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

# The Ajmide Text-To-Speech System for Blizzard Challenge 2020

*Beibei Hu, Zilong Bai, Qiang Li*

Ajmide Media, Shanghai, P.R. China

hubeibei@ajmide.com

## Abstract

This paper presents the Ajmide team's text-to-speech system for the task MH1 of Blizzard Challenge 2020. The task is to build a voice from about 9.5 hours of speech from a male native speaker of Mandarin. We built a speech synthesis system in an end-to-end style. The system consists of a BERT-based text front end that process both Chinese and English texts, a multi-speaker Tacotron2 model that converts the phoneme and linguistic feature sequence into mel spectrogram, and a modified WaveRNN vocoder that generate the audio waveform from the mel spectrogram. The listening evaluation results show that our system, identified by P, performs well in terms of naturalness, intelligibility and the aspects of intonation, emotion and listening effort.

**Index Terms**: text-to-speech, Blizzard Challenge 2020, end-to-end, BERT, Tacotron2, WaveRNN

## 1. Introduction

In order to better understand different speech synthesis techniques on a common dataset, Blizzard Challenge was held annually since 2005 [1]. This year's Blizzard Challenge has two tasks, hub task 2020-MH1 and spoke task 2020-SS1, which provide 9.5 hours of speech data from a male native speaker of Mandarin and 3 hours of speech data from a female native speaker of Shanghainese, respectively. The participants were asked to build text-to-speech system based on the provided data and the valid external data. The systems were evaluated by the synthetic speech, from the aspects of naturalness, similarity, intelligibility, and paragraph performance.

Until now, there are mainly three types of popular synthetic techniques: concatenation synthesis [2,3], statistical parametric synthesis [4-6] and deep learning-based synthesis [7-12]. Each approach has its own advantages and limitations. Due to the significant improvement on audio quality and the simplified training pipeline, neural networks based end-to-end TTS models have drawn much attention recently.

For the task 2020-MH1, we have developed an end-to-end speech synthesis system based on the deep neural networks. Our system has three main components: BERT (Bidirectional Encoders Representations from Transformers) [13]-based front-end, multi-speaker Tacotron2 acoustic model and modified WaveRNN vocoder. To benefit from the external speech data of other speakers we employ a multi-speaker Tacotron2 model, which include a global speaker embedding. For Chinese, G2P, word boundary and prosodic boundary are important for synthesized speech. Recently, BERT has showed great success in many natural language processing tasks, inspiring us to build a BERT-based front end in our system.

This rest of the paper is organized as follows. Section2 introduces the MH1 task of Blizzard Challenge 2020. Section 3 describes our TTS system architecture in detail. Section 4 presents the evaluation results. Finally, the conclusions are drawn in Section 5.

## 2. Data and task

The task 2020-MH1 of this year's Blizzard Challenge is as follows:

- Hub task 2020-MH1: Mandarin Chinese - About 9.5 hours of speech data from a male native speaker of Mandarin. The task is to build a voice from this data.

The dataset contains 4365 utterances and corresponding text transcriptions. The audio format is one channel, 48kHz sampling rate, and 16 bit wav format.

## 3. Ajmide TTS System

### 3.1. Overall architecture

The overview of our system is illustrated in Figure 1 with 3 parts: a BERT-based front-end, Tacotron2 acoustic model and WaveRNN vocoder.

In the training stage, a BERT-based model was trained for polyphone disambiguation and prosody prediction. A multi-speaker Tacotron2 model was trained as the acoustic model. And a modified WaveRNN was trained as the vocoder.

In the inference stage, the test sentences were first analyzed into phoneme sequence and linguistic feature sequence by front-end. Then the phoneme and feature sequences were convert to mel spectrograms via the acoustic model. Finally, the mel spectrograms were vocoded into waveform by the WaveRNN vocoder.

### 3.2. Data

#### *3.2.1. Data preprocessing*

First, we checked the text and corresponding audio. Some inconsistencies between the text and audio were found, so manual annotations were performed including text transcription and prosodic boundary. Finally, the audio files were down sampled to 22 kHz.

#### *3.2.2. External data*

The following external data was employed to train the models in our system:

- an internal TTS dataset, about 71.5 hours of speech data from several male native speakers of Mandarin
- an internal Chinese polyphone dataset
- the text and prosody label of data-baker's open source TTS dataset [14]
- the pre-trained RoBERTa model for Chinese [15]

10.21437/VCC_BC.2020-13

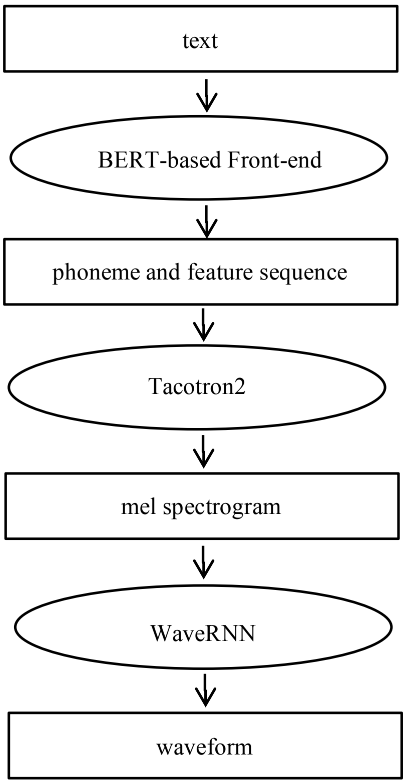

Figure 1: *The architecture of Ajmide TTS system*.

### 3.3. Front-end

In our system, the front-end is a pipeline-based system, which consist of text normalization (TN), Chinese word segmentation (CWS), part-of-speech (POS) tagging, grapheme-to-phoneme (G2P) conversion, and prosody prediction. In addition, the CMU Pronouncing Dictionary [16] is adopted for English words processing. The front-end analyzes the text, converts the sentence to phoneme sequence and outputs the phoneme level linguistic features.

A rule based TN is employed to process the special symbols. For the procedure of CWS and POS, jieba [17] is used.

We trained a BERT-based model for the polyphone disambiguation with the pre-trained Chinese RoBERTa model. An internal polyphone dataset, which includes 145 Chinese polyphonic characters, was used.

For the procedure of prosody prediction, manual annotation was performed on prosodic boundary for MH1 dataset. And the prosody labels from data-baker dataset were also used. We first trained the model by the data-baker prosody labeled sentences, then fine-tuned the model by the prosody labeled sentences of MH1. The prosody prediction model is also a BERT-based model.

The model size of the BERT backbone is 12 transformer blocks with hidden size of 768 and self-attention heads of 12. A linear classifier is adopted after the BERT-base. The loss function is cross-entropy loss. The polyphone disambiguation model and prosody prediction model were trained separately with the batch size of 32.

Considering the text of MH1 includes English words, we use the International Phonetic Alphabet (IPA) [18] phoneme set. Both pinyin and English words were converted to the IPA phoneme sequence.

For an input sentence, the output sequence of the front end contains the phoneme, word segment labels, POS labels and prosodic symbols.

### 3.4. Acoustic model

The sequence-to-sequence architecture for generating the acoustic features simplifies the traditional speech synthesis pipeline, and the synthetic speech can achieve a high MOS. Therefore, we use Tacotron2, an end-to-end style model, as our acoustic model. The acoustic feature is an 80-dim mel-frequency spectrogram with 50 ms frame size, 12.5 ms frame hop, and a Hann window. The reduction factor of the model is set to 1.

Although Tacotron2 can produce satisfactory speech for English TTS system, we found that the it's not good enough for Chinese, especially in speech pauses and prosody stability. To improve the performance in Chinese TTS, we employ the phoneme and the additional linguistic features as the input sequence of Tacotron2.

To benefit from the external audio data of other speaker, a speaker embedding module is added to Tacotron2. The speaker embedding module consists an embedding layer and an expand module. The speaker embedding dimension is 128. The speaker id is converted to a high dimension vector and expanded to the length of the encoder output sequence. The concatenation of encoder output and speaker embedding output is fed to the decoder.

During the training process, we adopted a 71.5-hour male data to train the multi-speaker Tacotron2 model and fine-tuned the model with the MH1 data. First, a multi-speaker Tacotron2 model was trained on both 71.5-hour male dataset and the MH1 dataset with a batch size of 32, using Adam optimizer with a fixed learning rate of 1e-4. Then the model was fine-tuned with the MH1 dataset only.

In our experiments it was hard to achieve an alignment with the only MH1 dataset. However, in the fine-tuning procedure the Tacotron2 model quickly got an alignment on the same dataset. Figure 2 shows the attention alignments of a validation sample.

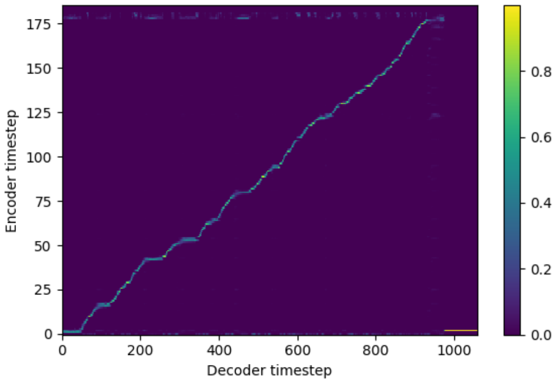

Figure 2: *Attention alignments.*

### 3.5. Vocoder

In our system, we choose a modified WaveRNN [19] as the vocoder. The audio was applied a 10-bits μ-law quantification.

In the training phase, first, we trained the model on all data, including an internal 71.5-hour dataset and the MH1 dataset, using the ground truth features. The cross-entropy loss was employed. The model was trained with a batch size of 128, using Adam optimizer with a constant learning rate of 1e-4.

Then a GTA fine-tuning was performed on the MH1 dataset. The Tacotron2 predicted mel spectrograms were used to fine-tune the WaveRNN. Experiments show that the GTA fine-tuning can significantly reduce the noise and improve the audio quality.

## 4. Evaluation

In this year's challenge, there are 17 systems in total, including 16 participating teams and one natural speech. System A is natural speech and system P is ours.

Table 1: *Evaluation sections for Task 2020-MH1.*

| Sections | Detailed Description |
|---|---|
| section 1 | similarity - news sentences |
| section 2 | similarity - PSC sentences |
| section 3 | naturalness – news sentences |
| section 4 | naturalness – PSC sentences |
| section 5 | various criteria - news paragraphs |
| section 6 | intelligibility |

The evaluation comprised six sections shown in Table 1. The results are based on all the listeners' responses, including paid listeners, experts and volunteers. Finally, our system has achieved good results in some criteria for the Challenge. Details are as follows.

### 4.1. Naturalness test

In naturalness test, listeners listened to one sample and chose a score which represented how natural or unnatural the sentence sounded on a scale of 1 [Completely Unnatural] to 5 [Completely Natural].

Figure 3 shows the boxplot of evaluation results of all systems on naturalness. Our system has an average score of 3.9 with 1.04 standard deviation. Besides, Wilcoxon signed rank tests show that there is no significant difference between the systems of B, C, D, E, F, K, L, M and P in naturalness test. Among the 16 systems participating in the challenge, our system is outperformed by two systems (I and O).

### 4.2. Similarity test

The similarity score represents how similar the synthetic voice sounded to the voice in the reference samples on a scale of 1 [Sounds like a totally different person] to 5 [Sounds like exactly the same person].

The boxplot of similarity evaluation results is presented in Figure 4. The similarity score of our system for all listeners is 3.7 with 1.15 standard deviation.

### 4.3. Intelligibility test

In this test, the listeners were allowed to listen to each sentence at most twice then typed in what they heard.

Pinyin error rates with tones (PTER) of all participant systems are presented in Figure 5. When evaluated by all listeners, the PTER of our system is 0.097 with 0.15 standard deviation. The results indicate that our system performs well on intelligibility. Besides, Wilcoxon signed rank tests shows that there is no significant difference between our system and the natural speech system.

### 4.4. Paragraph performance

In paragraph test, the listeners listened to one whole paragraph from news domain. Seven aspects of speech, including overall impression, pleasantness, speech pauses, stress, intonation, emotion, and listening efforts are evaluated separately. The score is on a scale of 1 to 60 for each aspect.

Overall results are shown in Figure 6. The mean opinion scores of our system are listed in Table 2. In aspects of intonation, emotion and listening effort, our system achieves good performance.

Table 2: *Paragraph listening test scores of our system.*

| Criterion | MOS |
|---|---|
| overall impression | 40 |
| pleasantness | 39 |
| speech pauses | 40 |
| stress | 40 |
| intonation | 41 |
| emotion | 41 |
| listening effort | 42 |

### 4.5. Discussion

Our system achieved a good score of PTER. We believe that it was benefit from the accurate front end processing, especially the polyphone disambiguation, tone sandhi and Erhua processing. The BERT-based model obtained a satisfying performance on Mandarin G2P.

We have reviewed our synthetic audio samples and found 2 types of defects that may lead to the performance degradation. The prosody and stress error may decrease the similarity and some aspects of paragraph performance, such as pleasantness and overall impression. For the paragraph audio samples, we synthesized the utterances separately and concatenated into the whole paragraph audio. The difference between two adjacent audio utterances would introduce a negative effect on paragraph scores. In the future, we will study the method for long-form speech synthesis.

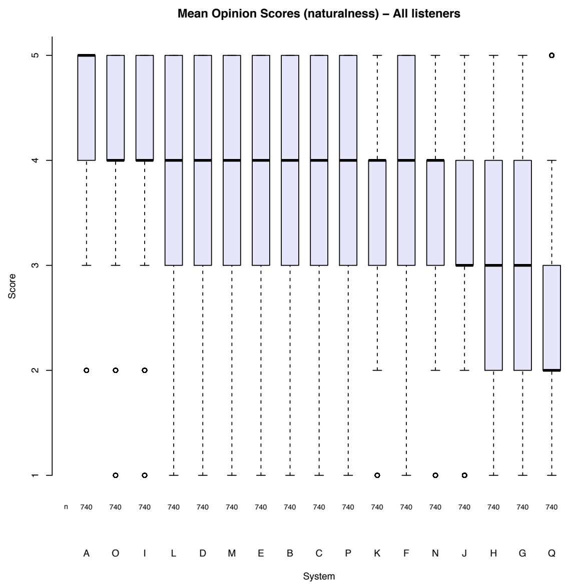

Figure 3: *naturalness scores for all listeners.*

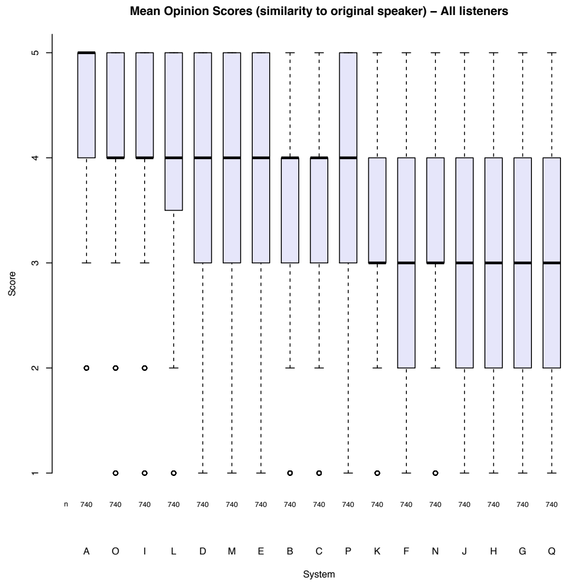

Figure 4: *similarity scores for all listeners.*

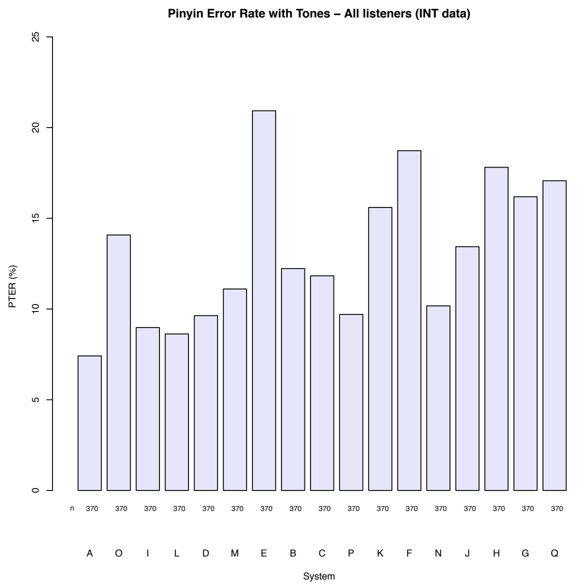

Figure 5: *Pinyin error rate with tones scores for all listeners.*

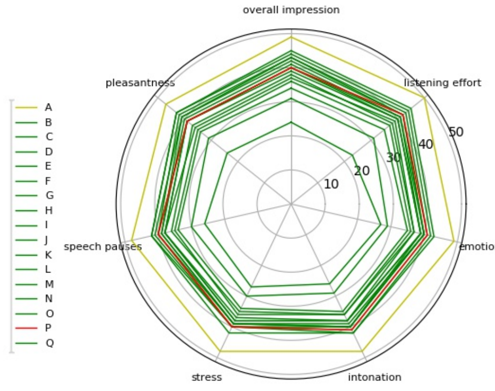

Figure 6: *paragraph scores for all listeners. Natural speech system is shown in yellow, our system in red and other participants are in green.*

## 5. Conclusions and future work

This paper presents the details of our submitted speech synthesis system and the evaluation results in Blizzard Challenge 2020. We built an end-to-end style acoustic model following with a WaveRNN vocoder. Our system achieved good performance on some criterion for the Challenge such as naturalness, intelligibility, and emotion. But the performance on similarity is not satisfiable.

In future work, we will make more attempts in the expressive speech synthesis. At the same time, we will study the training techniques to improve the performance on the out-of-domain sentences.

# 6. References

[1] A. W. Black and K. Tokuda, "The blizzard challenge-2005: Evaluating corpus-based speech synthesis on common datasets," in The 20*th Annual Conference of the International Speech Communication, September 4-8, Lisbon, Portugal, Interspeech 2005,* 2005, pp. 77-80.

[2] Z. H. Ling and R. H. Wang, "HMM-based hierarchical unit selection combining kullback-leibler divergence with likelihood criterion," in The *22nd IEEE International Conference on Acoustics, April 15-20, Honolulu, Hawaii, USA, Speech and Signal Processing,* 2007, pp. 1245-1248.

[3] Z. Yan, Y. Qian, and F. K. Soong, "Rich-context unit selection (RUS) approach to high quality TTS," in The *35th IEEE International Conference on Acoustics, March 14-19, Dallas, Texas, USA, Speech and Signal Processing,* 2010, pp. 4798-4801.

[4] H. Zen, K. Tokuda, and A. W. Black, "Statistical parametric speech synthesis," *Speech Communication*, vol. 51, no. 11, pp. 1039-1064, 2009.

[5] Y. Fan, Y. Qian, F. Xie, and F. K. Soong, "TTS synthesis with bidirectional LSTM based recurrent neural networks," in The *15th Annual Conference of the International Speech Communication, September 14-18, Association, Singapore, Interspeech 2014,* 2014, pp. 1964-1968.

[6] H. Zen and H. Sak, "Unidirectional long short-term memory recurrent neural network with recurrent output layer for low-latency speech synthesis," in The *40th IEEE International Conference on Acoustics, April 19-24, South Brisbane, Queensland, Australia, Speech and Signal Processing,* 2015, pp. 4470-4474.

[7] Y. Wang, R. J. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. V. Le, Y. Agiomyrgiannakis, R. Clark, and R. A. Saurous, "Tacotron: Towards end-to-end speech synthesis," in The *18th Annual Conference of the International Speech Communication Association, August 20-24, Stockholm, Sweden, Interspeech* 2017, 2017, pp. 4006-4010.

[8] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. J. Skerry-Ryan, R. A. Saurous, Y. Agiomyrgiannakis, and Y. Wu, "Natural TTS synthesis by conditioning wavenet on mel spectrogram predictions," in The *41st IEEE International Conference on Acoustics, April 15-20, Calgary, Alberta, Canada, Speech and Signal Processing,* 2018, pp. 4779-4783.

[9] N. Li, S. Liu, Y. Liu, S. Zhao, M. Liu, and M. Zhou, "Close to human quality TTS with transformer," in The 33rd AAAI Conference on Artificial Intelligence, *February 2-7, New Orleans, Louisiana, USA, Processing,* 2018, pp. 6706-6713.

[10] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," in The *9th International Symposium on Computer Architecture, September 13-15, Sunnyvale, California, USA, Speech Synthesis Workshop*, 2016, pp. 125.

[11] J. Valin and J. Skoglund, "LPCNET: improving neural speech synthesis through linear prediction," in The *42nd IEEE International Conference on Acoustics, May 12-17, Brighton, United Kingdom, Speech and Signal Processing*, 2019, pp. 5891-5895.

[12] 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 The 35th *International Conference on Machine Learning, July 10-15, Stockholm, Sweden, Proceeding,* 2018, pp. 2415-2424.

[13] D. J, C. M.W, and e. a. Lee K, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *North American Chapter of the Association for Computational Linguistics, July 29-31, Florence, Kentucky, USA, Human Language Technologies*, 2019, pp. 4171-4186.

[14] D. Baker, Chinese Standard Mandarin Speech Corpus, https://www.data-baker.com/open_source.html, 2017.

[15] L. Xu, RoBERTa for Chinese, Tensorflow & Pytorch, https://github.com/brightmart/roberta_zh, 2020.

[16] CMU Pronouncing Dictionary, http://www.speech.cs.cmu.edu/cgi-bin/cmudict, 2020.

[17] J. Sun, jieba, https://github.com/fxsjy/jieba, 2020.

[18] International Phonetic Alphabet, https://www.internationalphoneticalphabet.org, 2020.

[19] O. McCarthy, WaveRNN, https://github.com/fatchord/WaveRNN, 2020.