Conditional Text-To-Speech

Shreya, Saha
Department of Computer Science
UC San Diego La Jolla, CA 92093
ssaha@ucsd.edu

Ye Jin, Jeon
Department of Computer Science
UC San Diego La Jolla, CA 92093
yejeon@ucsd.edu

Parth, Doshi
Department of Computer Science
UC San Diego La Jolla, CA 92093
pdoshi@ucsd.edu

Venkat, Krishnamohan
Department of Computer Science
UC San Diego La Jolla, CA 92093
vkrishnamohan@ucsd.edu

Abstract

We aim to generate text-to-speech (TTS) audio samples which are conditioned on multiple parameters like emotions, gender, age, voice and accent. For the scope of this project, we first focus on emotion conditioning, expanding to other parameters depending on time constraints. This will allow users to specify what kind of emotion they wish to express in different parts of the transcript, leading to more expressive voices in applications like audio-books and personalized voice assistants. Most research in this area focuses on zero-shot adaptation - a reference voice sample is provided to ensure that the TTS model generates voices similar to the reference voice sample. We are solving a more general problem, wherein a voice can be conditioned on the emotions (and other parameters like gender), rather than just a specific voice sample. This would allow users to express the same sentence in multiple emotions. While some open-source projects do exist which work on this problem, they use older TTS models and the results are not available. We aim to create a model wherein the user supplies a sentence, a reference voice sample and an emotion label, and the output is the audio sample of the sentence, which would match the reference voice and the chosen emotion.

1 Introduction

Problem Definition. We want to generate audio samples given a transcript and a specific emotion. Most existing text-to-speech models such as FastSpeech2 [13] do not allow the synthesized audio to be conditioned on specific parameters like emotions, gender or a specific custom voice. Through this project, we want to introduce emotions as a conditional parameter, to synthesize audio that matches the transcript and specified emotion. Another conditioning factor is a specific voice sample, as used in the zero-shot adaptation method proposed in AdaSpeech 4 [18], which would facilitate custom-voice speech synthesis.

Problem Significance. Text-to-speech models fall under the category of generative modeling, and often involve architectures such as auto-regressive models and GANs. Since we can use these models to generate new speech samples based on the training data, it falls under the category of Deep Generative Models.

Current TTS models only extract phonemes and variances from a given transcript and generate a random voice based on extracted features. Our work aims to supplement the transcript with a reference voice to improve the variability and generality of TTS models. This model would be useful

Preprint. Under review.

for audio-books where characters express dynamic emotions. Furthermore, this would help people with speech disabilities express their thoughts and emotions via text. It should also be of help to autistic children to understand emotions in spoken statements. It should prove to be a great tool for people with learning disabilities.

Technical Challenge. For our specific use-case, it is difficult to find a high quality TTS dataset that has variation in speech attributes like emotions, accent, speaker gender and age, primarily for privacy concerns. Datasets also face the issue of having poor recordings, which adds noise to the model. We plan to focus on the use of emotions and a reference voice, since we have access to some datasets which provide this metadata. Also, as with all other TTS problems, there is no quantitative method to calculate accuracy of our model, and we plan on using variations of MOS scores. Another challenge in the Text-To-Speech domain is the difficulty of extracting features such as prosody and style from audio samples. It is also difficult to do explicit feature engineering for the emotion embedding. There are various ways to generate labels for emotion or extract emotional features in speech. Embedding emotions into the generated voice is another challenge.

State-of-the-Art. FastSpeech2 [13] is the current state-of-the-art non-autoregressive model for Text-to-Speech Synthesis. It can be trained with data collected from multiple speakers, and can synthesize examples for a specific speaker from the training corpus. However, it has no way of inferring the correct emotion to be expressed, and the user does not have the option of conditioning the audio with a specific emotion, which reduces the expressiveness of the synthesized audio sample.

This open-source repository [8] works on extending FastSpeech2 to include emotions by adding a simple emotion embedding to the decoder layers. However, we have not been able to analyse the results, since no audio samples are provided. Additionally, the emotion embedding is only added to the decoder, and ignores the relation between the encoding of the phonemes and how it is dependent on the emotion to be expressed.

GST [16] creates a store of style tokens during training on the TTS task. These style tokens (or embeddings) can be used during inference to generate audio samples with different characteristics. However, the style tokens are not explicitly guaranteed to model emotions, or other specific characteristics such as age, gender or accents.

Contributions.

- Contribution 1: Improve generation of speech conditioned on emotions.
- Contribution 2: Highlight the gaps in the state-of-the-art TTS models with regards to the variability and expressiveness of the synthesized voice.

2 Related Work

Traditional Speech Synthesis Converting text to natural speech has always been an area of interest in the domain of machine learning because of its wide range of applications, especially in helping people with learning and speech disabilities. Traditional text-to-speech methods [15] include Articulatory Synthesis (human articulator's behavior is simulated, e.g. - simulating the movement of the vocal tract), Formant Synthesis (speech is synthesized using an acoustic model based on features like fundamental frequency, voicing and noise levels, handpicked by linguists to mimic natural human speech), Concatenative Synthesis (database consists of speech units ranging from whole sentences to a few syllables, and during inference time speech units are mapped to input texts and are concatenated together) and Statistical Parametric Synthesis (instead of directly concatenating speech waveforms, acoustic parameters are generated from the input texts, and are then converted into speech). The main drawbacks of these methods were that they required large training sets, depended on domain-specific rules, and generated speech samples that were not smooth, difficult to understand, and did not sound like natural human speech.

Deep Learning-based TTS Many initial TTS [1, 17] models were seq-to-seq models. As they predict frames sequentially, they suffer from slow inference and lack of controllability. Further, these models only learn an averaged prosodic distribution over their input data, thereby generating less expressive speech. Global Style Tokens [16] provide TTS models the capability to choose a speaking style appropriate to the given context. FastSpeech [14] uses a feed-forward transformer network to speed up mel-spectrogram generation. FastSpeech2 [13] achieves speed-up over FastSpeech and introduces some variance in the synthesized speech, including pitch, energy, and more accurate duration.

Adaptive TTS Adaptive TTS can synthesize unseen voices. The main challenge of Adaptive TTS is insufficient data for target voices. To tackle this problem, few-shot adaptive TTS [4, 18, 12] is usually adapted by training a source TTS model on a large multi-speaker dataset and fine-tuning it on a few speech samples from target speakers. Zero-shot adaptive TTS [18, 3] can generate a new voice by only modeling the speaker characteristics from a reference speech, without adapting the source TTS model on the speech data of new speakers. These models extract, and condition on, speaker representation to synthesize speech.

Emotional Speech Synthesis [10] is a pioneering method for emotional TTS. It proposes a LSTM-based acoustic model, where several kinds of emotional category labels, such as a one-hot or perception vector, are injected into the decoder. [6] synthesizes expressive emotional speech by conditioning phoneme-wise emotion information based on fine-grained emotion intensity.

3 Methodology

Problem Setting. Given -

1. x = input transcript
2. $S(x)$ = Speech associated with transcript, in the mel-spectogram representation x
3. $P(x)$ = Phoneme Embedding of transcript x
4. $E(x)$ = Emotion Embedding of transcript x
5. Enc = Encoder model - returns a combined representation of the phonemes and the specified emotion
6. Dec = Decoder model - uses the combined embeddings to generate an appropriate mel-spectogram representation of speech
7. L_2 = Mean Squared Error
8. Goal :: $Dec(Enc(P(x) + E(x)))$
9. To minimize Loss

$$L = L_2(Dec(Enc(P(x) + E(x))),\ S(x))$$

where the addition symbol represents how the emotion embedding of 256-d is directly added to each of the phoneme embeddings in the sentence, which are also of 256-d.

Using a pre-trained FastSpeech2 model, we add or modify 3 components:

- Emotion encoder: It takes as an input a one-hot encoding of the emotion class. It generates an embedding of a fixed dimension D. The embedded emotions are passed to the encoder in our model, as opposed to the decoder in the Expressive-FastSpeech2 model.
- Encoder: This component already exists as part of the FastSpeech2 model. We will modify it to receive the emotion embedding and the reference audio as input along with the transcript. This would allow the model to focus on phonemes important for the selected emotion and voice.
- Mel decoder: This component exists as part of the FastSpeech2 model, but we will modify it to use the emotion embedding to generate better mel-spectograms.

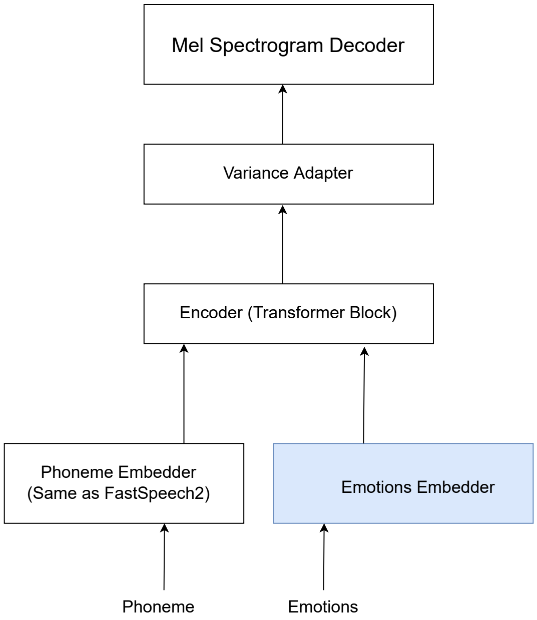

Figure 1: The ground truth speech will be converted to mel-spectograms and compared with the output of the mel-spectogram decoder, thus forming our objective function. The Emotions Embedder is highlighted.

Description. Given a text transcript (x) and an emotion label (E_l), our goal is to convert the text x to speech in the desired emotional tone. One of the stretch goals of our project is to output the speech in a given custom voice S(x). Our model structure is very similar to the FastSpeech2 architecture which uses an encoder built on top of a transformer model and a decoder (as described in Idea Summary). In the loss calculation, we try to minimize the mean squared error between the ground truth speech and the speech produced by our model (L2).

Implementation. For setting up the baselines, we have used the PyTorch framework. We implemented the FastSpeech2 model based on this open-source implementation [5]. As the original implementation was for the LJSpeech dataset, we changed the setup for our ESD baseline. We make the following changes:

- Aligning the ESD dataset samples: This involves using the Montreal Forced Aligner [11] to align the phonemes and audio sounds.
- Preprocessing: We preprocess the aligned samples by converting the audio samples to the correct sampling rate, and extracting features, like pitch, frequency, and mel-spectograms, which are utilized by the FastSpeech2 model.
- Training: We wrote new training configs for the ESD dataset, which differ from the LJSpeech dataset due to the presence of multiple speakers.

4 Experiments

Datasets and Tools. These are the three datasets that we have heavily used in our experiments -

- Interactive Emotional Dyadic Motion Capture IEMOCAP [2] - 12 hours of audio visual data, where the speech is categorized into emotion labels by annotators such as anger, happiness, sadness and neutrality.
- Emotional Speech Dataset [19] - 10 English speakers, 10 Chinese speakers with 350 utterances for each speaker covering the emotions - neutral, happy, angry, sad, and surprise. We have only used the English speech samples for this project.

Split	Happy	Sad	Angry	Neutral	Surprise
Eval Set	82.62	72.79	80.39	81.70	79.66
Test Set	79.10	62.12	79.03	74.44	83.22

Table 1: Validation and Test Accuracy for different emotions after training the SER model for 340 epochs

- LJSpeech Dataset - 13,100 short audio clips of a single speaker reading passages from 7 non-fiction books. No emotion labels are included in this dataset.

Baselines. We used the FastSpeech2 model as our first baseline. We have two versions of the baseline - the original FastSpeech2 model trained on the LJSpeech dataset (FS2-LJS), and the FastSpeech2 model trained on the ESD dataset (FS2-ESD). The emotion detection results on samples from both baselines are given in Table 1. The FastSpeech2 model generates mel-spectograms, which are passed through a pretrained vocoder. We used the HiFi-GAN architecture [7] for the vocoder.

Our second baseline implements the work of [9] over the base FastSpeech2 TTS model using an utterance-level emotion embedding. We use the Expressive-FastSpeech2 model (EFS2) as the baseline implementation (`https://github.com/keonlee9420/Expressive-FastSpeech2`). The emotion embeddings are first transformed, concatenated channel-wise, and then linearly projected with ReLU activation before fusing with the phoneme embeddings. The decoder is then conditioned on the resultant embedding for speech synthesis.

Experimental Model. Our experimental model is based on the EFS2 baseline, but we incorporate the emotion embedding into the encoder rather than the decoder so that the model can choose which phonemes to focus on given the specified emotion.

Evaluation Metrics. We trained a Speech Emotion Recognition (SER) model from scratch (reference code - `https://github.com/MeidanGR/SpeechEmotionRecognition_Realtime`) using the ESD samples. The SER model consists of two hidden LSTM layers with 64 nodes each, and an output layer with 5 nodes (one for each emotion in the ESD dataset). We use this SER model to quantify the emotions in the synthesized samples. The model was trained for a total of 340 epochs. During training, the entire dataset was split into train:valid:test sets with ratios of 87:9:4. As seen in the table below, the trained model does not perform well when it comes to recognising "Sad" data samples.

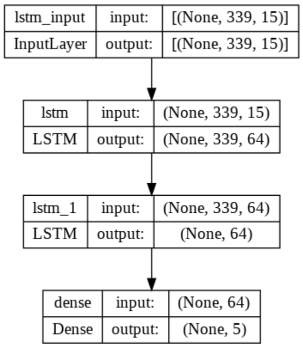
lstm_input | input: | [(None, 339, 15)]
InputLayer | output: | [(None, 339, 15)]
lstm | input: | (None, 339, 15)
LSTM | output: | (None, 339, 64)
lstm_1 | input: | (None, 339, 64)
LSTM | output: | (None, 64)
dense | input: | (None, 64)
Dense | output: | (None, 5)

Figure 2: SER Model Architecture

Quantitative Results. We note that the ESD model was trained to recognise 5 emotions - Happy, Sad, Angry, Neutral and Surprise. However, our evaluation examples do not include "Surprise". Due to time crunch, we found it easier to train our model with four of the most commonly used emotions. Also, when we analysed the ESD Dataset samples, we found the "Surprise" samples to be almost indistinguishable from the "Happy" samples. We chose 25 sentences in each category that we classified with a high confidence, i.e clearly belonging to a particular class. For example, 'I am going to ask her to marry me.' is an obvious happy sentence. We synthesized 4 speech samples, one

Model	Happy	Sad	Angry	Neutral	Total
FS2-LJS	0.0	0.04	0.68	0.32	0.26
FS2-ESD	0.2	0.04	0.76	0.0	0.25
EFS2-ESD	0.61	0.46	0.87	0.04	49.5
Mod-EFS2-ESD	0.63	0.45	0.88	0.07	50.75

Table 2: Inference Accuracy for various emotions of each baseline model and our proposed model (Mod-EFS2-ESD)

per emotion, for each sentence, and quantitatively assessed their quality with the SER's recognition accuracy.

As can be seen from Table 2, the SER model gave an average accuracy of 25.5% for the vanilla FastSpeech2 models, however the accuracy increased to 49% on the EFS2-synthesized samples. Samples from our model (Mod-EFS2-ESD) yields an accuracy of 50.75%, and performs better than all the baselines.

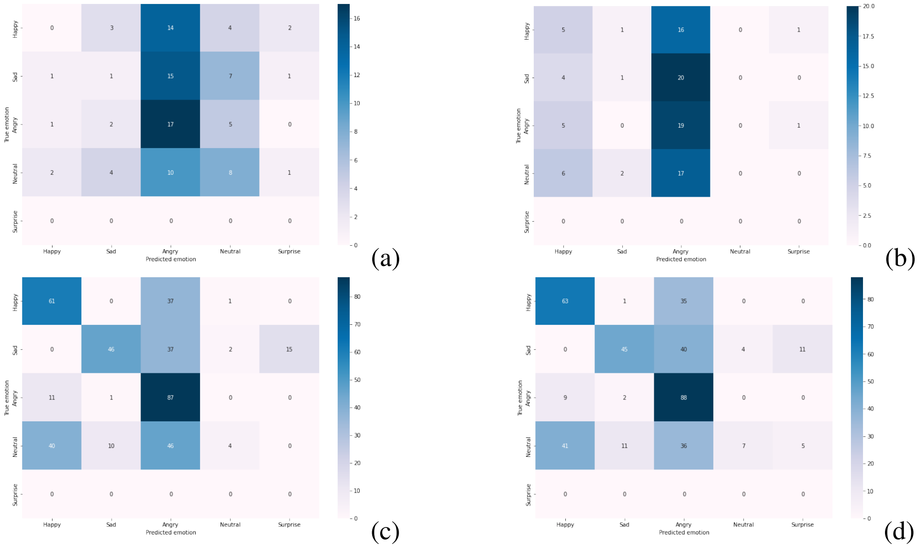

Figure 3: SER results of (a) FS2-LJSpeech, (b) FS2-ESD, (c) EFS2-ESD, (d) Mod-EFS2-ESD

Prior research[8] shows that pitch, speaking rate, and energy are important features for speech generation and emotion classification. We analysed the emotion-specific distribution of each feature for the synthesized results of each experiment and compare them with the feature distributions of training samples from the ESD dataset. As shown in Figure 4, features of samples generated by the FastSpeech2 baseline are consistent regardless of emotion type. However, the features of results synthesized based on Expressive-FastSpeech2 model vary according to the emotions and their distributions correlate to the training data distributions.

4.1 Qualitative Results.

We analyse the audio samples synthesized using various baseline models and our experimental model. For baselines 1 and 2, we utilize the FastSpeech2 model trained on LJSpeech and ESD datasets without any emotion embeddings. For baseline 3, we utilize the Expressive-FastSpeech2 model [8], which inserts an emotion embedding into the decoder. This model is trained on the ESD dataset.

For our experimental model, we modified the Expressive-FastSpeech2 model to pass the emotion embedding into the encoder (mod-EFS2), rather than the decoder. This allows the encoder to focus on which phonemes to emphasize based on the emotion. This model is trained on the ESD dataset.

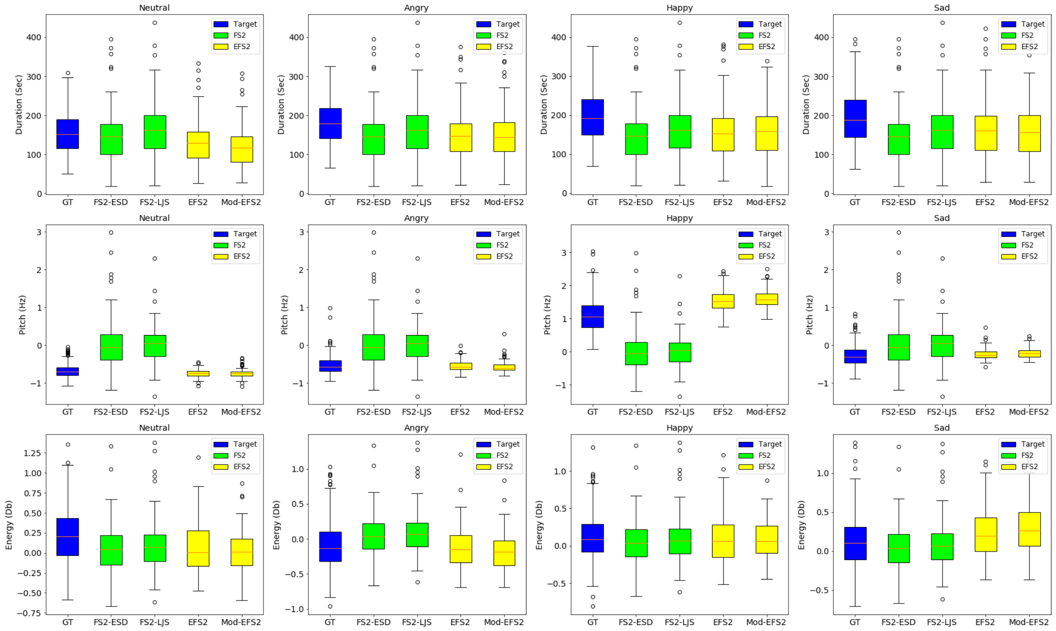

Figure 4: Duration, pitch, and energy distributions of ESD and synthesized samples of each model

4.1.1 Baseline 1 - FastSpeech2 trained on LJSpeech (FS2-LJSpeech)

We use a pre-trained FastSpeech2 model, as explained in the experiments section above. From the results, it can be observed that the model cannot correctly predict what emotion should be expressed solely based on the transcript. For example, the synthesized audio for the texts "I wish I'd never met you!" and "The professor will discuss a new topic today." are extremely similar in intonation, even though the first expresses anger and the second expresses a neutral sentiment. We can also observe this in the mel-spectograms for these samples, as shown in Figure 5.

However, for sentences like "I hate you and everything you stand for!", which contains words directly related to the emotion of anger, the synthesized audio reflects the emotion by raising the volume of the sound generated while synthesizing those specific words.

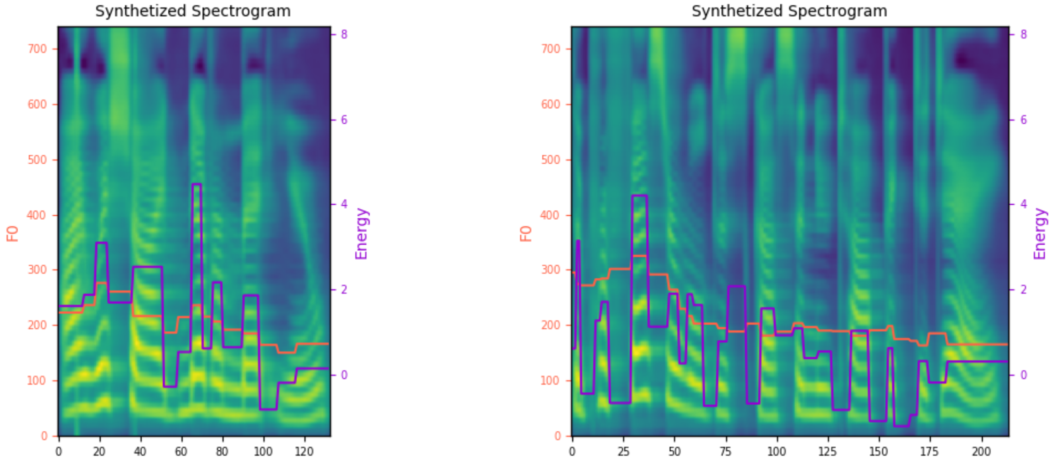

Figure 5: Mel-spectograms for the sentences "I wish I'd never met you!" and "The professor will discuss a new topic today."

4.1.2 Baseline 2 - FastSpeech2 trained on ESD (FS2-ESD)

We train a FastSpeech2 model on the ESD dataset. When comparing the results of the model trained on the ESD dataset with the model trained on the LJSpeech dataset, we can observe that the quality of the voice samples is more synthetic for the ESD baseline as compared to the more natural-sounding audio samples generated using the LJSpeech baseline model. There are three specific reasons for this.

- The LJSpeech dataset only has a single speaker, while we use all 10 English speakers from the ESD dataset.
- The LJSpeech dataset has considerably more audio samples than the ESD dataset.
- The LJSpeech pre-trained checkpoint was trained for much longer than the ESD checkpoint.

However, when focusing on the emotional aspect of the speech, the samples generated from the ESD baseline exhibit the same flaws as shown in the LJSpeech baseline. The model is unable to infer the appropriate emotions from the text alone, which highlights the need for an explicit conditioning factor as proposed in our solutions.

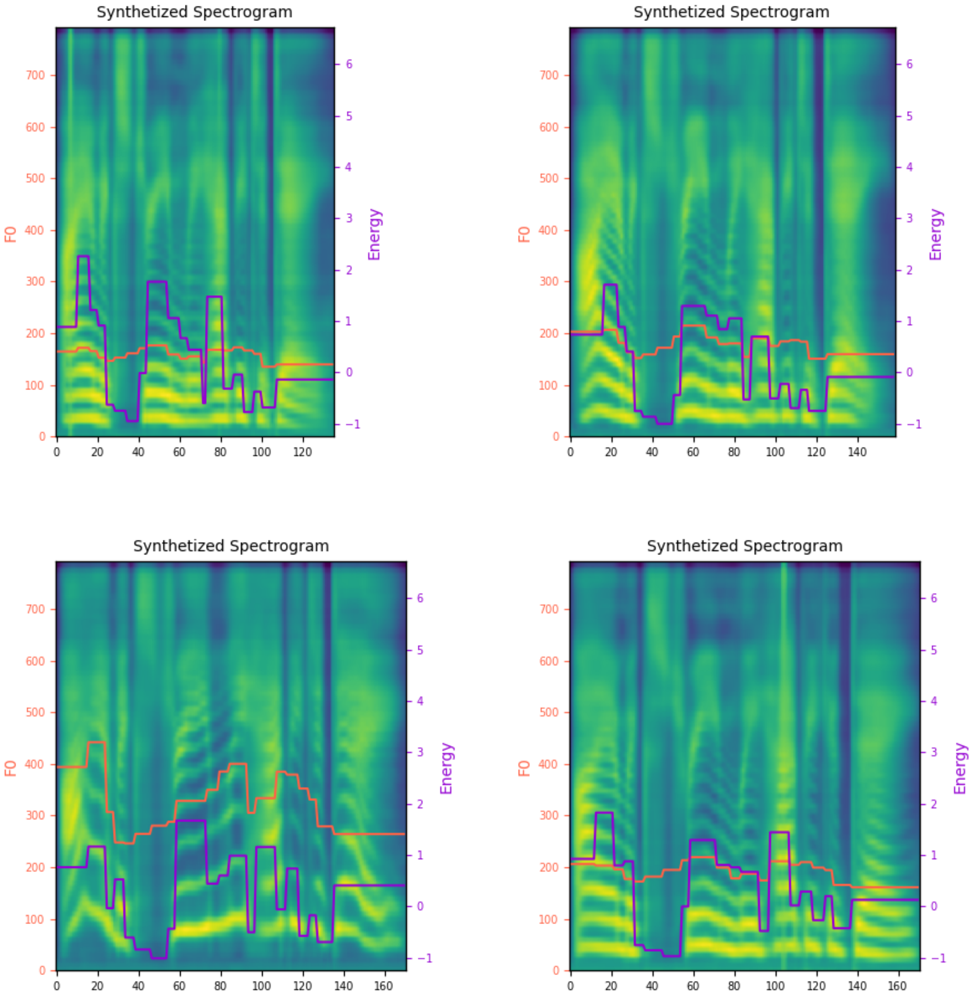

Figure 6: Expressive-FastSpeech2 generated mel-spectograms for the sentence "I am extremely happy today!" with the emotions neutral, angry, happy and sad.

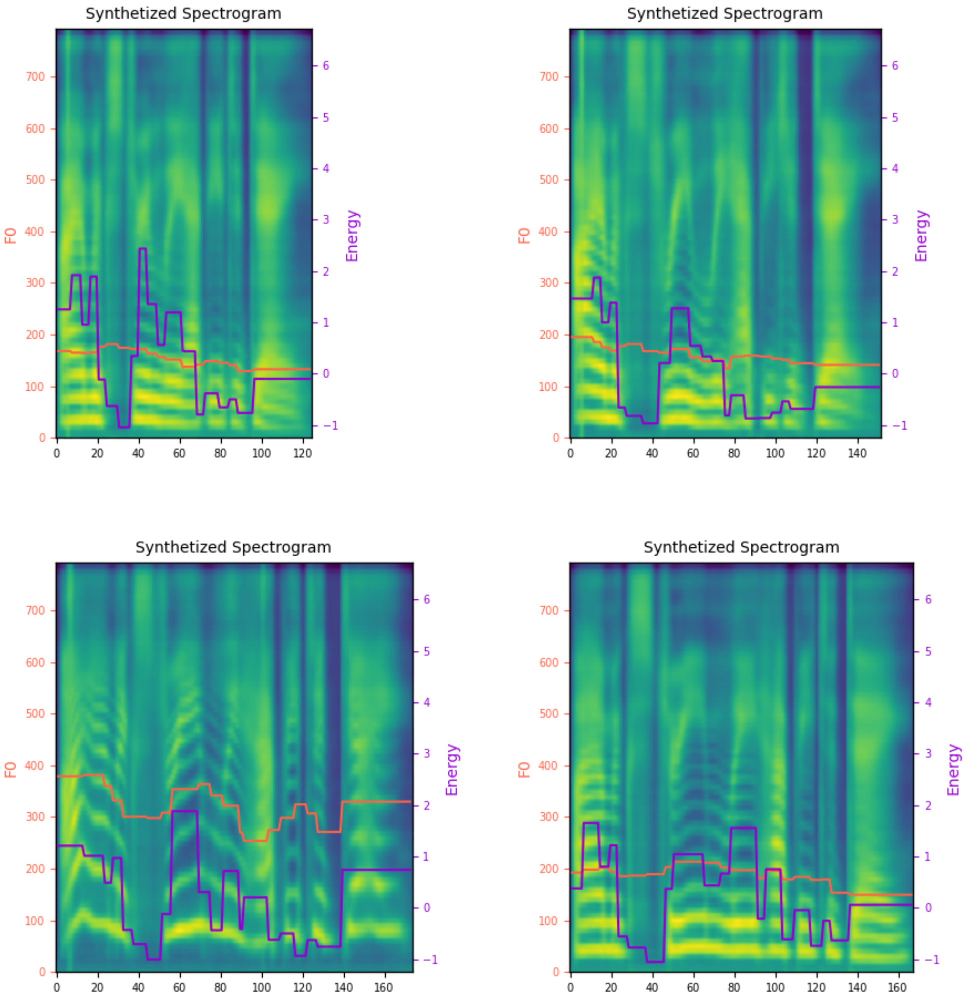

Figure 7: Modified Expressive-FastSpeech2 generated mel-spectograms for the sentence "I am extremely happy today!" with the emotions neutral, angry, happy and sad.

4.1.3 Baseline 3 - Expressive-FastSpeech2 trained on ESD (EFS2-ESD)

We observe a noticeable difference in audio quality, in terms of emotions, from the samples generated in Baselines 1 and 2. The variance adaptor learns the emotion specific distribution of audio features like duration and pitch, as can be inferred from the natural variation of these factors in the output audio. For example, "Happy" synthesized samples have the highest pitch, which correlates naturally with excitement. Similarly, "Sad" synthesized samples are very similar to "Neutral" samples, except are longer duration, resulting in slow, drawn-out speech that resembles sadness. A comparison of the mel-spectograms for the sentence "I am extremely happy today!" synthesized with the same speaker, but with the emotions Neutral, Angry, Happy and Sad, is shown in Figure 6.

4.1.4 Experimental - Modified Expressive-FastSpeech2 trained on ESD (Mod-EFS2-ESD)

We modified the Expressive-FastSpeech2 model to include the embedding as part of the encoder input instead of the decoder input to allow more interaction between the emotion embedding and the phonemes. This model was trained on the ESD dataset. Some of the samples can be found here [1]

[1] Modified EFS 2 Qualitative Examples

The results show significant differences in the audio samples when synthesizing the same speaker and transcript, but with a different emotion. For instance, we can compare the mel-spectograms for the sentence "I am extremely happy today!" when synthesized with the same speaker, but with the emotions Neutral, Angry, Happy and Sad, as shown in Figure 7. The mel-spectograms show varying duration and energy for different emotions, and different words are emphasized. The "sad" voice becomes very soft, since that is how the chosen speaker expresses sadness in the ground truth samples of the ESD dataset. Her "happy" voice is a high-pitched voice, while she speaks more haltingly when angry, as observed in the mel-spectograms. These phenomena match up with how speaker 0019, our chosen voice, speaks in the ground truth examples of the ESD dataset.

However, our model is not perfect. There are two main failure modes.

Rare words Since the model is trained on a relatively small dataset such as ESD which has limited variations in text, it is not able to generate perfect audio samples for rare words or phonemes which it may not have seen in training. The pronunciation of such words is flawed. One possible solution for this would be to pre-train a FastSpeech2 model on the massive LJSpeech dataset, before introducing emotion embeddings and fine-tuning on the ESD dataset.

Controllability of emotions Since we use a categorical label for each sentence, we cannot control the level of emotions or mix two emotions in the same audio sample. This could be resolved by using continuous labels, rather than discrete categories.

4.2 Bottlenecks.

Due to insufficient time, we dropped the idea of using a reference audio sample and focused solely on the emotion conditioning part of our proposal.

Acknowledgements (Portions of) the research in this paper used the ESD Database made available by the HLT lab, National University of Singapore, Singapore.

References

[1] Sercan Ö Arık et al. “Deep voice: Real-time neural text-to-speech”. In: *International Conference on Machine Learning*. PMLR. 2017, pp. 195–204.

[2] Carlos Busso et al. “IEMOCAP: Interactive emotional dyadic motion capture database”. In: *Language resources and evaluation* 42.4 (2008), pp. 335–359.

[3] Edresson Casanova et al. “Yourtts: Towards zero-shot multi-speaker tts and zero-shot voice conversion for everyone”. In: *International Conference on Machine Learning*. PMLR. 2022, pp. 2709–2720.

[4] Mingjian Chen et al. “Adaspeech: Adaptive text to speech for custom voice”. In: *arXiv preprint arXiv:2103.00993* (2021).

[5] Chung-Ming Chien et al. “Investigating on Incorporating Pretrained and Learnable Speaker Representations for Multi-Speaker Multi-Style Text-to-Speech”. In: *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2021, pp. 8588–8592. DOI: `10.1109/ICASSP39728.2021.9413880`.

[6] Chae-Bin Im et al. “EMOQ-TTS: Emotion Intensity Quantization for Fine-Grained Controllable Emotional Text-to-Speech”. In: *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2022, pp. 6317–6321.

[7] Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. “Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis”. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 17022–17033.

[8] Keon Lee. *Expressive-FastSpeech2*. `https://github.com/keonlee9420/Expressive-FastSpeech2`. 2021.

[9] Younggun Lee, Azam Rabiee, and Soo-Young Lee. *Emotional End-to-End Neural Speech Synthesizer*. 2017. DOI: `10.48550/ARXIV.1711.05447`. URL: `https://arxiv.org/abs/1711.05447`.

[10] Younggun Lee, Azam Rabiee, and Soo-Young Lee. “Emotional end-to-end neural speech synthesizer”. In: *arXiv preprint arXiv:1711.05447* (2017).

[11] Michael McAuliffe et al. “Montreal Forced Aligner: Trainable Text-Speech Alignment Using Kaldi”. In: *Proc. Interspeech 2017*. 2017, pp. 498–502. DOI: `10.21437/Interspeech.2017-1386`.

[12] Dongchan Min et al. “Meta-stylespeech: Multi-speaker adaptive text-to-speech generation”. In: *International Conference on Machine Learning*. PMLR. 2021, pp. 7748–7759.

[13] Yi Ren et al. “Fastspeech 2: Fast and high-quality end-to-end text to speech”. In: *arXiv preprint arXiv:2006.04558* (2020).

[14] Yi Ren et al. “Fastspeech: Fast, robust and controllable text to speech”. In: *Advances in Neural Information Processing Systems* 32 (2019).

[15] Xu Tan et al. “A survey on neural speech synthesis”. In: *arXiv preprint arXiv:2106.15561* (2021).

[16] Yuxuan Wang et al. “Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis”. In: *International Conference on Machine Learning*. PMLR. 2018, pp. 5180–5189.

[17] Yuxuan Wang et al. “Tacotron: Towards end-to-end speech synthesis”. In: *arXiv preprint arXiv:1703.10135* (2017).

[18] Yihan Wu et al. “Adaspeech 4: Adaptive text to speech in zero-shot scenarios”. In: *arXiv preprint arXiv:2204.00436* (2022).

[19] Kun Zhou et al. “Seen and unseen emotional style transfer for voice conversion with a new emotional speech dataset”. In: *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2021, pp. 920–924.

A Speech Emotion Recognition Training Details

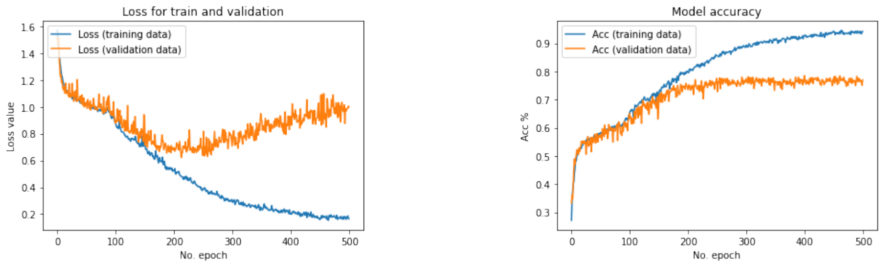

Figure 8: Training and validation loss and accuracy per epoch during training the SER model