Joint Workshop for the Blizzard Challenge and Voice Conversion Challenge 2020
30 October 2020, Shanghai, China

Submission from SCUT for Blizzard Challenge 2020

Yitao Yang, Jinghui Zhong, Shehui Bu

School of Computer Science & Engineering, South China University of Technology, Guangzhou, China

ytyang_scut@163.com, jinghuizhong@scut.edu.cn, bushehui@scut.edu.cn

Abstract

In this paper, we describe the SCUT text-to-speech synthesis system for the Blizzard Challenge 2020 and the task is to build a voice from the provided Mandarin dataset. We begin with our system architecture composed of an end-to-end structure to convert acoustic features from textual sequences and a WaveRNN vocoder to restore the waveform. Then a BERT-based prosody prediction model to specify the prosodic information of the content is introduced. The text processing module is adjusted to uniformly encode both Mandarin and English texts, then a two-stage training method is utilized to build a bilingual speech synthesis system. Meanwhile, we employ forward attention and guided attention mechanisms to accelerate the model's convergence. Finally, the reasons for our inefficient performance presented in the evaluation results are discussed.

Index Terms: Blizzard Challenge 2020, speech synthesis, end-to-end speech synthesis, bilingual speech synthesis

1. Introduction

Since 2005, the Blizzard Challenge has been organized annually to practically verify the effectiveness of research techniques for speech synthesis. The Blizzard Challenge 2020 has two tasks: (1) Hub task to build a speech synthesis system with the provided Mandarin data. (2) Spoke task to build a speech synthesis system with the provided Shanghainese data. And we select the former as our task.

The unit selection and statistical parametric speech synthesis (SPSS) have been widely applied in the speech synthesis field in the last decade. For the phoneme sequences predicted from the input texts, the unit selection system selects the appropriate waveform fragments for each phoneme from a sufficiently large speech database and concatenates them into the target speech[1, 2, 3]. By using real speech segments, the unit selection system can produce highly natural speech. However, it is hard for the unit selection system to eliminate the discontinuity between speech fragments, resulting in incoherent speech. Also, constructing a database that covers diverse phonetic and prosodic information is costly. Unlike the concatenative approach, the HMM-GMM based system[4], a representative of SPSS, aims to map the text space to the acoustic space and achieve fluent pronunciation on arbitrary text. Since the HMM model is a short-term probability statistical model, although the synthesized speech is smooth, the intonation is rigid and unnatural.

In recent years, the SPSS based on deep neural networks (DNN) have became a hot research topic. Approaches such as [5, 6] have shown that the DNN can be utilized to overcome the limitations of conventional HMM-based systems. To further simplify the structure of the speech synthesis system, end-to-end speech synthesis systems have been proposed in the past few years, such as Tacotron 1[7], Tacotron 2[8], Deep voice 2[9], and Clarinet[10]. These end-to-end systems mainly have an attention-based sequence-to-sequence framework, which directly converts textual embedding into acoustic features. Finally, a back-end vocoder is employed to generate the waveform. The end-to-end model can easily encode prosodic information, thereby implementing a controllable and expressive text-to-speech system[11, 12, 13, 14]. Due to the superiority of the end-to-end architecture, we adopt it as the backbone of our system for the Blizzard Challenge 2020.

As for vocoder, the Wavenet[15] with autoregressive structure has yielded extreme performance in waveform restoring, despite its slow sampling speed. Following the significant progress made by WaveNet, researchers have proposed various DNN-based vocoders to take account of both speed and quality of their productions, such as Fast WaveNet[16], Parallel WaveNet[17], WaveRNN[18], and WaveGlow[19]. For the competition, we select WaveRNN as the vocoder to maintain speech quality under limited computing resources.

The rest of this paper is organized as follows: We describe our system architecture in section 2 and the experiment details will be presented in section 3. Then, section 4 shows the evaluation results and investigates the shortcomings of our system. Finally, we conclude our paper in section 5.

2. System Description

2.1. Overall Architecture

As shown in Figure 1, our system uses the same encoder-decoder architecture as Tacotron 2 does, followed by a back-end vocoder to achieve high-fidelity speech synthesis. Functionally, the encoder extracts linguistic features from phoneme sequences. Then the frame-level mel-spectrogram, a time-aligned acoustic feature, is output by decoder, which consumes the historical embedding and the alignment context supplied by the attention mechanism in each step. Finally, the vocoder restores the mel-spectrograms to waveforms. The method has been repeatedly verified to be easily trained and generate high-quality acoustic features. Since the Mandarin dataset provided by the Blizzard Challenge committee contains not only Chinese sentences, but also some English words and letters, we hope to build a speech synthesis system that can support both Chinese and English. To support bilingual speech synthesis, however, the above method needs modification. First, we implemented a front-end module that can process both Mandarin and English characters. Then a two-stage training with different corpora is performed in the training phase, so that we can uniformly embed Mandarin and English texts and generate bilingual acoustic features.

10.21437/VCC_BC.2020-6

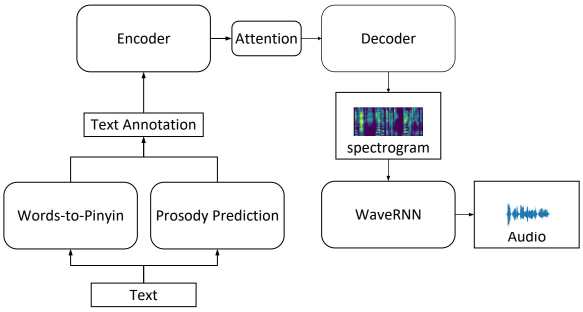

Figure 1: *Overall Architecture of our system*

2.2. Data

The Blizzard Challenge committee provides a 9.6-hour Mandarin dataset of a single male speaker including 4365 sentences and corresponding transcription. The audio has been segmented and the transcription has also been normalized in the dataset, Thus, we can use it as training corpus directly without complicated pre-processing. To improve the performance of the speech synthesis system, the Databaker dataset[20], which has 12-hour record of a single female speaker including 10000 Mandarin sentences, is introduced as external data for training. In addition, to build a bilingual speech synthesis system, the English corpus is indispensable to English pronunciation training, we also chose the 24-hour LJ Speech English dataset[21] recorded by a single female speaker for training. All audios in the data set are resampled to 22.5 kHz and their mel-spectrograms are extracted using the librosa library[22]. Also, the transcriptions are processed by the front-end module before entering the encoder module.

2.3. Front-End processing

The front-end processing module in our system consists of two components: (1) Grapheme to phoneme(G2P) module to convert Mandarin or English characters to phoneme sequences. (2) Prosody prediction model for extracting prosodic information from Mandarin texts and it allows us to generate more expressive speech with a controllable style.

2.3.1. Grapheme to phoneme

This module aims to organize texts into phonemes, which are the smallest units of speech that make one waveform sound different from another. For Mandarin characters, the phonemes are named Pinyin, which can be divided into three parts: initial, final, and tone. The tones always function on the finals, and numbers 1 to 5 are used to denote different Mandarin tones. Then plain initials and variable-tonal finals are combined to form the pronunciation of Mandarin characters. Therefore, in this system, we also give the initial a tone of 0 to indicate its flat tone. The Pypinyin toolkit we used in the Pinyin sequence extracting may output inaccurate labels for polyphonic characters, so manual correcting is required. For English texts, our G2P module maps English words into phonemes by consulting the BEEP pronunciation dictionary[23], and ”e_” is inserted in front of the English phonemes as a mark to distinguish Pinyin and English phonemes. We also mark the tones of English phonemes as 6 for unified representation. Table 1 presents the transformation from texts to phoneme sequences.

2.3.2. Bert-based prosody prediction

It is common sense that highly natural speech usually involves intricate prosodic information relevant to the corresponding texts. However, the end-to-end speech synthesis system will inflexibly follow the implicit prosodic pattern derived from the training data if it does not model the mapping between the text's latent prosodic information and acoustic target. Under these conditions, we introduced a prosody prediction model to explicitly specify the prosody annotation of the input texts. Since the English text in the Blizzard Challenge 2020 dataset usually appears in the form of single letters or words while the Mandarin characters account for the majority of sentences, our prosody prediction model is only responsible for Mandarin texts' prediction.

In Mandarin, the prosody structure of a sentence includes syllable, prosodic word, prosodic phrase, and intonational phrase, which are marked with ”#0” to ”#4” in the texts. At the same time, ”#5” is the label for English words in the texts. For example, the corresponding labeled sequence of the transcript “AI算命的背后是一条吸金的生意链” is ”A #5 I #5 算 #0 命 #0 的 #1 背 #0 后 #2 是 #0 一 #0 条 #1 吸 #0 金 #0 的 #1 生 #0 意 #0 链 #4”. Since the prosody prediction can be regarded as a sequence tagging task, the Bert-based BiLSTM-CRF model, which is widely employed in end-to-end sequence tagging, is ideal for prosody prediction tasks.

The first component of our prosody prediction system is a pre-trained language representation model, namely BERT[24]. The BERT is trained in massive unlabeled corpus and it provides reasonable embedding for texts. The next module is a 256-unit bidirectional long short term memory (BLSTM) layer to construct the interaction among the characters in sentences. And finally, as same as [25], a conditional random field (CRF) layer is employed to establish the relationship between the target tags. The prosody prediction process is presented in Figure 2. Relying on 10000 sentences with prosodic annotations in the transcript of Databaker corpus, we jointly trained the pre-trained Bert model named BERT-Base from [26] and the other components of our model.

Instead of pushing the prosodic embedding into the decoder module as usual, we attach the prosodic labels to phoneme sequences and send them to the encoder module together. The adjustment is an adaptation to the two-stage training method which will be described below. The detailed front-end process is shown on Table 1.

Table 1: *An example of front-end pre-processing*

Procedure	Output
Original	AI算命的背后是一条吸金的生意链.
Grapheme to phoneme	e_ey 6 e_ay 6 s 0 uan 4 m 0 ing 4 d 0 e 5 b 0 ei 4 h 0 ou 4 sh 0 ih 4 y 0 i 1 t 0 iao 2 x 0 i 1 j 0 in 1 d 0 e 5 sh 0 eng 1 y 0 i 4 l 0 ian 4 .
Prosody Annotation	e_ey 6 #5 e_ay 6 #5 s 0 uan 4 #0 m 0 ing 4 #0 d 0 e 5 #1 b 0 ei 4 #0 h 0 ou 4 #2 sh 0 ih 4 #0 y 0 i 1 #0 t 0 iao 2 #1 x 0 i 1 #0 j 0 in 1 #0 d 0 e 5 #1 sh 0 eng 1 #0 y 0 i 4 #0 l 0 ian 4 #4 .

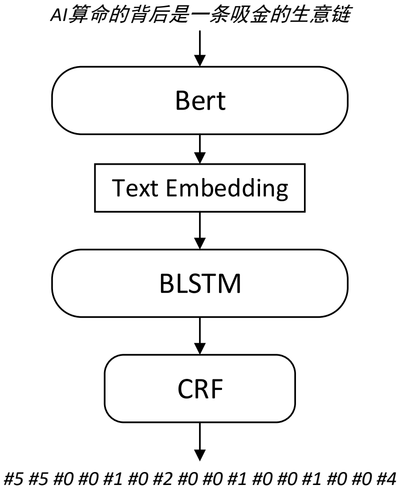

Figure 2: *The process of BERT-Based BLSTM-CRF prosody prediction*

2.4. End-to-end speech synthesis model

2.4.1. Encoder

When the phoneme sequences output by the front-end module are transmitted to the encoder module, it will be first transformed into one-hot embedding based on the phonemes set we used. Then a linear layer is utilized to project the one-hot vectors into 512-dimensional embeddings, followed by a stack of 1-D convolution layers with 512 filters and a single BLSTM network with 256 units in each direction. Finally, the hidden states of BLSTM at all times are output as encoded features which have the same length as the phoneme sequences.

2.4.2. Attention mechanism

The attention mechanism helps the decoder module to automatically capture the encoded outputs that should be referenced at each step. Since the attention mechanism takes responsibility for the alignment between input text sequences and output acoustic features, the naturalness and coherence of generated speech varies with the attention's validity.

To acquire an available attention module fast, we utilize the forward attention mechanism from [27], which contains stability and rapid convergence. The forward attention mechanism takes advantage of monotonic alignment from text sequences to acoustic sequences. Intuitively, in each decoding step, the decoder should concern either the linguistic embedding it concerned in the previous step, or the next linguistic embedding behind the embedding it concerned in the previous step. Therefore the optimization of the attention module has a much smaller search space and the training efficiency can be improved. We also incorporate the guided attention loss proposed in [28], which uses the same principle as forward attention. This loss will strictly penalize the non-monotonic weights derived from attention during the training phase to forcibly diagonalize the alignment output by attention. In our experiments, these methods are indeed effective to improve the training speed.

2.4.3. Decoder

The decoder module is an autoregressive model, which receives the aligned content through the attention mechanism and the output frame of the previous step, then outputs a frame of mel-spectrogram at each time step. During this phase, the hidden state of each step will be sent to a fully connected layer, which is followed by the sigmoid activation and finally outputs the probability of terminating the production. After getting the mel-spectrogram sequence of multiple frames, the Tacotron 2 architecture also includes a post-process network to adjust the entire sequence from a global perspective. 512 filters with batch normalization, which has 5x1 kernel, are employed to compose the presented model, followed by the tanh activations except the last layer. When it comes to the training phase, both the mel-spectrograms generated by the autoregressive network and the post-process network will be involved to minimize the mean squared error (MSE) with target mel-spectrograms.

2.5. WaveRNN Vocoder

We use WaveRNN to restore the waveform audio from the mel-spectrograms. It has impressive sampling speed though it generates speech with less naturalness compared with WaveNet, which is recognized as the best vocoder in the speech expressiveness category. The WaveRNN is designed as a single-layer recurrent neural network with sparse parameters to deduce the model's complexity. It uses two 8-bit output spaces to generate the high and low parts of the final 16-bit output, in this way the scale of the model can be further compressed. The compact and brief framework significantly decreases the computation of WaveRNN. However, in order to stabilize the inference and upgrade the sound quality as much as possible, we did not use the subscaling method in the original paper, but split the input sequences into segments and then generate speech for each segment in parallel.

The material for training our vocoder comes from the Blizzard Challenge 2020 dataset because our system is to generate the corresponding speaker's voice. It is worth mentioning that the acoustic models based on encoder-decoder architecture and the vocoder are independent of each other, so the two modules can be trained separately.

3. Experiment

3.1. Two-stage training for bilingual speech synthesis

As mentioned above, in order to fully train a bilingual text-to-speech system, we employ three datasets that come from different speakers, including the Blizzard Challenge 2020 Mandarin dataset, the DataBaker Mandarin dataset, and the LJSpeech English dataset. Since our target productions should be voices from the speaker of BC 2020 datasets, we divided the model training into two stages, using different datasets as training materials.

Figure 3 illustrates our training process. In the first stage, the text and mel-spectrogram pairs from the DataBaker[20] and LJSpeech dataset are sampled to tune the encoder-decoder model's parameters. Because the Mandarin and English text are unified in the font-end processing, the model can learn how to pronounce Mandarin Pinyin and English phonemes at the same time. In this step, the entire model must be fully trained until it can generate fluent and natural Mandarin and English speech because we need a fully functional encoder before the second stage of training.

Since the first stage training releases a fully-trained encoder module that can extract linguistic features from mixed Chinese and English texts, we fix the parameters of the encoder module and only employ the Blizzard Challenge 2020 dataset to fine-tune the other parts of the model. The main purpose of this stage is to adjust the decoder module, which finally generates the acoustic features, to make it generate the voice of the target speaker whether it is to synthesize Mandarin or English. At the same time, it also allows the model to learn the coherent pronunciation between Chinese and English words in each sentence.

3.2. Scheduled sampling training

In the speech synthesis based on the autoregressive model, the output of the previous step is usually the input in the next step. Therefore, for fast convergence, it is practical for the decoder to acquire the real elements in the target sequence as its input instead of the output from the previous step, and this method is namely teacher forcing training. However, attributable to the different distribution of training data and testing data, the teacher forcing training will weaken the model's robustness while accelerating the training, so we apply scheduled sampling training, which is widely employed in sequence-to-sequence tasks.

The scheduled sampling method maintains a probability indicating whether to use the ground truth element as input in each step of decoding, and the probability can be reduced over time in some way, such as linear decay and exponential decay. In that way, the convergence will speed up in the early stage of training because a large number of ground truth data can be received as input. As the training progresses, the decoder will gradually consume the generated elements to ensure its robustness. In our setting, we let the probability decay linearly from 1 to 0.

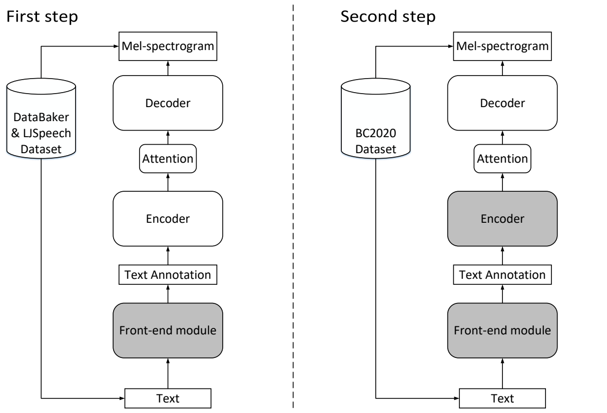

Figure 3: *Two-stage training process. The parameter-fixed modules are grayed out.*

3.3. Synthesis

Apparently, an autoregressive speech synthesis model is powerless to process excessively long texts because of its long-term dependency issue. Meanwhile, a really short input text results in lowly expressive speech because the system is incapable of expressing plentiful prosodic information due to lack of context. To alleviate these problems, we segment the texts according to symbols such as commas and periods before sending the texts to our text-to-speech system, and we also ensure that the minimum segment has at least 6 characters. Finally, after the speech of each segment is synthesized separately, we will concatenate them together as the output of the entire paragraph.

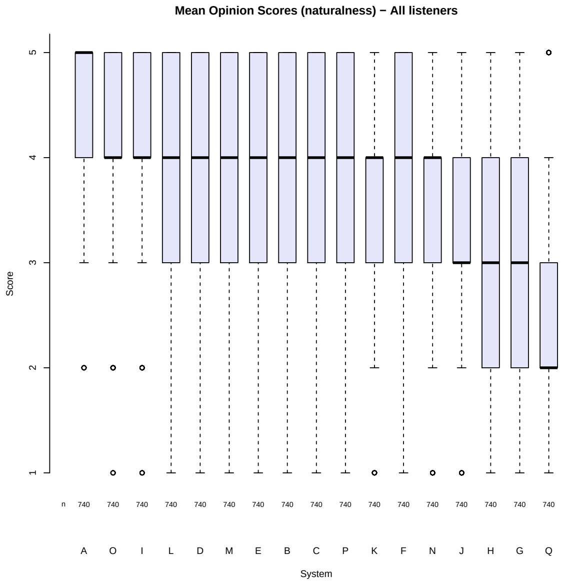

Figure 4: *Mean Opinion Scores (naturalness)*

4. Results

4.1. Subjective evaluation

The subjective evaluation assesses the system from three aspects: naturalness, similarity, and Pinyin error rate with tones (PTER). In the evaluation of naturalness and similarity, the Mean opinion score (MOS) ranges on a scale from 1 (bad) to 5 (excellent) to express the quality of generated speech. As for the PTER, it reports the intelligibility of the synthetic speech of the Intelligibility of sentences (INT) section which generates speech for random combinations of Chinese phrases.

The results composed of MOS (naturalness), MOS (similarity to original speaker), and PTER are presented in Figure 4, Figure 5, and Figure 6 respectively. Our system is denoted by letter Q in the evaluation and obviously, it performs poorly in all aspects. In terms of the naturalness of generated speech, we only got an average MOS of 2.6, which is far lower than the performance of other teams. Also, our PTER is 17.1%, which cannot meet the requirements of a high intelligibility speech synthesis system. As for the voice similarity to the original speaker, it is also weakened because of the unclear output speech, which is caused by the imperfect vocoder.

4.2. Discussion

After system analysis, we attribute the system's fault to our two-stage training method and the vocoder module.

In the second stage of training, the parameters of the encoder are constant and we hope that the encoder is stable enough to extract certain features for bilingual text. Nevertheless, the coherence between characters and words of our target speech, which comes from the Blizzard Challenge 2020 corpus, is different from the two datasets we apply in the first-stage train-

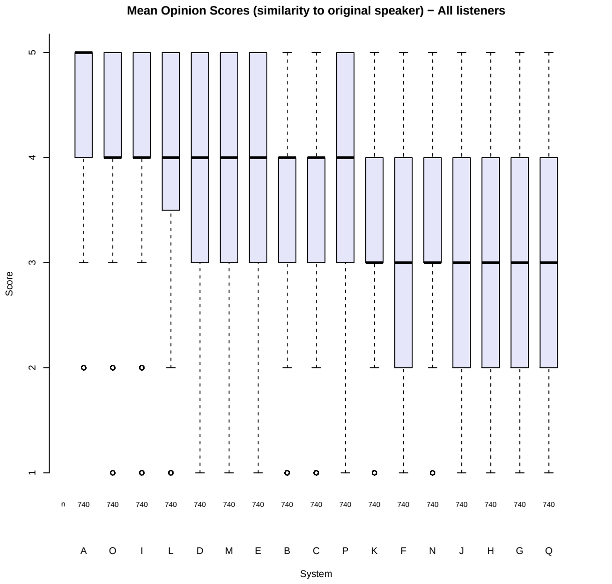

Figure 5: *Mean Opinion Scores (similarity to original speaker)*

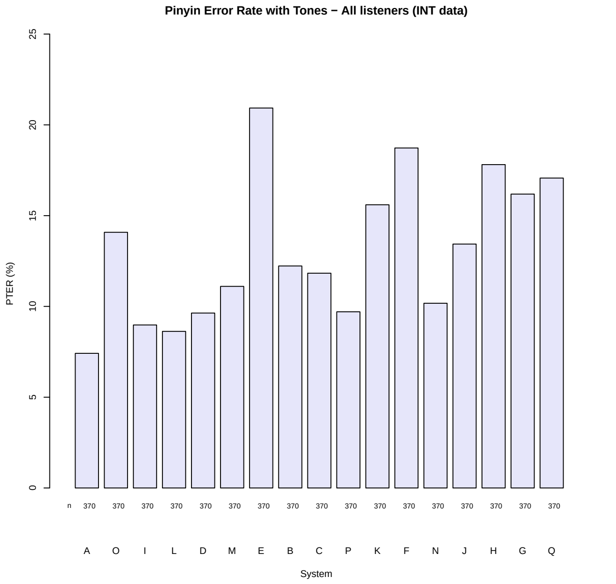

Figure 6: *Pinyin Error Rate with Tones*

ing, so the encoder should also be fine-tuned in the second stage. Furthermore, the parameter-fixed encoder also harms the effect of the attention mechanism, which results in confusing pauses in speech. What's worse, the Blizzard Challenge 2020 dataset does not contain all the Mandarin and English phonemes we need, so it may not completely convert the voice of all phonemes to the target speaker. As for the vocoder, WaveRNN uses a lightweight structure to reduce the amount of calculation, thereby increasing the sampling speed, but at the same time sacrificing the fidelity of the restored waveform. Finally, since we generate speeches for each subsentence divided from the input text separately and concatenate them as the whole sentence's speech, our approach also suffers from the same coherence issue between speech segments as the unit selection system.

5. Conclusions

In this paper, we have presented the SCUT bilingual speech synthesis system based on the Tacotron 2's end-to-end framework. For code-switched synthesis, we adjust the preprocessing module and introduce a two-stage training method. Also, our vocoder also achieves efficient sampling without compromising their quality as much as possible. Although our performance is ineffective and the synthesized voice is not smooth enough naturally enough as illustrated in the evaluation results, the challenge is an impressive experience for us who participated for the first time. In future work, we will investigate building a high-fidelity speech synthesis system with various prosodic styles based on end-to-end architecture.

6. References

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

[2] Z. Ling and R. Wang, "Hmm-based hierarchical unit selection combining kullback-leibler divergence with likelihood criterion," in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*, vol. 4, 2007, pp. IV–1245–IV–1248.

[3] A. Chalamandaris, S. Karabetsos, P. Tsiakoulis, and S. Raptis, "A unit selection text-to-speech synthesis system optimized for use with screen readers," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 3, pp. 1890–1897, 2010.

[4] K. Tokuda, Y. Nankaku, T. Toda, H. Zen, J. Yamagishi, and K. Oura, "Speech synthesis based on hidden Markov models," *Proceedings of the IEEE*, vol. 101, no. 5, pp. 1234–1252, 2013.

[5] H. Ze, A. Senior, and M. Schuster, "Statistical parametric speech synthesis using deep neural networks," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pp. 7962–7966, 2013.

[6] Y. Fan, Y. Qian, F. Xie, and F. K. Soong, "TTS synthesis with bidirectional LSTM based Recurrent Neural Networks," *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, no. September, pp. 1964–1968, 2014.

[7] Y. Wang, R. J. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, and Q. Le, "Tacotron: Towards end-To-end speech synthesis," *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, vol. 2017-Augus, pp. 4006–4010, 2017.

[8] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan, R. A. Saurous, Y. Agiomvrgiannakis, and Y. Wu, "Natural TTS Synthesis by Conditioning Wavenet on MEL Spectrogram Predictions," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, vol. 2018-April, pp. 4779–4783, 2018.

[9] S. O. Arik, G. Diamos, A. Gibiansky, J. Miller, K. Peng, W. Ping, J. Raiman, and Y. Zhou, "Deep voice 2: Multi-speaker neural text-to-speech," *Advances in Neural Information Processing Systems*, vol. 2017-December, no. Nips, pp. 2963–2971, 2017.

[10] W. Ping, K. Peng, and J. Chen, "Clarinet: Parallel wave generation in end-to-end text-to-speech," *7th International Conference on Learning Representations, ICLR 2019*, 2019.

[11] Y. Wang, D. Stanton, Y. Zhang, R. J. Skerry-Ryan, E. Battenberg, J. Shor, Y. Xiao, F. Ren, Y. Jia, and R. A. Saurous, "Style

tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis," *35th International Conference on Machine Learning, ICML 2018*, vol. 12, pp. 8229–8238, 2018.

[12] R. J. Skerry-Ryan, E. Battenberg, Y. Xiao, Y. Wang, D. Stanton, J. Shor, R. J. Weiss, R. Clark, and R. A. Saurous, "Towards end-to-end prosody transfer for expressive speech synthesis with tacotron," *35th International Conference on Machine Learning, ICML 2018*, vol. 11, pp. 7471–7480, 2018.

[13] D. Stanton, Y. Wang, and R. J. Skerry-Ryan, "Predicting Expressive Speaking Style from Text in End-To-End Speech Synthesis," *2018 IEEE Spoken Language Technology Workshop, SLT 2018 - Proceedings*, pp. 595–602, 2019.

[14] T.-Y. Hu, A. Shrivastava, O. Tuzel, and C. Dhir, "Unsupervised Style and Content Separation by Minimizing Mutual Information for Speech Synthesis," 2020. [Online]. Available: http://arxiv.org/abs/2003.06227

[15] A. van den 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," pp. 1–15, 2016. [Online]. Available: http://arxiv.org/abs/1609.03499

[16] T. L. Paine, P. Khorrami, S. Chang, Y. Zhang, P. Ramachandran, M. A. Hasegawa-Johnson, and T. S. Huang, "Fast Wavenet Generation Algorithm," pp. 1–6, 2016. [Online]. Available: http://arxiv.org/abs/1611.09482

[17] A. Van Den Oord, Y. Li, I. Babuschkin, K. Simonyan, O. Vinyals, K. Kavukcuoglu, G. Van Den Driessche, E. Lockhart, L. C. Cobo, F. Stimberg, N. Casagrande, D. Grewe, S. Noury, S. Dieleman, E. Elsen, N. Kalchbrenner, H. Zen, A. Graves, H. King, T. Walters, D. Belov, and D. Hassabis, "Parallel WaveNet: Fast high-fidelity speech synthesis," *35th International Conference on Machine Learning, ICML 2018*, vol. 9, pp. 6270–6278, 2018.

[18] N. Kalchbrenner, E. Elsen, K. Simonyan, S. Noury, N. Casagrande, E. Lockhart, F. Stimber, A. Van Den Oord, S. Dieleman, and K. Kavukcuoglu, "Efficient neural audio synthesis," *35th International Conference on Machine Learning, ICML 2018*, vol. 6, pp. 3775–3784, 2018.

[19] R. Prenger, R. Valle, and B. Catanzaro, "Waveglow: A Flow-based Generative Network for Speech Synthesis," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, vol. 2019-May, pp. 3617–3621, 2019.

[20] D. Baker, "Chinese standard mandarin speech copus," [Online] Available:https://www.data-baker.com/open_source.html, 2017.

[21] K. Ito, "The lj speech dataset," https://keithito.com/LJ-Speech-Dataset/, 2017.

[22] B. McFee, M. McVicar, S. Balke, C. Thomé, C. Raffel, O. Nieto, E. Battenberg, D. Ellis, R. Yamamoto, J. Moore, R. M. Bittner, K. Choi, F.-R. Stöter, S. Kumar, S. Waloschek, Seth, R. Naktinis, D. Repetto, C. Hawthorne, C. Carr, hojinlee, W. Pimenta, P. Viktorin, P. Brossier, J. F. Santos, JackieWu, Erik, and A. Holovaty, "librosa/librosa: 0.6.0," 2018.

[23] A. Robinson, "Beep pronunciation dictionary," *Retrieved from World Wide Web: ftp://svr-ftp. eng. cam. ac. uk/pub/comp. speech/dictionaries/beep. tar. gz*, 1996.

[24] 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.

[25] Y. Zheng, J. Tao, Z. Wen, and Y. Li, "BLSTM-CRF based end-to-end prosodic boundary prediction with context sensitive embeddings in a text-to-speech front-end," *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, vol. 2018-Septe, no. September, pp. 47–51, 2018.

[26] I. Turc, M.-W. Chang, K. Lee, and K. Toutanova, "Well-read students learn better: On the importance of pre-training compact models," *arXiv preprint arXiv:1908.08962v2*, 2019.

[27] J. X. Zhang, Z. H. Ling, and L. R. Dai, "Forward Attention in Sequence- To-Sequence Acoustic Modeling for Speech Synthesis," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, vol. 2018-April, pp. 4789–4793, 2018.

[28] H. Tachibana, K. Uenoyama, and S. Aihara, "Efficiently Trainable Text-to-Speech System Based on Deep Convolutional Networks with Guided Attention," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, vol. 2018-April, pp. 4784–4788, 2018.