CS230

Multispeaker Speech Synthesis with Configurable Emotions

Jiaxiang Fu
jiaxiang@stanford.edu

1 Introduction

Most of widely used speech synthesis systems are only able to generate voices of a limited number of speakers, and they usually require large amount of voice samples from these speakers. Furthermore, they can only generate monotonous speeches with very little variations in tone or emotion.

In this project we propose a way to condition our speech synthesis model on both speaker voice and configurable emotions. Given an input text, the model should generate the corresponding speech audio with a voice similar to the speaker in a given reference audio, and convey an emotion either of a given category or similar to another reference audio. Ideally, the model should work reasonably well with reference audios shorter than 20 seconds.

2 Related work

Tacotron [12] introduced the first end-to-end model for speech synthesis trained directly on text-audio pairs. This allowed us to avoid any hand-crafted feature representations. Tacotron 2 [7] improved upon Tacotron by using WaveNet [10] as its vocoder to generate human-like natural speeches. Importantly, the encoder-decoder seq2seq architecture used in Tacotron 2 made it easy for us to alter the encoding to generate speeches with different styles.

Jia et al. [5] applied transfer learning from speaker verification to generate speeches that mimic the voices of different speakers. They do so by concatenating the speaker embeddings produced by a pre-trained speaker verification model with the encodings produced by Tacotron 2, and fed the concatenated representation back to the decoder and vocoder of Tacotron 2. However, the model does not support varied emotions.

It has been shown that embeddings can also be used to condition the Tacotron decoder to generate speech with different prosody styles [8, 13]. Based on this, Um et al. [9] trained embeddings that encode the emotions of speeches. Further, they proposed a linear interpolation method to control the intensity of emotions in the synthesized speech. This model only supports a single speaker.

Recently, many have developed models for recognizing speech emotions with high accuracy. Particularly, [1] has shown in 2019 that one can obtain results close to state-of-the-art performances with an architecture as simple as a stacked bidirectional LSTM plus fully-connected layers. This is significant to our work as this architecture can be easily adapted to create emotion embeddings.

In this work we apply both the speaker embedding method in [5] and emotion encoding method in [9] to generate speech that mimics the voice of a reference speaker and the emotion of another reference audio.

CS230: Deep Learning, Fall 2021, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

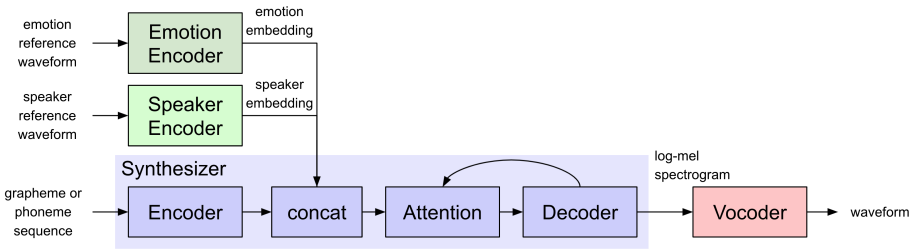


Figure 1: Model overview. All four components are trained independently. This figure is adapted based on Figure 1 in [5]

3 Datasets

3.1 Emotional Speech Dataset - IEMOCAP

The primary dataset we use in this work is the IEMOCAP [2] dataset developed at USC. It is a multimodal dataset that contains speech audios, videos, text transcripts and motion capture of the faces of speakers. It is designed to cover rich emotions in human communication, covering a set of 10 emotion categories: *Anger, Disgust, Excited, Fear, Frustration, Happiness, Neutral, Sadness, Surprise and Other (e.g. apologetic, confused, resigned etc.)*. It contains roughly 10k utterances from 10 speakers and in total approximately 12 hours of speech. The speech audios are recorded in a lab environment where the background noises are minimal. For the purpose of this project, we only use the speech audios, transcripts and the emotion category labels.

We also evaluated the MELD dataset but didn't use it to train our models. See Appendix C for details.

3.2 Multispeaker Speeach Dataset

There are plenty of readily available multispeaker speech synthesis datasets. Notably, VCTK [11] has 44 hours of speech from 109 speakers. LibriSpeech [6] has 436 hours of speech from 1,172 speakers. We do not use these 2 datasets directly to train our models. Instead, they are used in [5] to train their speaker encoder and speech synthesizer, which are then used by us as pretrained models. As such, we do not discuss these 2 datasets in detail. Interested readers may find more details in [6, 11].

4 Approach

4.1 Data Preprocessing

The raw audio files in the IEMOCAP dataset are *.wav* files. We resampled the raw inputs at 16kHz, normalized the audio volume, and removed long silences. Further, similar to [5, 7, 8, 9, 13], we converted the wave signals into Mel Spectrogram. Lastly, we discarded utterances that are too short. The last step removed roughly 29% of the samples, leaving us with 7,130 utterances. Lastly, the remaining samples are split into train/dev/test sets in a stratefied manner with a ratio of 80%/10%/10%.

4.2 Overall Model

As shown in Figure 1, we adopted the encoder-decoder-vocoder architecture with 4 main components similar to [5, 8, 9, 13]. The 4 components are (1) Emotion Encoder, (2) Speaker Encoder, (3) Synthesizer, and (4) Vocoder. Particularly, the Synthesizer includes a text encoder whose outputs are concatenated with the outputs of the emotion and speaker encoders. Together the output of all 3 encoders are used by the attention-based decoder and the vocoder to generate a waveform.

All four of the components are trained independently. For the Speaker Encoder and the Vocoder, we reuse the model proposed in [5] and trained by [4]. For the other two components, we train our own models on the IEMOCAP dataset [2] both from scratch and with transfer learning.

Table 1: Accuracy of emotion encoders when trained using classification. There are 10 categories

Model	Transfer Learning	Top-1 Error	Steps to Converge
LSTM + Linear + Norm	No	39.1%	$\approx$ 160k
LSTM + Linear + Norm	Yes	13.3%	$\approx$ 100k
Stacked BiLSTM in [1]	No	16.1%	$\approx$ 50k

4.3 Emotion Encoder

4.3.1 Training with Classification

We trained our emotion encoder using classification with three different configurations. First, we used the a simple LSTM + Linear + Normalization Layer architecture as the encoder. We added an additional full-connected layer to produce the classification output, and trained it on IEMOCAP from scratch. In the second configuration, we used the same model but initialized it with pretrained weights from the speaker encoder of [5]. In the last approach, we implemented and adapted the stacked bidirectional LSTM model proposed in [1]. In order to produce more meaningful embeddings, we converted the $ReLU$ activation function in the second last layer of the model (i.e. the final layer for the encoder) to $tanh$, and updated the final layer to match our task. We trained the model from scratch as the authors did not publish trained models.

4.3.2 Training with Triplet Loss

We also trained the same emotion encoder architectures using triplet loss. During each epoch, all utterances are sampled as the anchor exactly once, whereas the positive and negative examples are randomly samples given the anchor. We used L_2 distance function and a margin of 1.0. The triplet loss is defined as follows, where A, P, N denotes anchor, positive, negative examples respectively.

$$L(A, P, N) = max\{d(enc(A), enc(P)) - d(enc(A), enc(N)) + margin, 0\}$$

4.4 Synthesizer

The Synthesizer is trained in a strictly supervised manner. Given a text-speech pair, the inputs to the Synthesizer are the texts and the emotion/speaker embeddings, and the labels are the Mel Spectrogram of the speech audios. We used the same loss function proposed in [5] which is the sum of L_2 and L_1 loss, as well as a binary cross entropy loss on the prediction of <END> token. The authors argued that empirically adding L_1 loss made the model more robust to noisy data.

We experimented training the Synthesizer both from scratch and with transfer learning. For transfer learning, due to the additional input of emotion embeddings, some weights of our model are not present in the pretrained models. To tackle this, we repeated the pretrained weights dedicated for the speaker encoder (e.g. encoding projection layer, attention layer etc.) to initialize the weights for the emotion encoder.

5 Result Evaluation & Analysis

5.1 Emotion Encoders

5.1.1 Training with Classification

Table 1 summarizes the classification performances of the emotion encoders proposed in Section 4.3.1. With transfer learning, the base model (i.e. LSTM + Linear + Normalization) not only drastically reduces the prediction error, but also converges much faster. Furthermore, by adopting an architecture proven effective for emotion recognition in [1] we were able to achieve comparable top-1 error without transfer learning, and further reduces the training steps required.

Figure 2 is a visualization of the emotion embeddings projected in 2D by t-SNE and PCA. It shows that while same emotion tend to form cluster among themselves, related emotions partially overlap in a meaningful way. For example, the *Anger* cluster has a considerable overlap with *Frustration*, and so does *Happiness* and *Excited*. On the other hand, *Happiness* has very little overlap with *Sadness*.

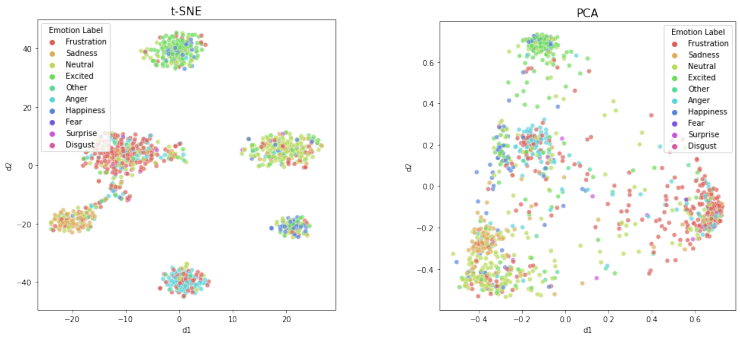


Figure 2: Visualization of emotion embeddings extracted from uetterances in IEMOCAP dataset

Table 2: Evaluation of Synthesizers with Different Configurations

Synthesizer Pretraining	Emotion Encoder Pretraining	Average Loss	Steps to Converge
No	No	0.253	$\approx$ 140k
No	Yes	0.208	$\approx$ 140k
Yes	No	0.321	$\approx$ 15k
Yes	Yes	0.262	$\approx$ 15k

5.1.2 Triplet Loss Training

Despite best effort, no configuration of triplet loss training was found to reliably converge. See Appendix D for more details. We do not really know why. Possible causes include inadequate hyper-parameter searching and insufficient training data. Another contributing factor may be the small number of emotion categories in the training data. As we konw, most prominent uses cases of triplet loss involve a large number of classes. For instance, face recognition datasets usually contain images of thousands or millions of unique individuals. By contrast, we only have 10 emotion categories to train our encoders. Future work is required to identify and resolve this issue.

5.2 Synthesizer

As far as we know, there is no widely accepted standard evaluation metric to measure the "goodness-of-fit" for Mel Spectrograms. Therefore, we use the average loss as a proxy and rely on visualizations to compare the performance of our synthesizers.

Table 2 compares the average loss of the different variants of synthesizers we experimented. To better evaluate the effect of transfer learning, we used the base model for emotion encoder which we have the option of transfer learning. Surprisingly, the synthesizers trained from scratch performed significantly better than those starting from pretrained weights. However, they do take almost 10 times longer to achieve such results. Unsurprisingly, the emotion embeddings trained with transfer learning resulted in better performance in both cases which is aligned with the classification accuracy evaluation.

Figure 3 show an example visualization of the predicted Mel Spectrograms. More examples are available in Appendix B. These diagrams show that our Synthesizer is able to predict accurately the timing of salient sounds or phonemes. The predicted pitch and relative amplitude also reasonably match those of the ground truths. However, there are two apparent problems. First, the patterns in the predictions are blurry and more spread out. This is due to imperfect prediction and expected. Second, the absolute amplitude predicted can differ considerably from the ground truth (Notice the difference in color scale of the two diagrams). This is a major cause for concern as the same is not observed in [5] or [13]. This problem will also be discussed further in the Future Work section.

5.3 Vocoder / Overall Model

We had originally planned to evaluate the final outputs with Mean Opinion Score (MOS), which is also the primary metric used in [5, 7, 10, 9, 12, 13]. Unfortunately, the quality of the final outputs

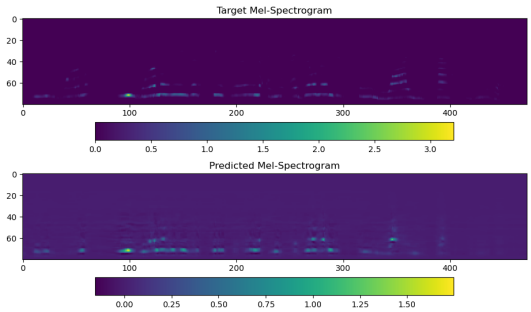


Figure 3: Example synthesized Mel Spectrogram compared to the ground truth. More examples are available in Appendix B

generated by the model is really poor and the audios are mostly unintelligible. As a result, we did not execute this step and have to leave it for future work. See Appendix A for details on the original plan.

Upon further inspection, it is revealed that the main problem lies in the Vocoder. To isolate the problem, we also fed the Vocoder the true Mel-Spectrogram computed from target audios for comparison. Further, we also implemented the Griffin-Lim algorithm [3] and used it to construct speech audios from both the synthesized and true Mel-Spectrograms. By comparing some of the produced samples, we found that the Vocoder is the main problem, as its outputs are still poor when true Mel-Spectrograms are fed as input, whereas the outputs given by Griffin-Lim algorithm are considerably better. This indicates that we should train or fine-tune the Vocoder on our own data in future work, as [10, 7, 5] have all shown that WaveNet (the underlying architecture of our Vocoder) can produce higher quality of audios compared to Griffin-Lim.

However, even though Griffin-Lim can produce better results, the resulting audio quality is still hardly satisfactory. There are two parts to this problem. First, even when fed with true Mel-Spectrogram the outputs are still somewhat difficult to discern for a human. This is due to the inherent weakness of the algorithm and maybe some details in data processing. The second part is due to imperfect predictions by the Synthesizer, which are partially caused by the added complexity introduced by the emotion encoder. [5] observed that after they introduced the speaker encoder to their model, the naturalness of the final output audio decreased as compared to single speaker model. Nevertheless, this issue can likely be mitigated by improving the encoders and the Synthesizer in future work.

6 Conclusion & Future Work

We present a neural network approach that aims to synthesize speech that not only mimics a reference speaker, but also conveys emotions present in a reference audio. We trained our own models for the emotion encoder and the Synthesizer, and reused pretrained models from [5] for the speaker encoder and Vocoder. All except the Vocoder worked well. As a result, we were able to produce high quality meaningful emotion embeddings and fairly accurate Mel-Spectrograms predictions. However, the overall system was not able to produce good quality speech audios primarily due to the deficiency of the Vocoder.

The immediate steps for future work should be to train the Vocoder with our setup and evaluate the overall results with MOS. As discussed in Section 5.3 the Vocoder is the bottleneck of the overall model performance. After that, a few different directions may be explored. First, as mentioned in Section 2, the absolute amplitude predicted by the Synthesizer can sometimes differ significantly from the ground truth. Identifying the reason behind and resolving the problem can definitely improve the performance of the Synthesizer. We believe better data normalization before or during training can be potentially effective and worth trying. Another direction for future work is to further explore triplet loss training as mentioned in Section 5.1.2. In this project we did not find appropriate configurations that allows the model to reliably converge when trained with triplet loss. However, we feel optimistic that given the power of deep learning, some configurations exist that allow the training to converge or even produce better embeddings than training by classification.

Acknowledgement

The authors thank Sarthak Kanodia and Sharan Ramjee for their helpful feedback and suggestions.

References

[1] Bagus Tris Atmaja, Kiyoaki Shirai, and Masato Akagi. Speech emotion recognition using speech feature and word embedding. In *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 519–523, 2019.

[2] C. Busso, M. Bulut, C.C. Lee, A. Kazemzadeh, E. Mower, S. Kim, J.N. Chang, S. Lee, and S.S Narayanan. IEMOCAP: Interactive emotional dyadic motion capture database. *Journal of Language Resources and Evaluation*, 42(4):335–359, December 2008.

[3] Griffin D. and Lim J. Signal estimation from modified short-time fourier transform. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 32(2):236–243, 1984.

[4] Corentin Jemine. Real-time voice cloning. `https://github.com/CorentinJ/Real-Time-Voice-Cloning`.

[5] Ye Jia, Yu Zhang, Ron J. Weiss, Quan Wang, Jonathan Shen, Fei Ren, Zhifeng Chen, Patrick Nguyen, Ruoming Pang, Ignacio Lopez Moreno, and Yonghui Wu. Transfer learning from speaker verification to multispeaker text-to-speech synthesis. In *Proc. Neural Information Processing Systems 31*, pages 4485–4495, 2018.

[6] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. LibriSpeech: an ASR corpus based on public domain audio books. In *Proc. IEEE International Conference Acoustics, Speech and Signal Processing (ICASSP)*, page 5206–5210, 2015.

[7] Jonathan Shen, Ruoming Pang, Ron J. Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, RJ Skerry-Ryan, Rif A. Saurous, Yannis Agiomyrgiannakis, and Yonghui. Wu. Natural TTS synthesis by conditioning wavenet on mel spectrogram predictions. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2018.

[8] RJ Skerry-Ryan, Eric Battenberg, Ying Xiao, Yuxuan Wang, Daisy Stanton, Joel Shor, Ron J. Weiss, Rob Clark, and Rif A. Saurous. Towards end-to-end prosody transfer for expressive speech synthesis with Tacotron. In *Proc. ICML*, 2018.

[9] Se-Yun Um, Sangshin Oh, Kyungguen Byun, Inseon Jang, ChungHyun Ahn, and Hong-Goo Kang. Emotional speech synthesis with rich and granularized control. In *IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, pages 7254–7258, 2020.

[10] Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alexander Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. In *CoRR abs/1609.03499*, 2016.

[11] Christophe Veaux, Junichi Yamagishi, Kirsten MacDonald, and et al. CSTR VCTK corpus: English multi-speaker corpus for CSTR voice cloning toolkit. 2017.

[12] Yuxuan Wang, RJ Skerry-Ryan, Daisy Stanton, Yonghui Wu, Ron J. Weiss, Navdeep Jaitly, Zongheng Yang, Ying Xiao, Zhifeng Chen, Samy Bengio, Quoc Le, Yannis Agiomyrgiannakis, Rob Clark, and Rif A. Saurous. Tacotron: Towards end-to-end speech synthesis. In *Proc. Interspeech*, page 4006–4010, August 2017.

[13] Yuxuan Wang, Daisy Stanton, Yu Zhang, RJ Skerry-Ryan, Eric Battenberg, Joel Shor, Ying Xiao, Fei Ren, Ye Jia, and Rif A. Saurous. Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis. In *Proc. ICML*, volume 80, pages 5180–5189, 2018.

Appendix A Mean Opinion Score Evaluation

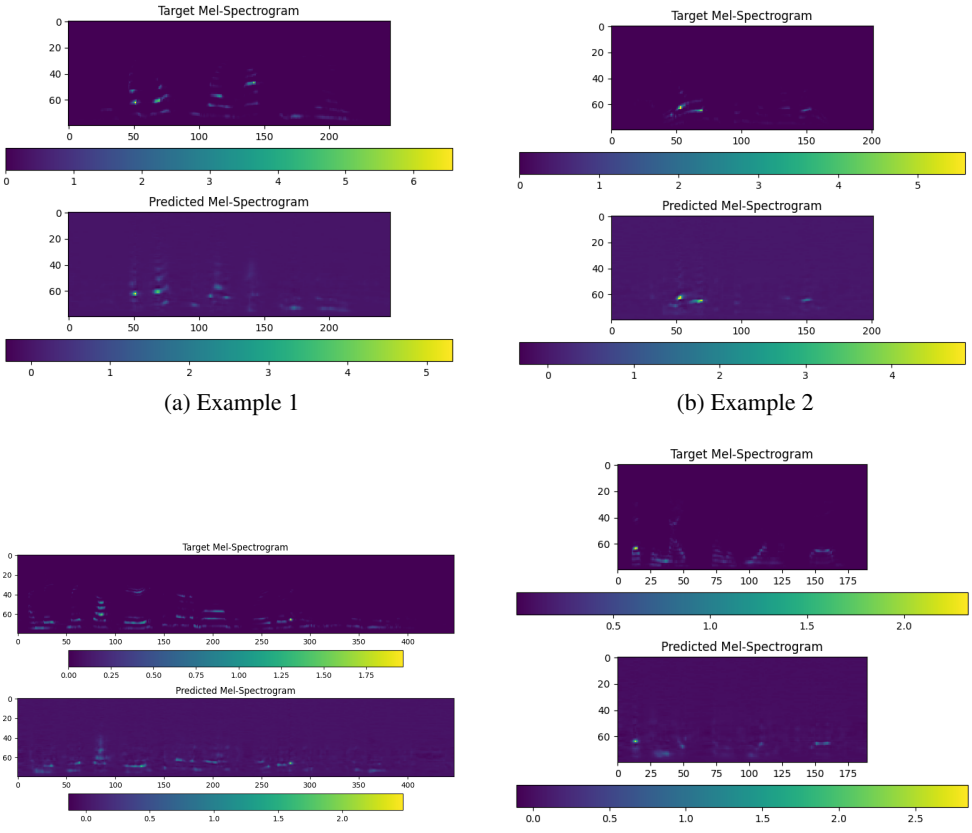
The primary metric we originally planned to used to evaluate the overall model is the Mean Opinion Score (MOS) which is also used in many related work [5, 7, 10, 9, 12, 13]. Below we describe this plan which is a useful reference for future work.

To the best of our knowledge, there is no published model that generate speeches conditioned on both speaker voice and emotion. As such, our baseline model would be the multispeaker TTS model proposed in [5].

To ensure fair comparison, we will present to the human listeners sets of audios produced by the baseline model and our models in randomized orders and without labels. Where applicable we will also include the ground truth in the same set so that human listeners can compare side by side. We evaluate 4 different aspects of the resulting audios - correctness, naturalness, similarity to target speaker and emotion richness.

We will source human listeners primarily from peers in the CS230 class. This is going to be a challenge, but hopefully we can get 5 - 10 human evaluators each of which evaluates 20 sets of audios.

Appendix B More Synthesized Mel Spectrogram Examples



(a) Example 1

(b) Example 2

(c) Example 3

(d) Example 4

Figure 4: More examples of predicted mel-spectrograms vs ground truths

Appendix C Evaluation of MELD Dataset

Besides IEMOCAP, the MELD dataset is another useful multimodal dataset that contains roughly 14 hours of audio, video and corresponding text transcripts. It is entirely collected from the popular

sitcom TV-series *Friends*, and includes about 13k utterances from more than 200 speakers, covering 7 emotion categories: *Anger, Disgust, Sadness, Joy, Neutral, Surprise and Fear*. However, the majority of the speeches are from the 6 main characters of the show.

The main drawback of this dataset is the "canned laughter"[1] mixed in the speech audios. Unlike the noises in a lab environment, these noises are not natural and are often as loud as the speech itself. Due to this reason, MELD is considered a secondary dataset and was not used to train our models at this stage. For future work this dataset can be very useful.

Appendix D Training with Triplet Loss

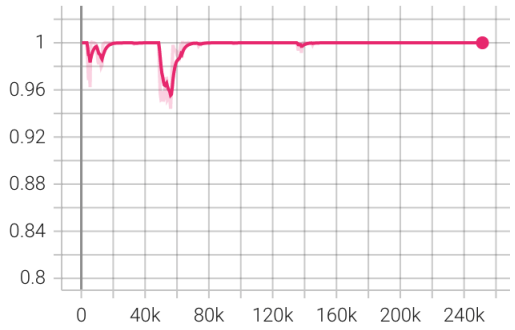


Figure 5: A typical learning curve of emotion encoder training using triplet loss. Despite best effort, no training configuration was found to reliably converge.

Figure 5 shows a typical learning curve of emotion encoder training using triplet loss. The loss stays around the margin (1.0), which implies the encoder wasn't able to learn to differentiate the different emotions at all.

[1]Background noises that are supposed to mimic audience reactions, usually laughter noises