Using Language Model for Implementation of Emotional Text-To-Speech

Mingguang Cao, Jie Zhu
Department of Electronic Engineering, Shanghai Jiaotong University, 800 Dongchuan Road, Shanghai 200240, China
Email:{mg_cao, zhujie}@sjtu.edu.cn

***Abstract*—With the development of neural network, Text-To-Speech (TTS) technology is booming unprecedentedly. The speech generated by modern text-to-speech systems almost sound as natural as human audio. However, the style control of synthetic speech usually limits to discrete emotion type and the emotion embedding which controls emotion transfer contains redundant transcript information. In this paper, we apply pre-trained language model Bidirectional Encoder Representations from Transformer (BERT) to our TTS system to achieve style control and transfer. Using BERT makes our proposed model study the relationship between text representations and acoustic emotion embedding. The experimental results show that our proposed model outperforms baseline Global Style Token (GST)-Tacotron2 model in both parallel and non-parallel style transfer.**

Keywords*—*emotional text-to-speech; style transfer; pre-trained language model

I. Introduction

Text-to-speech (TTS), also knowed as speech synthesis, is the technology which aims to synthesize intelligible and natural speech from raw text. Early speech synthesis techniques mainly include waveform concatenation [1][2] and statistical parametric speech synthesis [3]–[6]. A classic Statistical Parametric Speech Synthesis (SPSS) system usually includes three components which contain a front-end model (convert text symbols to linguistic features), an acoustic model (map linguistic features to acoustic features) and a vocoder (generate speech from acoustic features). In the past decades, this method was widely used in industrial production due to the advantages of robustness and efficiency. However, the generated speech of this method has lower naturalness and intelligibility because of artifacts such as muffled and noisy audio. The voice quality has been largely improved on neural network approaches [5][6] instead of Hidden Markov models (HMM) [4]. Deep Voice [7] still follows the three components in statistical parametric synthesis, but upgrades them with the corresponding neural network models. Furthermore, WaveNet [8], proposed to directly generate waveform from linguistic features, is regarded as the first modern neural TTS model.

Recent end-to-end speech synthesis models surpass traditional parametric systems in many ways, including the use of an encoder to replace linguistic features, a neural vocoder to replace the traditional vocoder, and an attention mechanism for the purpose of end-to-end training. Tacotron [9] is a sequence-to-sequence model which simplifies the traditional speech synthesis pipeline by replacing the production of magnitude spectrograms from text with a single neural network trained from data alone. Like many modern TTS systems, it learns an average prosody of the training data. Afterwards, Tacotron2 [10] gains a great success through refining Tacotron model structure and cascading with a modified WaveNet vocoder. Tacotron and Tacotron2 first generate mel spectrograms from text directly, then synthesize audio samples produced by a vocoder, such as Griffin Lim algorithm or WaveNet. Using an end-to-end network, the quality of synthesized audio is greatly improved and even comparable to human recordings on some datasets. The end-to-end TTS model contains two components, an encoder and a decoder. The encoder maps sequence of text into semantic space and generates a sequence of encoder hidden states, and the decoder, taking these hidden states as context information with an attention mechanism, constructs every mel spectrogram symbol per step. However, these generated models adopt recurrent neural network which limits the parallel processing capability both in training and inference stage. To deal with this problem, some models [11]–[13] leverage Transformer [14] network to replace recurrent neural network in TTS system. Among these models, Fast-Speech 1/2 [12][13] use self-attention mechanism in order to deal with long distance dependency problem on the last previous hidden state and improve parallelization capability. The generated audio of these models is more robust than that of sequence-to-sequence models. However, because the audio generated by these models only contains neutral prosody is limited in many scenarios like AI voice assistants and navigation systems, there has been an increasing interest in emotional TTS and the method to control the generated speech style.

In expressive TTS, the speaking style is modeled in supervised or unsupervised manner. Lee et al. [15] proposed an emotional end-to-end speech neural speech synthesizer, controlling speech emotion with discrete label. Luong et al. [16] introduces a DNN-based text-to-speech system which takes speaker, gender and age codes as inputs in order to modify synthetic speech characteristics based on the input codes. Lorenzo-Trueba et al. [17] evaluates a large-scale corpus of emotional speech from a professional voice actress for the purpose of investigating different representation for modeling and controlling multiple emotions in DNN-based speech synthesis. However, the control of speech emotion is only limited to the emotion category which, we have predefined and synthetic speech cannot convey a variety of emotion. With the rapid progress of sequence-to-sequence architecture, especially Tacotron family, reference-based style transfer has emerged as another solution with great potential to solve this problem. The reference-based model learns a latent style embedding from the reference audio and generate speech which matches the prosody of the reference speech even if their speakers are different from each other. To model reference speech as style input, there has evolved a plenty of work, such as Global Style Token (GST) [18][19], Variational Autoencoder (VAE) [20][21] and their variants. Global Style Token (GST) [19] introduces a reference encoder that extracts style embedding from the acoustic signal and encodes various speaking styles into a fixed number of tokens. Variational Autoencoder (VAE) [21] infers style representation through the recognition of VAE, then feeds it into TTS network to guide the style in synthesizing speech.

The remaining part of this paper proceeds as follows. Section II introduces related work. The overview and each component of the proposed model are described in Section III. Experiments and results are reported in Section IV. Lastly, the conclusion and future work are covered in Section V.

II. Related Work

In this section, we first introduce reference-based TTS model, followed by a brief description about language model in TTS.

A. Reference-based TTS model

The reference-based TTS model aims to synthesize speech whose style is transferred from reference audio. The most straightforward way is to obtain style embedding from reference speech and use it as condition control to guide speech synthesizing. Skerry-Ryan et al. [18] proposes the concept of prosody embedding and merges prosody encoder into Tacotron architecture for computing low-dimension information of reference speech. The embedding captures audio features independent of speech information and specific speaker features such as accent, intonation, and speech rate. At the inference stage, we can use this embedding to perform prosody transfer and produce speech from a completely different speaker's voice. The embedding can also transfer temporally aligned precise prosody from one phrase to a slightly different one, even though the reference and target phrases are similar in length and structure. On the basis of previous work [18], global style token (GST) [19] is an updated method to learn the style representation by encoding various speaking style into a fixed number of tokens. By adding an additional attention mechanism to Tacotron, it enables it to express the prosody embedding of any speech segment as a linear combination of a fixed set of base embedding. The attention weights represent the contribution of each style token, and style embedding is made up of the weighted sum of all style tokens. In the training stage, each token is randomly initialized in an unsupervised manner. During the inference step, we can use a different audio signal or specify the attention weights of style tokens to achieve style transferring and controlling. Um et al. [22] introduces an inter-to-intra emotional distance ratio algorithm to the embedding vectors which can balance the distance between the target emotion category and the other categories. Li et al. [23] is also a GST-based method for expressive TTS, where the authors insert two classifiers into GST-Tacotron2 [19] model for improving emotion discrimination ability of emotion and deliver emotional speech with preferred strength.

B. Language model in TTS

Language model (LM) is often used in many natural language processing applications, such as speech recognition, machine translation, syntactic analysis, handwriting recognition and information retrieval. With the development of neural network, language model becomes increasingly powerful and is exploited in TTS system to improve the quality of synthetic speech. Jia et al. [24] introduce a new encoder model which takes both phoneme and grapheme representations of text as input and is trained in a self-supervised manner. Fang et al. [25] uses BERT [26] in TTS system to know when to stop decoding and help faster converge during training. Zhang et al. [27] employs BERT in a unified front-end model for the purpose of improving polyphone disambiguation accuracy. In [28], style tag makes synthetic audio more interpretable and natural compared with style index of reference speech. Shin et al. [29] proposes a style encoder which models the relationship between the text embedding and speech embedding with a pre-trained language model.

III. Proposed Model

Our proposed model architecture is shown in Figure 1. The proposed model is based on Tacotron2 with an emotion recognition network, an additional network and a semantic network.

A. Encoder

The encoder is made up of a character embedding layer, 3 convolutional layer and a single bi-directional LSTM [30].

Because a character is represented as a 512-dimensional one-hot vector, the input character sequence which contains n characters is converted to a n × 512-dimensional character embedding through character embedding layer. In order to capture a longer range of contextual information and obtain features of character sequence, the character embedding is then passed through 3 convolutional layers and each layer has 512 filters where each cover 5 characters, followed by batch normalization and ReLU activation. The output of the last convolutional layer is sent to a bi-directional LSTM layer which contains 512 units to generate encoded features. After the above operations, the encoder finally encodes the input character sequence into a 512-dimensional hidden feature vector.

B. Decoder

As we all konw, the decoder is an autoregressive recurrent neural network trained from the input sequence of the encoder to predict the output mel spectrogram. The mel spectrogram at the previous moment is first passed into a pre-net which is comprised of two fully connected layer with 256 hidden ReLU activations. It is important for the pre-net to learn attention alignment mechanism. The output of pre-net and the anttention vector are connected with each other and sent to two one-way LSTM layers with 1024 units. The output vector of LSTM is concatenated with the attention context vector output by the encoder, and then passed to a linear projection

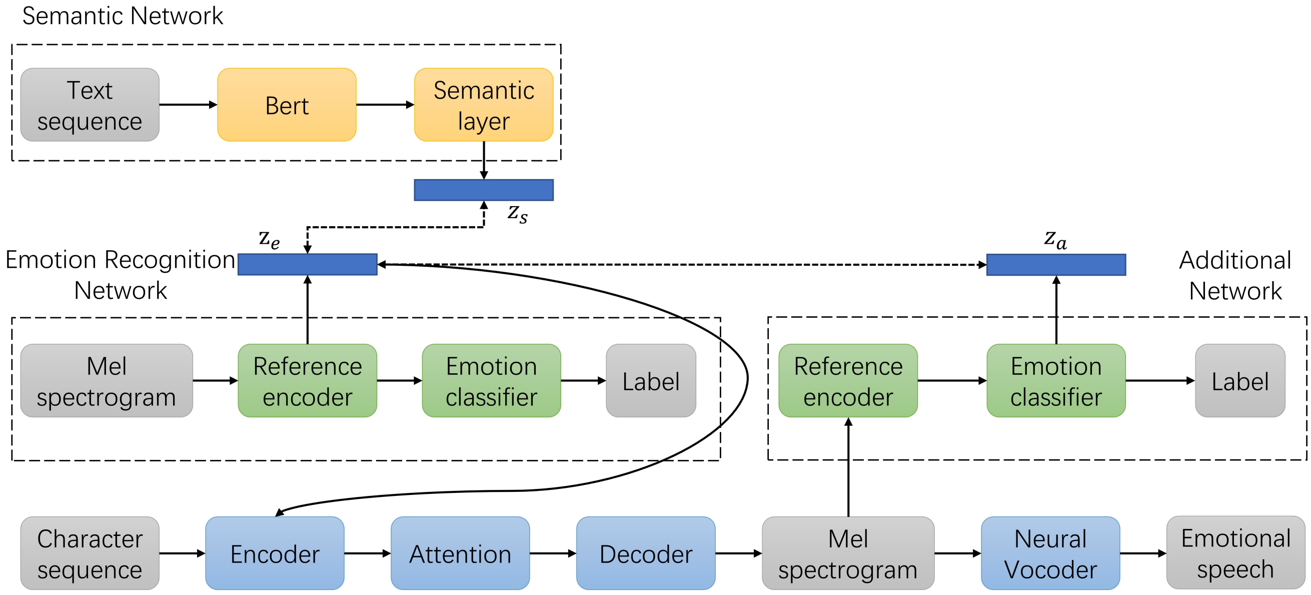


Figure. 1. Model Architecture

layer to predict the current mel spectrogram frame. Finally, the post-net containing 512 filters with shape 5 × 1 with batch normalization takes the predicted mel spectrogram as input to add the residual to previous mel spectrogram for improving reconstruction ability.

C. Emotion Recognition Network

From the top left of Figure 1, the emotion recognition network contains a reference encoder and an emotion classifier.

1) Reference encoder: The reference encoder we adopted in our model is the same in [18]. It is composed of six 2D convolutional network where each layer equipped with batch normalization has 3 × 3 filters with 2 × 2 stride. The output of convolutional network is then passed through a GRU [31] layer, and we use a fully connected layer followed by a tanh activation in order to map the final GRU state to our desired 128-dimensional embedding.

2) Emotion classifier: We use emotion classifier followed by reference encoder to facilitate the discrimination ability of emotion types. The classifier consists 5 fully connected layers with tanh activation. In the classifier, the size of first layer is 128-unit and that of the remaining layers is 256-unit. For the downstream emotion classifier task, a softmax layer is applied to produce the probability of each emotion category, such as neutral, happy and angry. The output of third layer and the hidden feature vector from the encoder are concatenated for teach-forcing speech waveform generation.

D. Additional Network

The additional network shares the same structure with the emotion recognition network, as is shown in the top right of Figure 1. In detail, we plug an additional network to the decoder , which enables the predicted mel spectrogram to identify emotion category. The output of third layer from additional network acts as an emotion embedding of the generated speech and is compared with the emotion representation of input audio for better training and optimizing.

E. Semantic Network

The semantic network is composed of a pre-trained BERT model and semantic layer. We use this network to map text sequence to semantic representations, which aims to remove text-related information from acoustic features and leverage transcript dataset to assist TTS training.

1) BERT: BERT is of great significance to a large amount of NLP tasks. It consist a stack of Transformer [14] blocks and is trained with Mandarin text data. The input text sequence is made up of many characters where each is transformed to a linguistic feature, and is encoded to capture contextual information from Mandarin text by BERT. BERT can be trained in two unsupervised manners, one is mask language modeling where we randomly replace the token in each training sequence with a [MASK] token and then predict the original word at the [MASK] position, and the other is next sentence prediction where the model has the ability to understand the relationship between two sentence in many downstream tasks.

2) Semantic layer: To adapt to downstream task, we design the semantic layer that is connected with BERT in order to modeling the input text sequence. Similar to Emotion classifier, the semantic layer is built with 3 fully connected layer followed by tanh activation. The output of semantic layer is used to reduce impact on acoustic representations and focus on emotion dimension of the synthetic speech.

F. Training and inference

During training, we use five loss terms summed as a total loss in our model. The loss function of the basic acoustic model, referred as L_{tac} which is followed with Tacotron2 [10], is the Mean Square Error (MSE) between the input ground-truth mel spectrogram and the predicted mel spectrogram. To make the reference encoder only extract emotion features, we adopt L_{emo_sem} that calculates the loss between the emotion vector z_e extracted from the emotion recognition network and the semantic embedding z_s extracted from semantic network.

$$L_{emo\_sem} = \sum_{i=1}^{N} (z_{ei} - z_{si})^2 \tag{1}$$

To improve the distinguish capability of the generating speech, the loss function, L_{emo_add}, is determined by MSE between emotion embedding z_e fetched from the emotion recognition network and addition embedding z_a fetched from the additional network as follows.

$$L_{emo\_add} = \sum_{i=1}^{N} (z_{ei} - z_{ai})^2 \tag{2}$$

Besides, L_{cls_src} and L_{cls_pre} denote the cross entropy loss for the source audio classifier in emotion recognition network and the predicted audio classifier in additional network, respectively. The total loss of the proposed model is:

$$L = L_{tac} + L_{emo\_sem} + L_{emo\_add} + L_{cls\_src} + L_{cls\_pre} \tag{3}$$

In the inference stage, we use reference speech or emotional vector to achieve style control and transfer. For the emotional vector method, emotional vector v_e is determined by averaging the samples of the corresponding emotion category as follows:

$$v_e = \frac{1}{N_e} \sum_{x_i \subset X_e} x_i \tag{4}$$

where X_e represents all the weight vectors of the emotional category and N_e and x_i donate the number of weight vectors and weight vector belonging to the emotional category, respectively.

IV. Experiments and results

In this section, we evaluate the performance of our proposed model in parallel style transfer and non-parallel style transfer through subjective evaluations and objective evaluations.

A. Dataset and settings

In our experiments, we use a high-quality emotional speech corpus which is recorded by a Chinese female professional speaker and contains <text, audio, emotion label> tuples. The emotions of all the speeches are classified by 2 categories (neutral and happy). The corpus consists of 12000 utterances (12 hours), among which 10000 (10 hours) belong to the neutral emotion and the happy emotion has the remanent 2000 utterances (2 hours). We down-sample all the recordings from 48kHz to 22.05 kHz for model training. 80-dimensional mel spectrogram is extracted from the dataset as acoustic features. The frame length and frame shift are set to 50ms and 12.5ms, respectively.

The experimental environment configuration is shown in the Table I.

TABLE I
Experimental environment parameters

Operating System	Ubuntu18.04
Graphics Card	NVIDIA GeForce RTX 3090
Memory	24G
Python	3.7.6
PyTorch	1.8.1
CUDA	11.1

We train our model for 500 epochs with a batch size of 32 because using BERT makes our model size larger. We use the Adam [32] optimizer with a learning rate of 1e-5 to learn the parameters in a single GeForce GTX 3090 GPU. In the reference encode, the number of GRU hidden units is set to 128. As for speech generation, we build a WaveRNN [33] as the vocoder trained by ground-truth mel spectrogram.

B. Text preprocessing

In our acoustic model, we use traditional Chinese representation as the input sequence for speech generation. Figure 2 shows the process of translating traditional Chinese characters into phonemes. We first represent Chinese characters with Chinese phonetic alphabet, then convert Chinese phonetic alphabet to combination of initials and finals instead of English alphabet.

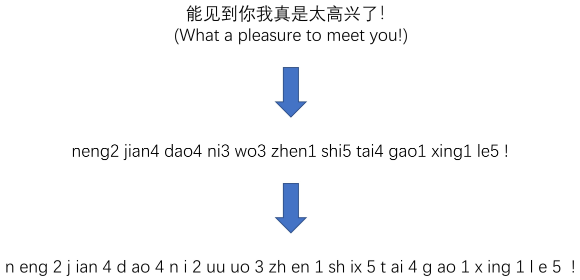

能见到你我真是太高兴了！
(What a pleasure to meet you!)
neng2 jian4 dao4 ni3 wo3 zhen1 shi5 tai4 gao1 xing1 le5 !
n eng 2 j ian 4 d ao 4 n i 2 uu uo 3 zh en 1 sh ix 5 t ai 4 g ao 1 x ing 1 l e 5 !

Figure. 2. Example of translating traditional Chinese characters into phonemes

C. Subjective evaluations

To subjectively evaluate the performance of our model, we compare our proposed model with baseline model on the Mean Opinion Score (MOS) and ABX preference subjective tests. Some samples can be found from https://light-cao.github.io/.

We evaluate the naturalness of the generated speech utterances by using the Mean Opinion Score (MOS) test. Ten native Mandarin speakers are asked to stay in a quiet room and listen recordings with noise canceling headphones, and then make their judgements of the performance with five-point scale (1: Bad, 2: Poor, 3: Fair, 4: Good, 5: Excellent). We randomly select 10 sentences from each emotion category for both parallel style transfer and non-parallel style transfer experiments with our proposed model and baseline model. Parallel style transfer means the transcript of the synthetic speech matches that of the reference audio. Non-parallel style transfer refers to synthesizing the audio with arbitrary text in the prosodic style of the reference signal.

The MOS test results in Table II confirm that our proposed model performs better than the baseline model both in parallel style transfer and non-parallel style transfer. Parallel style transfer outperforms non-parallel style transfer in speech quality because the transcript seen during training is beneficial to better modeling the synthetic audio.

TABLE II
MEAN OPINION SCORE(MOS) WITH 95% CONFIDENCE INTERVALS ON PARALLEL AND NON-PARALLEL STYLE TRANSFER

	parallel transfer	non-parallel transfer
Ground truth	4.25 ± 0.15	-
GST-Tacotron2	3.90 ± 0.16	3.57 ± 0.19
Proposed	4.07 ± 0.16	3.79 ± 0.18

To demonstrate that our model can control style transfer, we conduct a ABX preference test with the baseline GST-Tacotron2 and our proposed model. In this test, the participants are provided a fair number of samples from baseline and the proposed model and rated which sample is as expressive as the reference speech. If there is no obvious difference between the two samples, they can choose no preference. The results in Figure 3 show that our proposed model outperform the baseline model in both neutral and happy emotion category.

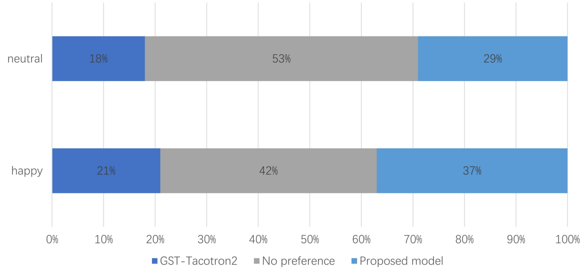
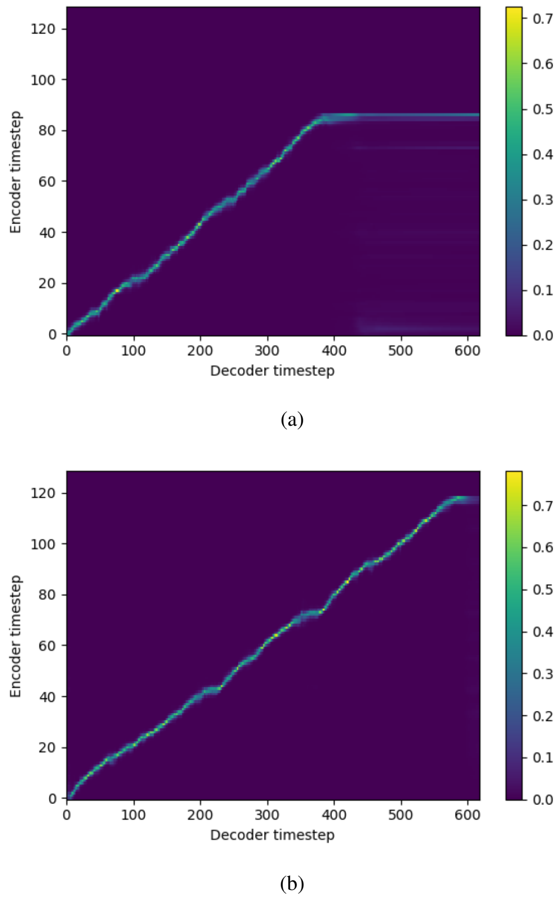


Figure. 3. ABX preference test results on two emotion categories between baseline model and the proposed model

D. Objective evaluations

We visualize the attention alignment of the decoder in Figure 4 and check if the attention mechanism learns how to align between the text sequence and the reference audio. In Figure 4, the attention alignment of our proposed model is slightly brighter than that of the baseline GST-Tacotron2 in many places, which indicates the proposed model surpass the baseline model. From the analogous shape of the proposed attention, we can see that our proposed model could align the reference speech to the text well.



(a)

(b)

Figure. 4. Comparison on attention alignment between text and speech. (a) From the baseline GST-Tacotron2. (b) From our proposed model

V. CONCLUSION AND FUTURE WORK

In our work, we utilized semantic network in our model to control and transfer style. In order to deliver the emotion more accurate, we inserted two classifiers after the reference encoder to enhance the emotion discriminative ability of the emotion embedding and the predicted mel spectrogram. Compared with the baseline model, the proposed model improved the quality of synthetic speech and achieves excellent performance on parallel and non-parallel style transfer. Besides, our proposed model could align the reference speech to the text better than the baseline model. In the future work, we want to improve the model for excluding the transcript information in acoustic features more precisely and experiment on more emotion categories.

REFERENCES

[1] A. J. Hunt and A. W. Black, “Unit selection in a concatenative speech synthesis system using a large speech database,” in *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, vol. 1. IEEE, 1996, pp. 373–376.

[2] A. BLACK, “Automatically clustering similar units for unit selection in speech synthesis,” *Proc. EUROSPEECH, Sep 1997*, 1997.

[3] H. Zen, K. Tokuda, and A. W. Black, “Statistical parametric speech synthesis,” *speech communication*, vol. 51, no. 11, pp. 1039–1064, 2009.

[4] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, “Speech parameter generation algorithms for hmm-based speech synthesis,” in *2000 IEEE international conference on acoustics, speech, and signal processing. Proceedings (Cat. No. 00CH37100)*, vol. 3. IEEE, 2000, pp. 1315–1318.

[5] H. Ze, A. Senior, and M. Schuster, “Statistical parametric speech synthesis using deep neural networks,” in *2013 ieee international conference on acoustics, speech and signal processing*. IEEE, 2013, pp. 7962–7966.

[6] Y. Qian, Y. Fan, W. Hu, and F. K. Soong, “On the training aspects of deep neural network (dnn) for parametric tts synthesis,” in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 3829–3833.

[7] S. Ö. Arık, M. Chrzanowski, A. Coates, G. Diamos, A. Gibiansky, Y. Kang, X. Li, J. Miller, A. Ng, J. Raiman *et al.*, “Deep voice: Real-time neural text-to-speech,” in *International conference on machine learning*. PMLR, 2017, pp. 195–204.

[8] A. v. d. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, “Wavenet: A generative model for raw audio,” *arXiv preprint arXiv:1609.03499*, 2016.

[9] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio *et al.*, “Tacotron: Towards end-to-end speech synthesis,” *arXiv preprint arXiv:1703.10135*, 2017.

[10] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan *et al.*, “Natural tts synthesis by conditioning wavenet on mel spectrogram predictions,” in *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2018, pp. 4779–4783.

[11] N. Li, S. Liu, Y. Liu, S. Zhao, and M. Liu, “Neural speech synthesis with transformer network,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 6706–6713.

[12] Y. Ren, Y. Ruan, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, “Fastspeech: Fast, robust and controllable text to speech,” *Advances in neural information processing systems*, vol. 32, 2019.

[13] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, “Fastspeech 2: Fast and high-quality end-to-end text to speech,” *arXiv preprint arXiv:2006.04558*, 2020.

[14] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” *Advances in neural information processing systems*, vol. 30, 2017.

[15] Y. Lee, A. Rabiee, and S.-Y. Lee, “Emotional end-to-end neural speech synthesizer,” *arXiv preprint arXiv:1711.05447*, 2017.

[16] H.-T. Luong, S. Takaki, G. E. Henter, and J. Yamagishi, “Adapting and controlling dnn-based speech synthesis using input codes,” in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 4905–4909.

[17] J. Lorenzo-Trueba, G. E. Henter, S. Takaki, J. Yamagishi, Y. Morino, and Y. Ochiai, “Investigating different representations for modeling and controlling multiple emotions in dnn-based speech synthesis,” *Speech Communication*, vol. 99, pp. 135–143, 2018.

[18] R. Skerry-Ryan, E. Battenberg, Y. Xiao, Y. Wang, D. Stanton, J. Shor, R. Weiss, R. Clark, and R. A. Saurous, “Towards end-to-end prosody transfer for expressive speech synthesis with tacotron,” in *international conference on machine learning*. PMLR, 2018, pp. 4693–4702.

[19] Y. Wang, D. Stanton, Y. Zhang, R.-S. Ryan, E. Battenberg, J. Shor, Y. Xiao, Y. Jia, F. Ren, and R. A. Saurous, “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.

[20] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” *arXiv preprint arXiv:1312.6114*, 2013.

[21] Y.-J. Zhang, S. Pan, L. He, and Z.-H. Ling, “Learning latent representations for style control and transfer in end-to-end speech synthesis,” in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6945–6949.

[22] S.-Y. Um, S. Oh, K. Byun, I. Jang, C. Ahn, and H.-G. Kang, “Emotional speech synthesis with rich and granularized control,” in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7254–7258.

[23] T. Li, S. Yang, L. Xue, and L. Xie, “Controllable emotion transfer for end-to-end speech synthesis,” in *2021 12th International Symposium on Chinese Spoken Language Processing (ISCSLP)*. IEEE, 2021, pp. 1–5.

[24] Y. Jia, H. Zen, J. Shen, Y. Zhang, and Y. Wu, “Png bert: Augmented bert on phonemes and graphemes for neural tts,” *arXiv preprint arXiv:2103.15060*, 2021.

[25] W. Fang, Y.-A. Chung, and J. Glass, “Towards transfer learning for end-to-end speech synthesis from deep pre-trained language models,” *arXiv preprint arXiv:1906.07307*, 2019.

[26] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” *arXiv preprint arXiv:1810.04805*, 2018.

[27] Y. Zhang, L. Deng, and Y. Wang, “Unified mandarin tts front-end based on distilled bert model,” *arXiv preprint arXiv:2012.15404*, 2020.

[28] M. Kim, S. J. Cheon, B. J. Choi, J. J. Kim, and N. S. Kim, “Expressive text-to-speech using style tag,” *arXiv preprint arXiv:2104.00436*, 2021.

[29] Y. Shin, Y. Lee, S. Jo, Y. Hwang, and T. Kim, “Text-driven emotional style control and cross-speaker style transfer in neural tts,” *arXiv preprint arXiv:2207.06000*, 2022.

[30] A. Graves and A. Graves, “Long short-term memory,” *Supervised sequence labelling with recurrent neural networks*, pp. 37–45, 2012.

[31] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, “Learning phrase representations using rnn encoder-decoder for statistical machine translation,” *arXiv preprint arXiv:1406.1078*, 2014.

[32] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

[33] N. Kalchbrenner, E. Elsen, K. Simonyan, S. Noury, N. Casagrande, E. Lockhart, F. Stimberg, A. Oord, S. Dieleman, and K. Kavukcuoglu, “Efficient neural audio synthesis,” in *International Conference on Machine Learning*. PMLR, 2018, pp. 2410–2419.