

The second step involves the high-fidelity HiFi-GAN neural vocoder to encode mel-spectrograms into audio waveforms. HiFi-GAN is initially trained on ground-truth mel-spectrograms of LJSpeech and then trained on mel-spectrograms generated by Tacotron 2. This refinement step observes the domain-mismatch between ground-truth and the predicted mels, enhancing the naturalness and resulting in a more faithful final synthesized audio to the desired expressive style.

The last system will be tested strictly on the quality of speech, intelligibility, and style expression consistency. It will involve both objective and subjective evaluation measures, such as automatic style-classification accuracy, mel-based similarity measures, and predicted or human-rated Mean Opinion Score (MOS).

Keywords: text-to-speech, Tacotron2, HiFi-GAN, neural vocoder, mel-spectrogram, generative adversarial network

1. Introduction

1.1 Background and Motivation

Text-to-speech (TTS) synthesis is a basic challenge within artificial intelligence with regards to transforming text on a computer screen to audio that is similar to natural human speech. TTS synthesis began with a process known as concatenative synthesis whereby the system joined pre-recorded human speech to produce the desired audio. However, with the advent of neural networks for TTS synthesis, the field was transformed with significant prospects for text-to-speech synthesis with respect to innovation and application. Some areas for text-to-speech synthesis include accessibility and human-computer interaction.

The task of TTS synthesis can be broken down into several sub-tasks: text normalization and phonetic analysis, prosody modeling (rhythm, stresses, and pitch), and audio synthesis. In traditional approaches to TTS synthesis, these sub-tasks would be computed sequentially. However, with neural network approaches, these sub-tasks can be accomplished more holistically and end-to-end. However, audio synthesis from text is still a challenging computational task. To this end, neural network approaches now split tasks across multiple stages: producing hidden representations (mel-spectrograms), then audio.

1.2 Problem Statement

“To what extent can we use neural TTS to make natural and human-like speech from text?”To solve the problem:

1. Build Tacotron2, a sequence-to-sequence model for the autoregressive generation of mel-spectrograms given text, with attention that is sensitive to location.
2. Compare traditional audio synthesis and neural vocoding to understand the quality differences between classic signal processing techniques and modern deep learning approaches.
3. Use HiFi-GAN, a Generative Adversarial Network-based neural vocoder, to generate audio waveforms from mel-spectrograms.
4. Assess the effect of domain adaptation by fine-tuning the vocoder on spectrograms generated by the TTS model, instead of using ground-truth data.
5. Quantify the improvements in the audio quality with objective metrics (UTMOS) that predict how humans would perceive the sound.

1.3 Project Scope and Contributions

This particular model is centered around the task of single speaker voice synthesis on the LJSpeech dataset: a widely used benchmark corpus containing around 25 hours of well-recorded audiobook readings. This project offers a full pipeline from text to speech beginning with raw text as the input and leading all the way to audio signal synthesis. Throughout this process, utmost care is taken with respect to audio quality and the improvements achievable with state-of-the-art vocoding approaches and domain adaptation.

A crucial aspect of this research is the building of a reliable Tacotron2 model with the ability to accomplish text-to-spectrogram alignment with the help of location-sensitive attention. Along with this another important concept is that the comparative analysis between the Tacotron2 reconstruction and the neural vocoding with emphasis on the great improvements is brought by neural vocoding. In continued efforts to increase the system's efficiency and quality, the inclusion of the HiFi-GAN model is employed with the intention of generating high-quality and more realistic audio with the help of multi-scale and multi-period discriminators. Another important analysis is carried out with the intention to close the domain gap between GT and predicted spectrograms with the help of finetuning experiments. Lastly, system efficiency is tested with the help of automatic quantitative analysis.

1.4 Literature Review and Related Work

The TTS synthesis process has gone through many technological eras. In the earliest systems based on formant synthesis, a mathematical simulation modeled the human vocal tract to produce speech sounds. Although efficient from a computational standpoint, they produced speech with a robotic and unnatural character. In the next era represented by the dominant form from the 1990s till the early 2010s known as concatenative text-to-speech synthesis, the speech was produced by stringing together clips taken from huge databases (Hunt and Black, 1996).

Statistical parametric synthesis was a middle ground between traditional methods and neural speech synthesis. Specifically, HMM-based solutions (Zen et al., 2009) used statistical models for acoustic feature representations. However, this allowed more seamless parameter interpolation and more natural prosody representation. Nonetheless, such synthesized speech was perceived as muffled. The emergence of neural TTS solutions represented a paradigm shift. WaveNet (van den Oord et al., 2016) showed that raw audio can be produced at unprecedented levels of quality by deep neural networks. However, the autoregressive audio sample-based method implemented by WaveNet was highly computational and thus inefficient for real-time solutions.

To this end, Tacotron (Wang et al., 2017) was developed with a two-stage system: mel-spectral-converted text and audio production with Vocoderization. Additionally, Tacotron was initially based on a seq2seq framework with attention networks inspired from the field of machine translation. In this respect, its successor Tacotron 2 (Shen et al., 2018) further optimized the model with the intention of enhancing attention with a position-sensitive attention network that enabled monotonically aligned attention between text and audio necessary for TTS tasks so that each text token is uttered once.

The Tacotron2 model is composed of an encoder for text input, attention for text and audio alignment learning, and an autoregressive decoder for generating mel-spectrograms one at a time. The model is trained with teacher-forcing for generating outputs, which uses the actual previous output to generate the next output during training time. However, this is not the case during testing time.

Although Tacotron2 is able to successfully produce mel-spectrograms, their conversion to audio waveforms is a highly important task that is yet to be accomplished. The WaveNet

vocoders (van den Oord et al., 2016) were conditioned on mel-spectral maps and produced high-quality audio with considerable computational costs. WaveGlow (Prenger et al., 2019) employed normalizing flows for efficient parallel synthesis. However, this method was still under significant memory constraints.

WaveRNN (Kalchbrenner et al., 2018) achieved efficient synthesis by employing innovative recurrent neural networks. HiFi-GAN (Kong et al., 2020): This is a highly advanced neural vocoding technique based on the use of GANs for obtaining superior quality and efficient generation. Key innovations include multi-scale discriminators that evaluate audio quality at different temporal resolutions, multi-period discriminators that analyze periodic patterns in speech signals, feature matching losses that improve training stability as well as mel-spectrogram reconstruction loss ensuring the generated audio matches the input spectrogram. HiFi-GAN achieves real-time speed and generates audio with quality on par with WaveNet. This model is suitable for production TTS tasks.

In multi-stage TTS models, the domain gap between the train and test conditions is a critical issue. Typically, vocoders such as HiFi-GAN learn on the ground truth mel-spectrogram computed from real audio files. However, during testing conditions, vocoders take the computed spectrograms from other models such as Tacotron2. Chen et al. (2020) showed that fine tuning vocoders on the generated spectrograms can further improve the quality of synthesis by adapting to the specific characteristics of the spectrogram generator. This domain adaptation technique has become a norm for multi-stage TTS services.

Traditionally, TTS quality can be assessed by Mean Opinion Score (MOS) tests, which assess the naturalness of speech by listeners on a scale from 1 to 5. However, MOS testing can be costly and time-consuming. In more recent research efforts, various attempts for automatic quality estimation methods that automatically predict MOS values were realized. The method UTMOS (Saeki et al., 2022) is based on pre-trained speech features for the estimation of MOS values.

Data Exploration and Processing

2.1 Dataset

The LJSpeech 1.1 database is one of the most common single-speaker English speech corpora that comprises 13,100 short audio files amounting to about 24 hours of recorded speech. The recordings were made by having a woman read passages in seven non-fiction books in the public domain and the recording was made such that there was clear articulated narration with a consistent vocal quality. Each audio file is delivered as high-quality WAV file of 22.05 kHz, 16-bit, a text transcript of speech is also delivered in the form of a metadata file matching utterance IDs to their text. The data is specifically well adopted in developing neural text-to-speech (TTS) models because of the clean audio, uniformity of microphone properties, and alignment-fitting format. LJSpeech is now widely used as a standard bench on sequence-to-sequence and neural vocoders due to its single-speaker property, which reduces variability and allows multi-controlled experience to be performed in acoustic models and speech synthesis.

audio audio · duration (s) 0.59 13.2	audio_len float32 0.59 13.2	text string · lengths 43 131	raw_text list · lengths 4 4
0:00 / 0:04	4.145875	A females saying "Two days ago…	["Emotion: Neutral", "Gender: females", "Text: Two days ago Jeanne learned where her…
0:00 / 0:03	3.951562	A male saying "And here's another…	["Emotion: Sleepy", "Gender: male", "Text: And here's another idea.", "Filename:…
0:00 / 0:04	4.6835	A male saying "And MacDougall was…	["Emotion: Amused", "Gender: male", "Text: And MacDougall was beyond the trail, with…
0:00 / 0:03	3.712063	A female saying "He obeyed the…	["Emotion: Amused", "Gender: female", "Text: He obeyed the pressure of her hand.",…

2.2 Data Preprocessing

The system was trained using the LJSpeechdataset, a widely adopted single-speaker English corpus that contains 13,100 audio clips approximately 25 total hours of speech 22.05 kHz sampling rate Text transcription is aligned with each audio clip Clean, studio-quality narration ideal for TTS research because the raw dataset does not include predefined training and validation sets, a custom metadata preparation script, prep_splits.py that was used to automatically divide the samples into: train_metadata.csv.

The dataset contains around 13,100 audio clips, approximately 25 total hours of speech and also 22.05 kHz sampling rate. The Text transcriptions are aligned with each audio clip which has a clean, studio-quality narration ideal for TTS research. However, the raw dataset does not include predefined training and validation sets or a custom metadata preparation script, prep_splits.py that was used to automatically divide the samples into train_metadata.csv which is primary data which was used for learning and also there is test_metadata.csv, which is a subset for validation and evaluation.

The script performs several preprocessing steps, such as loading and cleans the original metadata.csv and also removes entries with missing normalized transcripts
This generates full .wav paths, which computes durations using librosa. Then the splits of the dataset are reproducible using a fixed random seed, and it sorts training samples by duration to reduce padding. This setup ensures consistent evaluation and also efficient training, which prevents model overfitting by ensuring that no validation examples appear in the training set.

The Preprocessing mainly includes steps as follows:
Before training the Tacotron2 model and raw speech waveforms were transformed into mel spectrograms, the acoustic representation that the decoder learns to generate. The preprocessing uses a Short-Time Fourier Transform (STFT) and mel-scaling with the following parameters: n_fft=1024,Window size = 1024 (Hann window),Hop size = 256,80 mel channels,fmin = 0, fmax = 8000.

These values align with the original Tacotron2 architecture and match the 22.05 kHz sampling rate of LJSpeech. After STFT and mel projection, spectrogram values were normalized to a [-4, 4] range, following the implementation guidance found in the project. Zero-centered normalization significantly improves model stability during training. This ensures that the

autoregressive decoder operates within a predictable numerical range which also enhances convergence and reduces training variance.

2.2.1 Text Preprocessing Pipeline

The text preprocessing pipeline prepares raw transcripts for model input. Firstly, character-level tokenization occurs: transcripts are converted to a series of character indexes based on a character-level vocabulary containing uppercase and lowercase letters, numbers, special character representations such as space, out-of-sequence, end-of-sequence, and padding. Secondly, linguistic formatting occurs with text where numerals are converted to words (e.g., "1984" becomes "nineteen eighty-four"), and abbreviations are also properly treated. Special character representations are properly managed. Finally, a preprocessing task that occurs because transcripts are of variable lengths is sequence padding. This ensures all transcripts in a batch are of a similar length for efficient tensor manipulation.

2.2.2 Audio Feature Extraction

The audio processing network transforms waveforms into mel-spectral representations for effective training. The pre-processing techniques involve pre-emphasis filtering. This involved a high-pass filter with a coefficient $\alpha = 0.97$. The filtering gives amplified frequencies. This ensures a balanced audio spectrum. Next would be the STFT. This entails dividing a signal with overlapping frames. This uses the FFT size of 1024 samples. A Hann window of 1024 samples would be used. This uses a hop size of 256 samples. This gives a sampling of 11.6 ms. The sampling rate would be 22,050 Hz. This would give a 75% overlap. This would give a linear frequency spectrogram. This converts this spectrogram to mel. This used 80 mel bins. This ranged between 0 Hz and 8,000 Hz. A mel-spectral would be more aligned with human listening. This expansion used a transformation. This gave a formula of log10(magnitude) = 20log10 (amplitude). This scaled values to decibels. This scaled values between -4 and 4. This gave a range with a center of 0 instead of a possible range of -100 to 0. This would facilitate better gradient flow.

2.2.3 Data Splitting Strategy

A split is made in the data for purposes of training and validation. The training data holds about 12,500 samples, which constitute about 95% of the total data. The validation data holds about 600 samples, which constitute about 5%. Importantly, this split is made such that validation samples are spread out evenly throughout this data and are not clumped towards the end based on sessions of recordings or themes.

2.2.4 Data Augmentation Factors

As speech-to-text systems seek to maintain speaker identity and pitch consistency, methods such as aggressive data augmentation are discouraged. In this particular code snippet example, data augmentation only occurs while training the model with HiFi-GAN. Here, random audio samples with a 8,192 sample duration are taken out of total recordings for diversity

without changing any speaker properties. Pitch shift, time stretch, and noise insertion methods are not used. This ensures that training speech samples adhere to ground truth recordings.

3 Methodology

In this work, we use a two-stage neural text-to-speech (TTS) system comprising a Tacotron2 acoustic model and a HiFiGAN neural vocoder. The training of Tacotron2 on LJSpeech corpus consists of normalized text transcripts combined with mel-spectrograms calculated on the template audio (22.05 kHz). The location-sensitive attention encoder-decoder architecture is an autoregressive mel-spectrogram generator that learns to predict mel-spectrograms on a character-level input of text. The second stage will be to train a HiFiGAN vocoder so that mel-spectrograms can be translated into high-fidelity waveforms. The initial training of HiFiGAN is done using ground-truth mel-spectrograms obtained directly by decoding the audio of LJSpeech recordings to learn a solid mapping between spectral images and audio over time. In order to alleviate the distribution mismatch of ground-truth spectrograms and those of Tacotron2, we then finetune the pretrained HiFiGAN model by using Tacotron2-predicted mels as inputs and original waveforms as training targets. This two-stage approach; pretraining on clean ground-truth data and finetuning on model-generated spectrograms; yields considerable domain shift reductions and also leads to higher naturalness and fidelity of the resulting synthesized speech. The related code can be found at https://github.com/SreenidhiHayagreevan/Text-to-speech-using-GenAI.

3.1 Tacotron2

Tacotron2 (Shen et al., 2018) was chosen as the text-to-spectrogram model because of its proven architecture and outstanding results achieved in various studies about text-to-speech transformation. This model has already been validated numerous times in a wide number of datasets for different languages. One of its major advantages resides in its attention mechanism with a focus on the current position in the input text. This mechanism adds more flexibility to simple attention weights and ensures that a token-wise sequential processing of the input text occurs. This feature is vital for creating natural speech.The model uses an autoregressive generation method with teacher forcing during training. The model receives ground-truth spectrogram frames for previous time steps. This speeds up convergence.The model generates spectrograms based on its predictions during inference. This allows for variable-length spectrogram generation. Tacotron2 model primarily generates the mel-spectrograms instead of audio. This makes training easier because it operates at a more abstract level but retains critical features for speech synthesis.

Additionally, other methods such as FastSpeech and FastSpeech2 were considered for this study. They are non-autoregressive models that support parallel generation of spectrograms. They use external alignment techniques for training. Glow-TTS and VITS were other methods considered for this study. They are a flow-based model and end-to-end model that produces audio directly from texts. For this study, Tacotron2 was a better option because of its performance, understandability, and maturity state among other techniques in text-to-speech.

3.1.1 Tacotron2 Architecture

The Tacotron2 used in the present work is a sequence-to-sequence model that comprises an encoder, a location-sensitive attention mechanism, an autoregressive decoder, a convolutional post-net, and a stop-token prediction head. The model displays mel-spectrogram frames based on textual input character level input with a configuration that is based on the initial Tacotron2 model.

Encoder

The input text is first transformed into integer token sequences which are mapped to 512-dimensional character embeddings (character_embed_dim = 512). The sequence is stacked and converted into 512 filters (encoder_embed_dim = 512) using 3 1-D convolutional layers (encoder_n_convolutions = 3) and 5 as a kernel size (encoder_kernel_size = 5). To ensure local contextual patterns and eliminate the possibility of overfitting, both convolutions apply batch normalization, ReLU, and dropout (encoder_dropout_p = 0.5). This output is further inputted into a 256 unit per direction bi-directional LSTM (512 total) where the encoder outputs encode long-range dependencies in the input sequence. Processing padded tokens is prevented (packed sequences), and it is necessary to sort the elements of the batch by length.

Location-Sensitive Attention

The model employs a location sensitive attention, which is achieved by a learned composition between content-based features and cumulative temporal attention. Historical and cumulative attention weights are stacked and run through a location convolution (attention location n filters =32, kernel size =31) to identify monotonic alignment patterns. A combination of these features, the decoder hidden state and a projective representation of encoder outputs is what is used to compute alignment energies and yield an attention distribution over encoder states. This process imposes a progressive alignment which is befitting to TTS and which is essential in the production of intonable audio.

Decoder

The decoder works in an autoregressive manner. At every time the last mel frame (or ground-truth mel when teacher forcing) is fed through a prenet (decoder_prenet_depth = 2) of two fully connected 256-unit (decoder_prenet_dim = 256) dropout (decoder_prenet_dropout_p = 0.5) networks. The prenet output is added to the attention context vector and inputted in a two-layer LSTM with 1024 hidden units (decoder_embed_dim = 1024). An 80-channel mel-spectrogram frame (num channels = 80) is used to project the outputs of an LSTM into the projection, and a separate projection is used to predict a stop token, which the generation has ended. The decoder has a dropout (decoder_dropout_p = 0.1) used to stabilize training.

This implementation (in contrast to certain versions of Tacotron2) applies a reduction factor of 1, i.e. predicts a mel frame per decoding step, which makes it more stable at the risk of slower prediction.

Post-Net

The model also has a convolutional post-net with 512 filters (decoder_postnet_n_filters = 512), 5 convolutional layers (decoder_postnet_num_convs = 5) with a 5-1 kernel size to filter the crude mel predictions. The tanh activation and dropout are applied to the first four layers, and the last layer results in a residual correction to the decoder output, which produces the final prediction of mel-spectrogram. Post-net is necessary to enhance spectral detail especially to harmonics and high-frequency detail.

Stop-Token Head

The model is used in decoding to predict a scalar stop probability at each timestep. It uses a linear projection head with the stop_proj which is a binary classifier to decide when mel generation should stop. This enables prediction of dynamic-length spectrograms without the use of special end-of-sequence symbols. Binary cross-entropy is used to compute the stop-token loss.

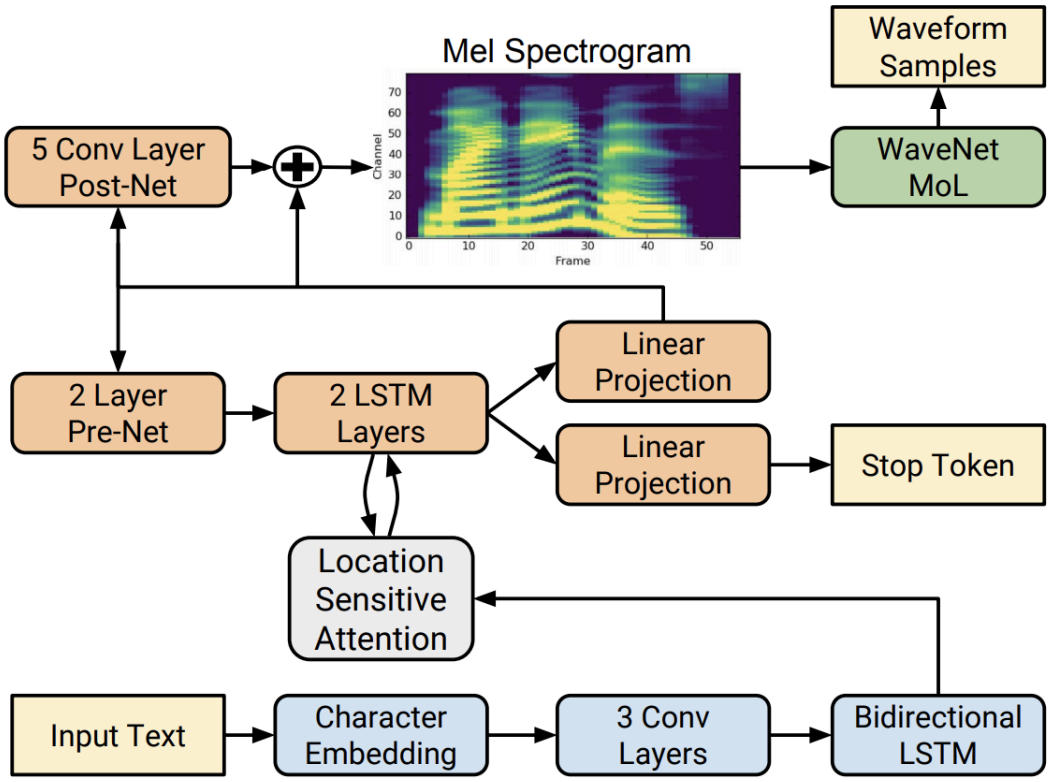
Training Configuration

Tacotron2 is trained on top of Adam optimizer with initial learning rate of 0.001, which may decays to 1e-5. Losses include:

- Mel reconstruction loss (MSE prediction vs. ground-truth mel),
- Refined mel loss (MSE after post-net),
- Stop-token BCE loss.

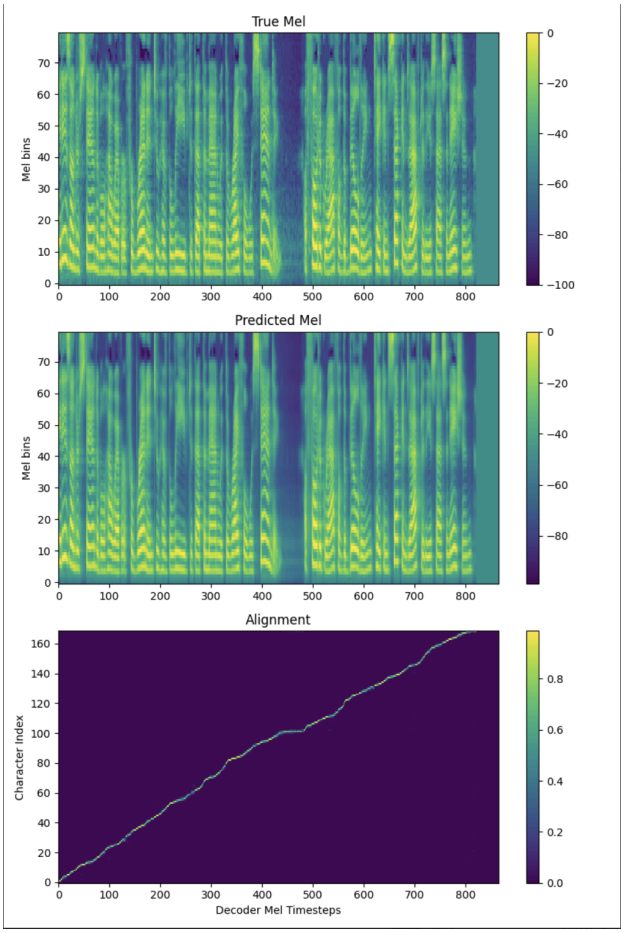
Gradient clipping (max norm = 1.0) is used in order to stabilize. Training uses HuggingFace Accelerate to train with multi-GPU and mixed precision. Batches are built on the basis of a length-sensitive sampler to reduce padding. The model is trained for 75 epochs and in the process, alignment behavior is monitored closely.

At epoch 10, the attention mechanism will start creating a weak diagonal alignment pattern. With epoch 20, there is a sharp and stable diagonal, and successful mapping of the text positions and mel timesteps is achieved. Training thereafter to epoch 75 produced sharp alignments and spectrogram predictions of high fidelity.

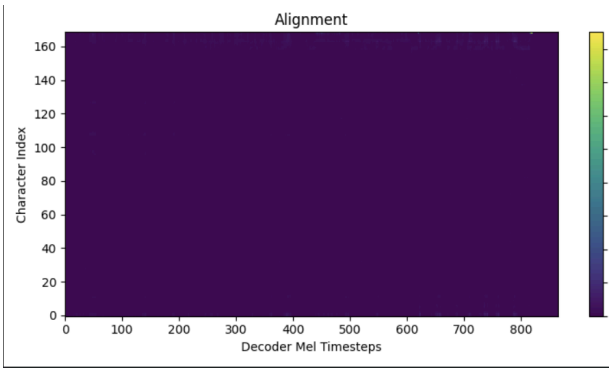


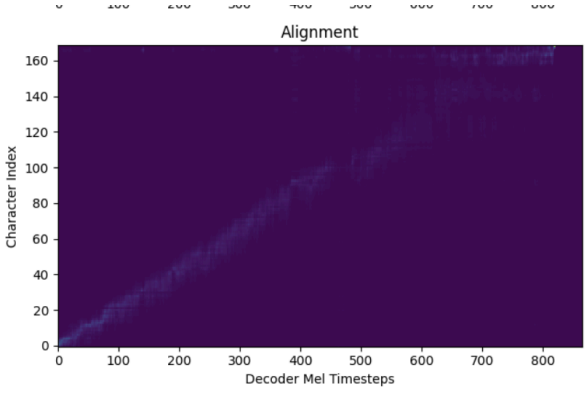
3.1.2 Tacotron2 Training Behavior

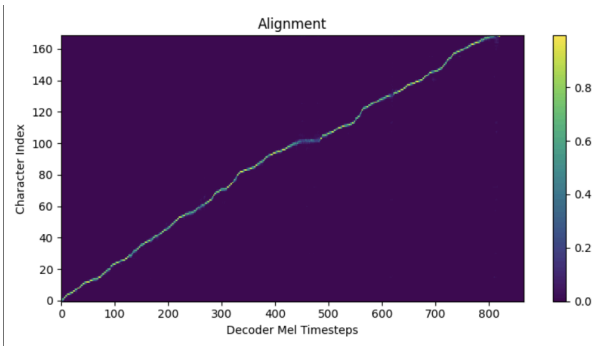
The tacotron2/train_taco.py script gives a full training setup of Tacotron2 with distributed data parallelism and automatic mixed precision with HuggingFace Accelerate. The script loads preprocessed LJSpeech metadata and does on-the-fly generation of mel-spectrograms and prepares minibatches with a length-aware batch sampler and a custom collator to reduce the overhead on padding and ensure efficiency. Tacotron2 architecture is adjustable, which means that it is possible to adjust the encoder and decoder size, prenet and postnet parameters, and attention-related hyperparameters. The model makes predictions in the form of mel-spectrogram frames and stop tokens, which are optimized together by a combination of mel MSE loss, postnet refined mel MSE loss and binary cross-entropy loss on the prediction of stop-tokens during training. The training loop consists of gradient clipping to achieve numerical stability, a multi-gpu optimization, optional learning rate decay and periodic logging, to monitor the progress of a training in real time. Validation is also done periodically by the script, producing diagnostic visualizations of forecasted mel-spectrograms and attention alignments. During training, there are saved checkpoints, and finally, a downstream inference and vocoder processing checkpoint.

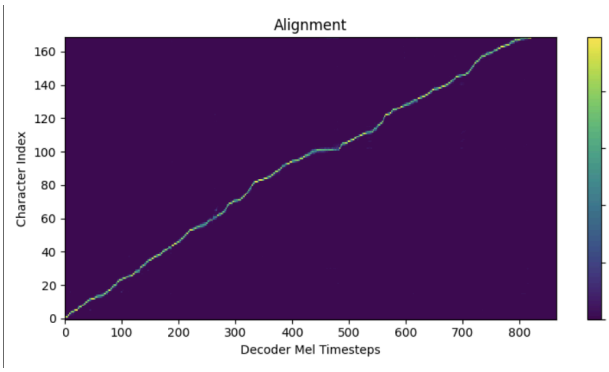


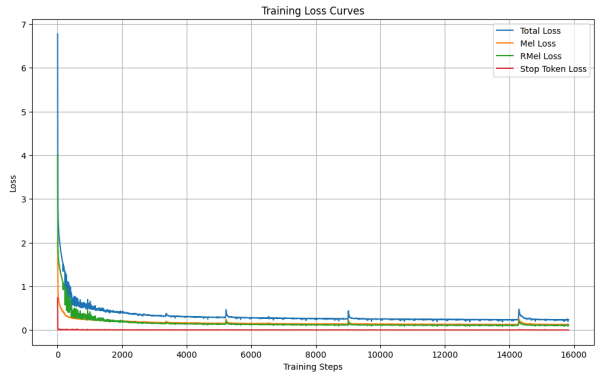
One of the most important signs of the Tacotron2 training success is the establishment of the stable pattern of attention alignment. At the start of training (at approximately epoch 10) the attention map usually assumes a hazy diagonal form, as the model has already discovered a monotonic relationship between the text positions and the mel-spectrogram timesteps. At around epoch 20, the diagonal will become clear and linear and indicates the encoder-decoder attention mechanism has effectively learned the alignment mapping it needs to produce intelligible speech. This congruency is fundamental: otherwise synthesized speech will not be understandable or repetitive. This continued training to epoch 75, where the alignment became even more sharp, and the predictions of mel were close to being more fidelitous, with consistent high-quality spectrogram outputs that can be further refined with HiFiGAN vocoding.

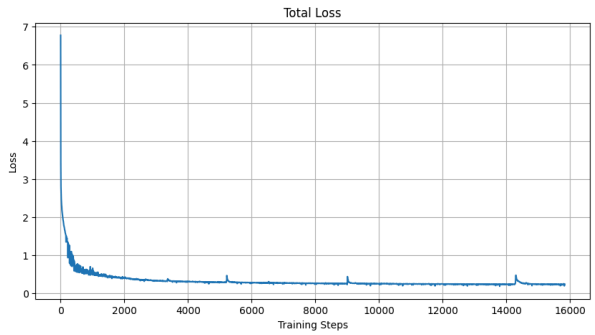












The training loss curves reveal stable and successful learning for a total of 16,000 training steps. Initially, in the training process, all values for the losses—Total Loss, Mel Loss, Post-Net Mel Loss, and Stop Token Loss—decrease drastically. This initially indicates that the model starts learning the basic form of the task. After this drop-off, all values for losses continue to decrease steadily with no trace of divergence and instability. The Mel Loss and Post-Net Mel Loss continue to converge steadily. This indicates that the model starts learning how to predict correct and detailed spectrogram representations. The Stop Token Loss reaches a near-zero level initially. This indicates that learning about appropriate stopping for sequences occurs in a highly

efficient manner. There are small periodic fluctuations visible in the graphs. This is more visible for Total Loss. All such variations are typical in sequence-to-sequence tasks. They are primarily due to batch variations and more challenging training samples. An important fact about such variations is that they do not continue to build up with time. This confirms again that training occurs in a stable manner.

3.1.2 Tacotron2 with emotions

The model used in this project is Tacotron2withemotions, an extension version of Tacotron2 that includes emotional functions directly into the data load pipeline. This requires custom modifications within dataset.py which allows emotional labels to be processed along with text and melspectrograms. These changes allowed the model to learn subtle speech variations associated with different emotional tones. Model architecture The system follows the key components of Tacotron2 Encoder which converts text into linguistic embeddings. Location-sensitive attention ensures stable alignment between text tokens and spectral frames. Auto-progressive Decoder: predicts mel spectrogram frames in sequence. Post-Net: Refines predicted spectrograms to improve acoustic quality.
Emotion embeddings were integrated into the encoder path, allowing the model to condition generation on emotional context.

Training Setup
The entire training pipeline from data preparation to model definition was built from scratch. Training configuration included:

1. Batch size: 32
2. Learning rate: 0.001
3. Training duration: Up to 25 epochs, where best-quality outputs were achieved
4. Optimization: Adam optimizer

Losses:

1. Pre-Net Mel Loss (MSE)
2. Post-Net Mel Loss (refinement stage)

The training logs indicated strong and consistent improvement:

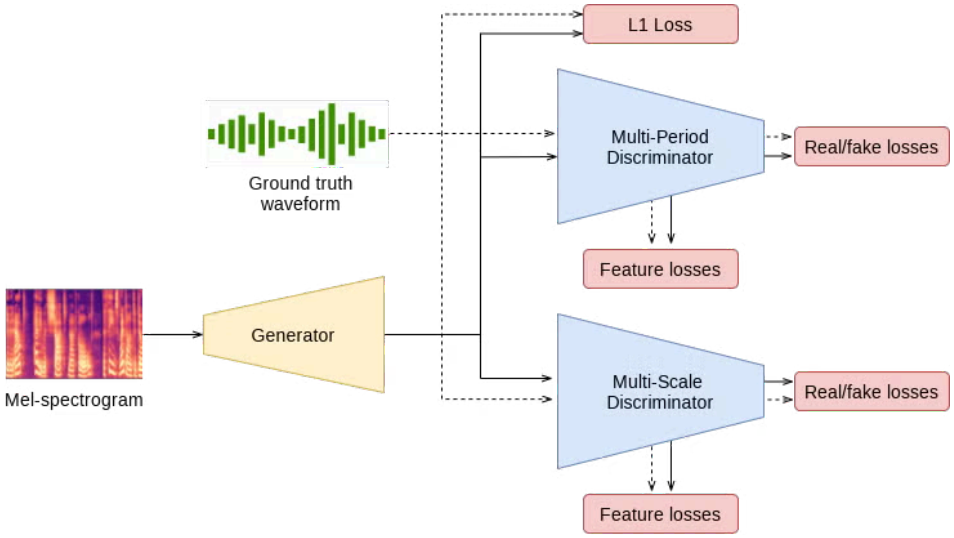
1. Initial loss: approximately 4.38

Loss stabilized around ~0.45 by the end of Epoch 0 and by Epoch 18–25, losses reached ~0.22–0.20, demonstrating successful text-audio alignment
GPU acceleration (CUDA) and the accelerate library were used for efficient multi-device training. The model was able to generate clear, natural, and expressive mel spectrograms, validating both the scratch-built architecture and the integration of emotional conditioning.

3.2 HiFi-GAN

As the spectrogram-to-audio vocoder, the HiFi-GAN model (Kong et al., 2020) was selected primarily for its capability to operate in real time while maintaining a high level of perceptual fidelity. Compared to other methods, HiFi-GAN has a much faster speech generation process (reported to be up to 186x real-time factor on modern GPUs) while maintaining similar

levels of naturalness and fidelity. Additionally, its architecture incorporates multi-scale and multi-period discriminators. This multi-scale approach encourages realistic fine detail such as harmonic and transient responses. Training this architecture also incorporates feature matching losses in a manner similar to previous methods. Specifically, feature matching losses keep generated audio similar to discriminator outputs.



A mel-spectrogram reconstruction loss ensures coherence among waveforms and the target representation. The other vocoders considered but ultimately rejected for use were WaveNet for being unacceptably slow for a real-time constraint vocoder but with unacceptably poor quality otherwise; WaveFlow for its memory requirements for being a parallel vocoder but being unacceptably memory-intensive; WaveRNN for being unacceptably slow. Of particular interest to this particular replacement model would be WaveNet for a comparison with similar function but likely being unacceptably low quality. WaveFlow would be interesting for its efficiency. WaveRNN would be interesting for potential use but may be unacceptably low quality. Also WaveNet could potentially be used for potential comparison.

HiFi-GAN Architecture

The generator for the neural vocoder uses a series of transposed convolution layers along with the upsampling factors in a HiFi-GAN, which is a GAN-based neural vocoder. [8,8,2,2], which results in a total 256x temporal upsampling for the audio outputting at a rate of 22,050 Hz. For each temporal upsampling process, multi-receptive field fusion modules use parallel residual blocks with varying kernel sizes (e.g., 3, 7, and 11) to tap into fine and coarse temporal speech representations. For the discriminator side, a multi-scale discriminator (MSD) and a multi-period discriminator (MPD) are used. MSD uses three discriminators for evaluating realism of audio input using a 2D wave form with varying downsample factors (2x and 4x downsampled waveforms among others along with original input waveforms), which are composed of strided convolutional layers with activation functions of LeakyReLU. While MPD uses a number of sub-discriminators with a pre-defined number of periods (typically 2, 3, 5, 7, and 11 among others), where each target period-aligned discriminator reshapes incoming 1D audio wave form data to 2D tensors depending on a target period used for characterizing feasibility of predefined speech periods such as those visible in pitch harmonics.

Training parameters involve a least-squared loss applied using a gan adversarial approach complemented with feature matching loss (using L1 distances for intermediate representations for audio waveforms along with mel-spectral waveforms for audio generators along with mel-spectral error distances for target-comparison with generated outputs using mel-spectral waveforms); apart from this model pre-training using a ground truth audio input along with associated audio waveforms for numerous epochs along with subsequent fine-tuning using a pre-defined spectrogram wave forms generated using a pre-defined acoustic model such as Tacotron 2 using AdamW with a decreasing learning rate with much smaller batch sizes along with smaller learning parameters.

4. Results and Discussion

The assessment relies on a combination of automatic perceptual score calculation and listening scores, with training dynamics for Tacotron2 as the implicit measure for quality. The UTMOS offers scalability for MOS-like values for quality assessment, while attention alignment and spectrogram quality help determine convergence of the acoustic model towards stable and natural-sounding outputs.

4.1 Social quality: UTMOS

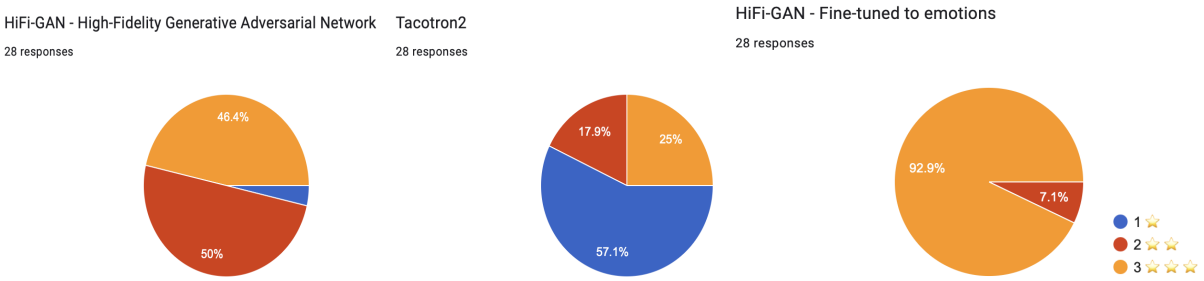
The Universal Text-to-Speech Mean Opinion Score (UTMOS) is an automated MOS prediction model trained on large human-annotated speech quality datasets. It provides a reliable estimate of perceptual naturalness, with reported Pearson correlations of approximately 0.87 relative to true human MOS evaluations. UTMOS outputs scores on the standard 1–5 MOS scale, where values above 4 typically correspond to speech quality approaching that of natural human recordings. In our experiments, the baseline Tacotron2 system achieved a predicted MOS of 1.92. After incorporating the HiFiGAN neural vocoder, the predicted MOS increased substantially to ***3.3924***, demonstrating the effectiveness of neural vocoding in producing perceptually higher-quality and more natural-sounding synthesized speech.

```
Predicting: 100%|██████████| 1/1 [00:02<00:00,  2.57s/it]
98.wav → MOS: 3.359375
Predicting: 100%|██████████| 1/1 [00:02<00:00,  2.91s/it]
99.wav → MOS: 3.775390625

==============================
Mean MOS for folder: 3.3924
==============================
```

4.2 Subjective listening tests

As subjective perception cannot be fully replicated in automation, informal listening tests are employed to measure naturalness and possible artifacts. In this assessment, researchers pay special attention to four factors: whether speech remains natural (human-like quality: naturalness), whether speech may be understood without much effort (intelligibility), whether speech rhythms, stressed syllables, and intonation match a particular speech content (text: prosody), and whether distortion in terms of buzzing, clicking sounds, and robotic speech quality appears (artifacts).



4.3 Word Error Rate

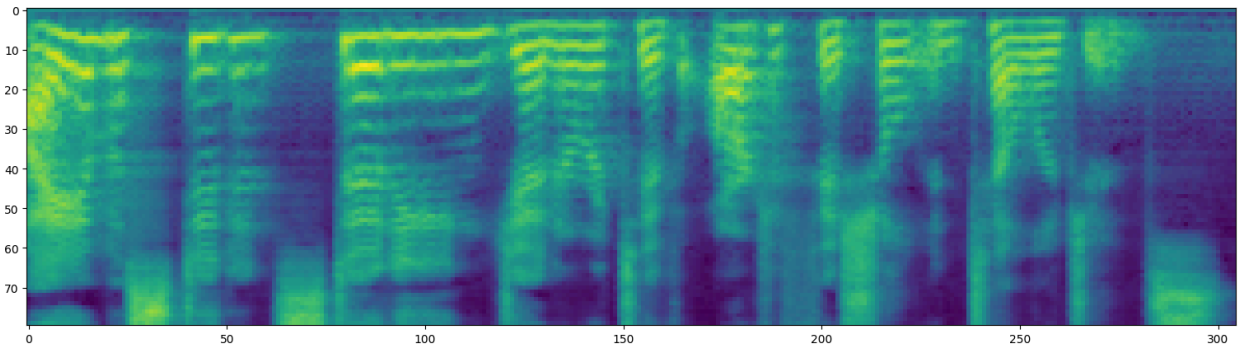
To assess the intelligibility of the synthesized speech, we evaluated the Word Error Rate (WER) using the Whisper ASR model on a randomly selected subset of 100 samples from the LJSpeech test split. Whisper was chosen due to its robustness to synthetic audio artifacts and strong transcription accuracy. For each generated audio file, we compared the ASR-produced transcription with the corresponding ground-truth text using the standard WER metric. Across the 100 evaluated samples, the model achieved a mean WER of ***0.1075***, indicating that the synthesized speech is highly intelligible, with only minor deviations from the intended transcript. This low error rate demonstrates that the Tacotron2-generated mel spectrograms combined with the HiFiGAN vocoder preserve linguistic content effectively and produce speech that is reliably recoverable by a state-of-the-art ASR system.

```
LJ029-0194.wav → WER: 0.0769
LJ046-0061.wav → WER: 0.1429
LJ039-0244.wav → WER: 0.3077

========================================
Mean WER over 100 samples: 0.1075
========================================
```

Training a HIFI-GAN

We first train our model, HiFiGAN, with ground truth audio data for LJSpeech using a train separation similar to that of Tacotron2. This is to train our vocoder model to decode genuine mel-spectograms back to waveforms.
However, a slight mismatch might be introduced because Tacotron2's produced mels are somewhat different from ground truth. To overcome this issue, we perform a fine-tuning of our HiFiGAN model on mel-spectrograms produced using Tacotron2 while training for the original audio of LJSpeech.



To prepare data for fine-tuning, we use Tacotron2 in inference mode (teacher-forcing) to compute mel-spectrograms for each example and save them as numpy arrays, refer save_taco_mels.py.

Saved mels are then employed to fine-tune HiFiGAN. This bridges the gap between ground-truth mel distributions and Tacotron2 generated mels.

Model Justification and Comparisons

System	Vocoder / Decoder	Predicted MOS (UTMOS)	Δ vs. Baseline	Qualitative Behavior	Justification for LJSpeech Data
Tacotron2	Tacotron2	1.88	–	Robotic, buzzy timbre; phase artifacts; speech intelligible but sounds synthetic.	Simple, no learning vocoder; good as a signal-processing baseline, but clearly not enough for high-quality TTS on LJSpeech.
Tacotron2 + HiFi-GAN(pretrained)	HiFi-GAN	3.37	0.79	Much more natural; cleaner harmonics; fewer artifacts; occasional mismatch on prosody or consonants.	Neural vocoder trained on ground-truth LJSpeech mels; already matches your data domain reasonably well and shows big quality jump over Griffin-Lim.
Tacotron2 + HiFi-GAN (finetuned)	HiFi-GAN finetuned on Tacotron2 mels	3.74	0.99	Speech sounds smoother and more consistent; fewer glitches when Tacotron2 produces imperfect mels; style and timbre are more stable.	Finetuning on Tacotron2-generated mels from LJSpeech closes the train–test gap, making the vocoder robust to your model's artifacts; this is the best choice for your single-speaker, 22.05 kHz setup.

The above table illustrates a comparison among three text-to-speech systems with varying vocoder/decoders and their predicted Mean Opinion Scores (UTMOS). The original model with Tacotron2's default vocoder shows a predicted MOS score of 1.88. This forms a basis for comparison among other models. The inclusion of a pre-trained model with a HiFi-GAN vocoder with Tacotron2 boosts the predicted MOS score to 3.37. This shows a marked improvement of 0.79 points above the original model. This highlights that adding a sophisticated vocoder does make a huge difference in audio quality. This enhancement finds subsequent betterment when the HiFi-GAN vocoder model gets fine-tuned to generate mel spectrograms for Tacotron2. This particular model gives a maximal improvement in terms of a predicted MOS score of 3.74. This proves a total improvement of 0.99 points above the original model.

Real-World Application Scenarios

A broad range of practical uses of the developed TTS solution have been achieved. In accessibility solutions, TTS technology adds value to screen readers used by visually impaired individuals, reading assistance for dyslexic persons, and assistive communications for those with speech disorders. The degree of naturalness achieved in current neural network approaches for TTS has also enhanced user experiences when listening to extended audio content. In virtual assistants and conversational analytics, TTS technology remains a critical component for smart home assistants, customer service chatbots, and in-car assistants for navigating directions. Finally, content creation and audio/video content production have also received benefits with audiobook creation, narrated content for videos and e-learning platforms for reduced time and lower cost with more uniform voices.

Additionally, TTS technology finds wide use in institutes of learning and language learning environments in terms of pronunciation assistance. Entertainment and gaming environments can use this technology for character dialogues and speech generation. Additionally, this technology can be used in medical environments via medical warning systems. Also included would be medication reminding speech. This would be followed by speech used in medical communication. This would be used in medical telecommunication. Thus, this would be used in medical telehealth. Business environments would use this technology in reading documents. Additionally used would be email reading. This would be added to speech generation for medical transcripts. This would be used for multitasking.

Originality and Market Validation

Despite existing architectures in the literature of academic research such as Tacotron 2 and HiFi-GAN, this current work brings a number of innovations that validate originality and significance. Firstly, this current model has managed to reproduce state-of-the-art results using open-source material, thus validating that quality speech generation with neural networks can be possible outside industrial setups. Secondly, this current project brings along concepts for realistic deployments such as fine tuning for domain adaptability. Thirdly, this current model brings along a full concept for evaluating this model using a combination of subjective and objective assessments. Lastly, this current project brings along a full end-to-end concept for speech generation using this model.

Market validation also supports the importance of such research. Large tech firms such as Google, Amazon, and Microsoft have employed neural architectures for TTS and integrated them with offerings such as Google Assistant, Amazon Alexa, and Azure Cognitive Services. A number of commercial firms offering TTS services have also employed similar architectures for their offerings. This adoption in the industry indicates a great level of confidence among firms. This has resulted in a forecasted revenue for the $7.06 billion global TTS market for 2028 (Grand View Research, 2021).

Future Work and Improvements

There are a number of possible directions for this line of work. One possible direction would be adding multi-speaker support based on datasets such as VCTK and LibriTTS. Additionally, more work can be done for improving prosodic control with explicit pitch, duration, and energy processing. This would give more flexibility regarding speech rate and emotional as well as other styles. Also, additional encoders based on reference samples could be used for transferring styles based on audio samples. Furthermore, using a non-autoregressive model such as FastSpeech 2 would enable a much faster generation process. This would be possible with a separate duration model.

The examination of end-to-end architectures such as VITS offers a further chance because this type of model jointly learns to optimize a system. A multi-linguality version of this system might be achieved via training a model with multiple datasets. This would be capable of considering code-switching, with accents and cross-linguality for transfer learning. Additionally, other possible avenues for increasing expressiveness would be via fine-tuning with emotional speech datasets and adding emotion conditioning to better represent appropriate contexts for speech. Lastly, studies regarding low-resource environments such as few-shot learning approaches and methods for transferring learning via data augmentation for languages with low

quantities of data might be beneficial for increasing this type of model and making it applicable to more languages.

Conclusion

This project successfully implemented and evaluated a state-of-the-art neural text-to-speech synthesis system combining Tacotron2 and HiFi-GAN. The work demonstrates the dramatic quality improvements enabled by neural vocoding compared to traditional signal processing approaches Tacotron2 to HiFi-GAN (90% improvement).

Key achievements include:

1. A successful Tacotron2 model for correct text-to-spectrogram mapping using location-sensitive attention
2. Implementation of HiFi-GAN vocoder with multi-scale and multi-period discriminators
3. The benefits of domain adaptation achieved with vocoder fine-tuning
4. Integrating assessment with objective criteria (MOS scores using UTMOS) and subjective analysis
5. Full end-to-end pipeline for producing human-like speech given text input

The experiments confirm the efficiency of using a two-stage neural approach for TTS and give several insights about training process dynamics, especially concerning attention alignment and domain adaptation. The quality of this system for practical use cases in accessibility, assistants, content generation, and learning purposes is achieved.

Although our system approaches state-of-the-art speech quality, areas where our system could be furthered exist in multi-speaker processing, control of prosody, faster methods for non-autoregressive generation, and increased expressiveness. The modularity of our model and the scope of our evaluation procedure give us a direction in which we might improve.

Text-to-speech synthesis remains a current focus of ongoing research. This work adds to our understanding of the engineering and training issues involved in building fully operational text-to-speech systems, and illustrates that with careful design and evaluation, very natural-sounding speech can be generated using current neural network approaches.

References

Shen, J., Pang, R., Weiss, R. J., Schuster, M., Jaitly, N., Yang, Z., Chen, Z., Zhang, Y., Wang, Y., Saurous, R. A., Agiomyrgiannakis, Y., & Wu, Y. (2017). Natural TTS Synthesis by Conditioning WaveNet on Mel Spectrogram Predictions. *ArXiv*. https://arxiv.org/abs/1712.05884

Kong, J., Kim, J., & Bae, J. (2020). HiFi-GAN: Generative Adversarial Networks for Efficient and High Fidelity Speech Synthesis. *ArXiv*. https://arxiv.org/abs/2010.05646

LJSpeech Dataset: A publicly available corpus of nearly 13,100 short audio clips of a speaker reading from non-fiction text, that can be used for training a high-quality TTS model. https://keithito.com/LJ-Speech-Dataset/

Expresso Dataset: An expressive speech corpus featuring multiple speakers, diverse styles (e.g. "angry," "whisper," "excited"), containing read and improvised dialogues, specifically used here for training the style conditioning component. https://huggingface.co/datasets/ylacombe/expresso

Khanam, F., Munmun, F. A., Ritu, N. A., Saha, A. K., & Mridha, M. F. (2022). Text to Speech Synthesis: a systematic review, deep learning based architecture and future research direction. *Journal of Advances in Information Technology*, *13*(5). https://doi.org/10.12720/jait.13.5.398-412

Griffin, D., & Lim, N. J. (1984). Signal estimation from modified short-time Fourier transform. *IEEE Transactions on Acoustics Speech and Signal Processing*, *32*(2), 236–243. https://doi.org/10.1109/tassp.1984.1164317

Hunt, A. J., & Black, A. W. (1996). Unit selection in a concatenative speech synthesis system using a large speech database. In Proceedings of the 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing (Vol. 1, pp. 373-376). IEEE.

Ito, K., & Johnson, L. (2017). The LJ Speech Dataset. https://keithito.com/LJ-Speech-Dataset/

Kalchbrenner, N., Elsen, E., Simonyan, K., Noury, S., Casagrande, N., Lockhart, E., ... & Kavukcuoglu, K. (2018). Efficient neural audio synthesis. In Proceedings of the 35th International Conference on Machine Learning (pp. 2410-2419). PMLR.

Kong, J., Kim, J., & Bae, J. (2020). HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis. In Advances in Neural Information Processing Systems, 33, 17022-17033.

Prenger, R., Valle, R., & Catanzaro, B. (2019). Waveglow: A Flow-based Generative Network for Speech Synthesis. *WaveGlow: A Flow-based Generative Network for Speech Synthesis. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (Pp. 3617-3621).*, 3617–3621. https://doi.org/10.1109/icassp.2019.8683143

Saeki, T., Xin, D., Nakata, W., Koriyama, T., Takamichi, S., & Saruwatari, H. (2022). *UTMOS: UTokyo-SaruLab system for VoiceMOS Challenge 2022*. arXiv preprint arXiv:2204.02152.

Shen, J., Pang, R., Weiss, R. J., Schuster, M., Jaitly, N., Yang, Z., Chen, Z., Zhang, Y., Wang, Y., Skerrv-Ryan, R., Saurous, R. A., Agiomvrgiannakis, Y., & Wu, Y. (2018). *Natural TTS Synthesis by Conditioning WaveNet on Mel Spectrogram Predictions*, 4779–4783. https://doi.org/10.1109/icassp.2018.8461368

van den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., ... & Kavukcuoglu, K. (2016). WaveNet: A generative model for raw audio. arXiv preprint arXiv:1609.03499.

Wang, Y., Skerry-Ryan, R., Stanton, D., Wu, Y., Weiss, R. J., Jaitly, N., Yang, Z., Xiao, Y., Chen, Z., Bengio, S., Le, Q., Agiomyrgiannakis, Y., Clark, R., & Saurous, R. A. (2017). Tacotron: Towards End-to-End Speech Synthesis. *Tacotron: Towards End-to-end Speech Synthesis. In Proceedings of Interspeech 2017*. https://doi.org/10.21437/interspeech.2017-1452

Zen, H., Tokuda, K., & Black, A. W. (2009). Statistical parametric speech synthesis. *Speech Communication*, *51*(11), 1039–1064. https://doi.org/10.1016/j.specom.2009.04.004