The RoyalFlush Synthesis System for Blizzard Challenge 2019

Ming Chen, Zeru Lu, Peng Zhang, Jian Lu, Xudong Zhao, Xinkang Xu, Xinhui Hu

Hithink RoyalFlush Information Network Co.,Ltd. HangZhou, P.R. China

chenming@myhexin.com

Abstract

The task of the Blizzard Challenge 2019 is to build a speech synthesizer based on an 8 hours of speech data from an internet talk show by a well-known Chinese character. We present the RoyalFlush synthesis system to address the above challenge in this paper. Based on the Google's Tacotron 2 architecture, we firstly trained a basic multi-speaker model using an external 30 hours of speech dataset from 22 speakers. Then we applied transfer learning to fine-tune the basic model with the 8 hours of speech data provided by the Challenge committee and obtained the final model. To capture speaker's personal characteristics, we added the speaker embedding in the encoder and generated speeches with speaker characteristics in the decoder. The output speech was generated by using Griffin-Lim algorithm in consideration of high speed response.

Among all the participating teams of the Challenge, the identifier for our system is H. Evaluation results demonstrated that our system achieved relatively good results in all aspects.
Index Terms: Blizzard Challenge 2019, speech synthesis, speaker embedding, transfer learning

1. Introduction

The purpose of the Blizzard Challenge, which has been held annually since 2005, is to better understand and compare research techniques in building corpus-based speech synthesizers on the same data. Meanwhile, it allows people to engage in speech synthesis work better, learn to each other and make progress together.

Speech synthesis is a technique that translates the normal text into speech, which simulates human as natural as possible. A computer system which can perform such tasks, in the form of software or hardware, is called a speech computer or speech synthesizer. Intelligibility and naturalness are two key points of a speech synthesis system. Concretely, intelligibility represents the degree that the synthesized speech can be understood, while naturalness describes how closely the synthesized speech sounds like real human speech. The ideal speech synthesizer should generate intelligible and natural speech. Existing speech synthesis systems, for example, WaveNet [1], Tacotron [2], and Deep Voice [3] intend to maximize these two characteristics.

Until now, there are mainly three types of popular synthetic techniques: concatenation synthesis, HMM-based synthesis and deep learning-based synthesis. Each approach has its own advantages and limitation.

- **Concatenation synthesis:** Concatenation synthesis is based on the concatenation (or stringing together) of recorded speech units. Ling et al. [4] presented HMM-based unit selection method to determine the selected speech units for generating speeches in Blizzard Challenge 2007. Concatenation TTS directly select natural speech units from a recorded speech database, which enable speech synthesis with natural quality. However, as the footprint of the stored data is reduced, desired units may be unavailable in the database, and audible discontinuities may result.
- **HMM-based synthesis:** HMM-based synthesis, which is also called statistical parametric synthesis, is a method to synthesize speech at the foundation of hidden Markov models (HMMs). In this method, the frequency spectrum (vocal tract), fundamental frequency (voice source) and duration (prosody) of speech are simultaneously formulated by using HMMs [5]. In this approach, speech is synthesized by using maximum likelihood criterion [6]. Post-filter processings [7, 8, 9] are generally used for optimization. Compared with concatenation synthesis, speech can be synthesized by this method using a relatively small corpus, but the quality is poorer than the concatenation approaches. However, because of its stability, the HMM-based method has been utilized in practical applications before the deep learning based approach appeared.
- **Deep learning-based synthesis:** With the development of deep learning technology, the approaches based on the deep neural network (DNN) have become the main techniques in speech synthesis field. With powerful nonlinear modeling ability, the DNN-based method can effectively improve modeling accuracy. It also provides a facilitating mechanism to tune some specified hidden layers using specified speaker's speech, so the personalized speech is easy to obtain. There are many TTS systems such as WaveNet, Tacotron and Deep Voice, showed that DNN-based approaches can achieve the quality of the human voice.

With the development and application of the speech technologies, the demand for personalized synthesized speech is getting higher and higher. Therefore, it has become a hot topic to achieve multi-speaker adaption and synthesize personalized speech in both industry and academia. However, it is impractical to build a model for each speaker. In this Challenge task, we realized our speech synthesis system by adapting a basic model which is trained by a speech corpus of multi-speaker, using the speech of the appointed speaker by the Challenge office.

The remainder of this paper is organized as follows. In Section 2, we introduce the task in Blizzard Challenge 2019. In Section 3, each module of our system is described. Section 4 gives the process of experiments in detail. The results of all the participating teams are given in Section 5. Finally, Section 6 presents some concluding remarks to end the paper.

2. The task in Blizzard Challenge 2019

The single task of the Blizzard Challenge 2019 is as follows:

- **Single task 2019-MH1: Mandarin Chinese Found Data** - About 8 hours of speech data from an internet

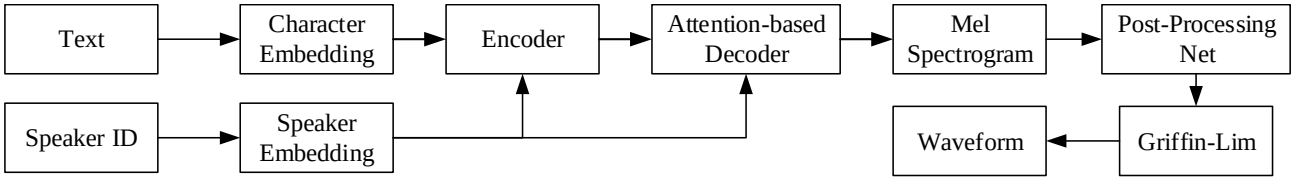


Figure 1: *The overall architecture of the RoyalFlush speech synthesis system.*

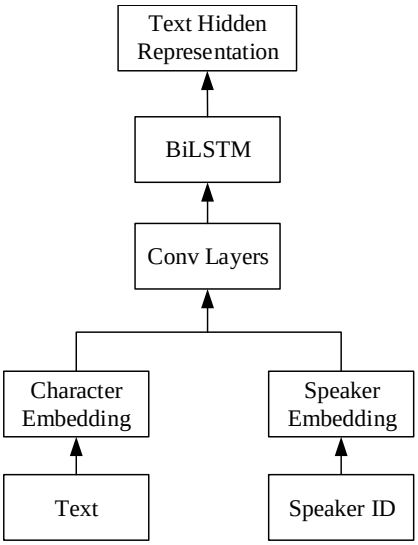


Figure 2: *The encoder architecture of our system.*

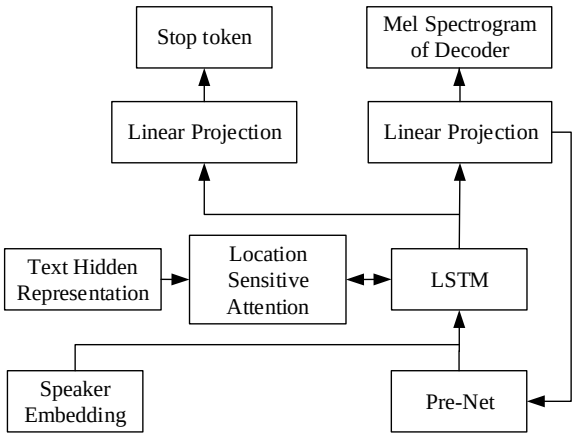


Figure 3: *The decoder architecture of our system.*

talk show by a well-known Chinese character will be released. The task is to build a voice from this data that is suitable for expressive TTS.

3. System description

The overall architecture of the RoyalFlush speech synthesis system is shown in Figure 1. It is basically based on Tacotron 2 architecture. There are three modules, encoder, attention-based decoder and post-processing net in this system. The backbone of the system is a sequence-to-sequence (seq2seq) model with attention.

As shown in the figure, the character embedding generated by text and the speaker embedding generated by speaker ID are simultaneously used as the encoder inputs to generate text hidden representation conditioned on speaker identity. In the attention-based decoder, the speaker embedding is also used as one of the input to control the mel spectrogram generation. Finally, after the post-processing net for the mel spectrogram, the speech waveforms are produced by a vocoder realized by the Griffin-Lim algorithm. The detailed descriptions of each module will be presented in the following parts.

3.1. Encoder

The encoder is used to covert text sequence into a fixed-dimensional vector of feature representations conditioned on speaker identity. The encoder architecture of our system is shown as Figure 2, it consists of 3 convolutional layers and 1 bi-directional LSTM [10].

Character embedding and speaker embedding are two parts of the encoder input.

Character embedding is used to extract the text information. Because the text contains both Chinese and English, we firstly convert Chinese words into Pinyin sequences and English words into phoneme sequences. Then we use these Pinyin and phoneme sequences to generate character embedding.

Speaker embedding is used to capture the characteristics of speakers. In our system, there is a speaker embedding table, which consists of embedding vectors of multiple speakers and is updated alongside the whole encoder-decoder architecture. The speaker embedding is obtained by searching the embedding table indexed by speaker identity. By mixing multiple speakers, the model internal representation can be shared among different speakers, so that the lack of data of one speaker can be compensated with data from other speakers, therefore, multi-speaker model can be more stable and robust.

In [11], text hidden representation is concatenated with the speaker embedding to form the encoder embedding that will be as one of the input of the decoder. Compared with this work, in the encoder of our system, the character embedding and speaker embedding together form the input of the encoder and finally generating the text hidden representation, in other words, the text hidden representation is conditioned on speaker identify. In this way, our system can efficiently capture the speaker characteristics.

3.2. Decoder

The decoder is used to predict mel spectrogram from the text hidden representation by using an attention-based neural network. The decoder architecture of our system is shown in Figure 3. Speaker embedding is used at each decoder step.

In the decoder, the attention context vector of text hidden representation is used as the input of LSTM, the output of LST-

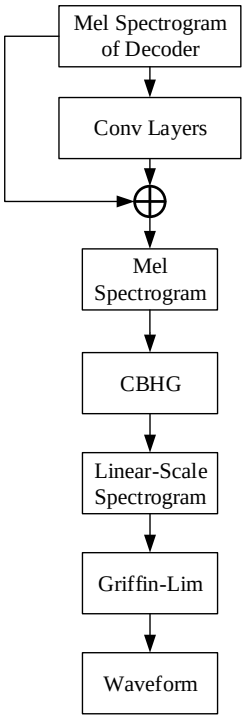


Figure 4: *The architecture of the post-processing net and waveform synthesizer.*

M is used to predict the mel spectrogram frame by using linear projection. In the whole process, the autoregressive process is the most important step that influence the quality of the final synthesized speech. Therefore, in order to make the synthesized speech conform to the characteristics of the speaker, we add the speaker embedding in the decoder. In other words, the pre-net output, the speaker embedding and the attention context vector are simultaneously used as the input of the LSTM. This changes in setting can help our system synthesize speech to adapt the speak characteristics efficiently. The components of the decoder are the same as the original Tacotron 2 except for the components mentioned above.

3.3. Post-processing net and vocoder

After the processing of the encoder and decoder, the mel spectrograms of the text are predicted. However, it is necessary to improve the overall reconstruction by using the post-processing net and synthesize the final speech waveforms. The architecture of the post-processing net and vocoder is shown as Figure 4.

The post-processing net of the system is a net with 5 convolutional layers. It is achieved by minimizing the summed mean squared (MSE) between the mel spectrogram of decoder and the mel spectrogram after the post-processing net.

In order to synthesize the final speech waveform from the mel spectrogram, the mel spectrogram is firstly processed by a CBHG module, which is proposed in Tacotron, to obtain linear-scale spectrogram, and then the final speech audio is synthesized by using Griffin-Lim algorithm [12]. We adopted this algorithm to generate waveform in consideration of high speed response.

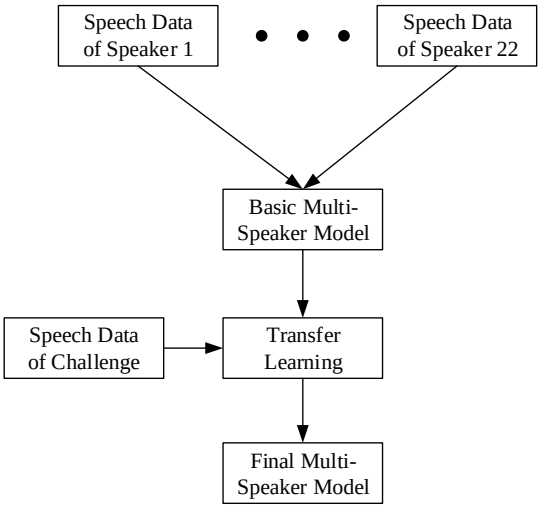


Figure 5: *The whole training phase.*

4. Experiments

4.1. Data processing

The task of the Blizzard Challenge 2019 is to convert the Chinese text into Mandarin Chinese speech audio. From the Challenge committee, an 8 hours of speech data from an internet talk show by a well-known Chinese character was provided. This dataset is defined as the official dataset in this paper. The dataset contains 480 paragraphs of speech data in the MP3 format together with their corresponding text. The sampling rate of these original speech data is 24 KHZ.

In the data processing stage, both the speech data and text are processed. The speech data are converted to WAV format and down-sampled to 16 KHZ. There are two operations, including normalization and segmentation, for processing text. In the normalization process, as discussed in Section 3.1, Chinese words and English words are converted into Pinyin and phoneme sequences, respectively. In the segmentation process, in order to overcome the alignment errors and slow convergence which are frequently happened in the case of long sentences in the end-to-end speech synthesis system, we split long sentences into relative short ones. Finally, the 480 paragraphs were split into 8699 sentences.

4.2. Training phase

In training phase, transfer learning strategy is utilized to improve the robustness. The whole training process is shown as Figure 5.

First, we trained the basic multi-speaker model on an external dataset containing about 30 hours of speech data by 22 speakers. It should be noted that these data are completely unrelated to the official dataset in terms of data contents and speakers. Compared with traditional speech synthesis systems, where dozens of hours of speech data of single speaker are used for training model, speech synthesis system trained by speech of multiple speakers can obtain large phoneme coverages and high robustness.

Then, transfer learning is applied. Concretely speaking, we used the official dataset to fine-tune the basic model's parameters and obtained the final model of the specific speaker.

4.3. Synthesis phase

There are 2546 sentences provided by the Challenge for evaluation. This evaluation set contains long and short sentences in which both Chinese and English are contained. In the synthesis phase, long sentences are firstly split into short ones in the same way as in the training phase. Then, Chinese text and English words are converted to Pinyin sequences and phoneme sequences, respectively. Next, the audio waveforms of these segmented text are synthesized using our system. Finally, these audio segments are spliced into a whole speech of the complete paragraph.

5. Results

In the Blizzard Challenge 2019, a total of 26 systems were submitted, among them, the natural speech and benchmark were also included. Here, system A is marked as the natural speech, system B is the benchmark system, and ours is system H.

The evaluation results, including naturalness test, similarity test and intelligibility test, of all participating systems are shown in Figure 6. The mean opinion score (MOS) is used to represent the naturalness of the system. The Pinyin (without tone) error rate (PER) and Pinyin (with tone) error rate (PTER) are used to indicate the intelligibility of the system.

Our system achieved relatively good results in all aspects, but there are still much space in improving performance. One of reason is that we use the Griffin-Lim to realize the vocoder, this decision is based on the tradeoff among stability, efficiency and speech quality. Through using the Griffin-Lim algorithm, our system is capable of synthesizing speech in real time. We believe that our system can get better results by using more complex synthesizer such as WaveNet.

6. Conclusions

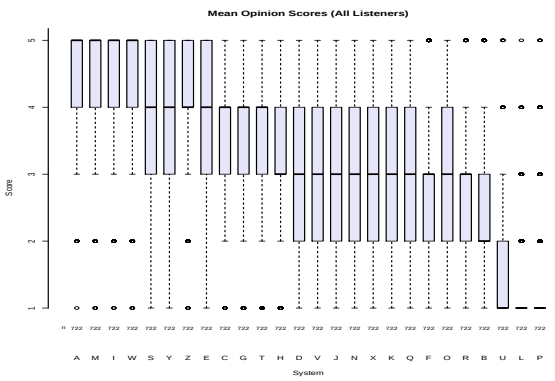
This paper gives the description of our submitted RoyalFlush speech synthesis system and the results in Blizzard Challenge 2019. Based on the Tacotron 2, the system added the speaker embedding in the encoder and decoder process for multi-speaker adaption. Our system achieved a good performance in all aspects of speech synthesis systems of the Challenge. However, there are still much space in improving performance. In future work, the fine-grained prosody control of multiple speakers with few samples is the main direction of our system.

7. References

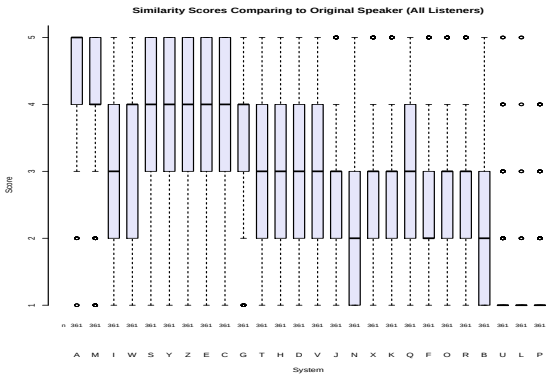
[1] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," in *The 9th ISCA Speech Synthesis Workshop, Sunnyvale, CA, USA, 13-15 September 2016*, 2016, p. 125.

[2] 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 *Interspeech 2017, 18th Annual Conference of the International Speech Communication Association, Stockholm, Sweden, August 20-24, 2017*, 2017, pp. 4006–4010.

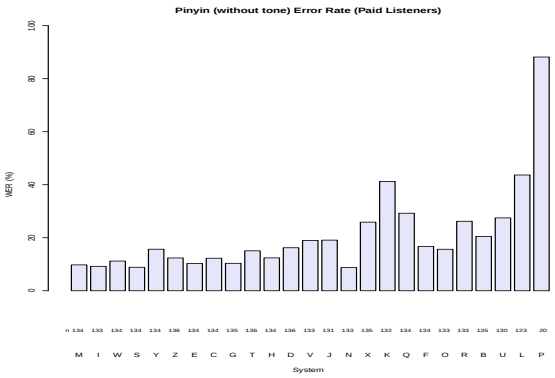
[3] S. Ö. Arik, M. Chrzanowski, A. Coates, G. F. Diamos, A. Gibiansky, Y. Kang, X. Li, J. Miller, A. Y. Ng, J. Raiman, S. Sengupta, and M. Shoeybi, "Deep voice: Real-time neural text-to-speech," in *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, 2017, pp. 195–204.



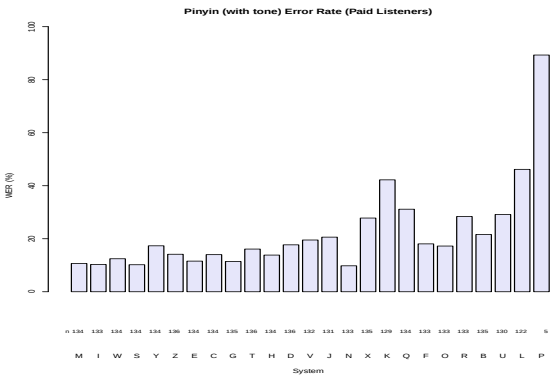
(a) Naturalness Test - MOS



(b) Similarity Test - Similarity



(c) Intelligibility Test - PER



(d) Intelligibility Test - PTER

Figure 6: *Evaluation results of the Blizzard Challenge 2019.*

[4] Z.-H. Ling, L. Qin, H. Lu, Y. Gao, L.-R. Dai, R.-H. Wang, Y. Jiang, Z.-W. Zhao, J.-H. Yang, J. Chen, and G.-P. Hu, "The ustc and iflytek speech synthesis systems for blizzard challenge 2007."

[5] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Simultaneous modeling of spectrum, pitch and duration in hmm-based speech synthesis," in *Sixth European Conference on Speech Communication and Technology, EUROSPEECH 1999, Budapest, Hungary, September 5-9, 1999*, 1999.

[6] K. Sangramsing and G. Bharti, "A text-to-speech synthesis for marathi language using festival and festvox," *International Journal of Computer Applications*, vol. 132, no. 5, pp. 30–41, 2015.

[7] T. Toda and K. Tokuda, "A speech parameter generation algorithm considering global variance for hmm-based speech synthesis," *IEICE Transactions*, vol. 90-D, no. 5, pp. 816–824, 2007.

[8] H. Silén, E. Helander, J. Nurminen, and M. Gabbouj, "Ways to implement global variance in statistical speech synthesis," in *INTERSPEECH 2012, 13th Annual Conference of the International Speech Communication Association, Portland, Oregon, USA, September 9-13, 2012*, 2012, pp. 1436–1439.

[9] T. Shinnosuke, T. Tomoki, N. Graham, S. Sakriani, and N. Satoshi, "A postfilter to modify the modulation spectrum in hmm-based speech synthesis," in *IEEE International Conference on Acoustics*, 2014.

[10] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

[11] Y. Jia, Y. Zhang, R. J. Weiss, Q. Wang, J. Shen, F. Ren, Z. Chen, P. Nguyen, R. Pang, I. Lopez-Moreno, and Y. Wu, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," in *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada.*, 2018, pp. 4485–4495.

[12] G. D. W. and L. Jae, "Signal estimation from modified short-time fourier transform," *IEEE Transactions on Acoustics Speech & Signal Processing*, vol. 32, no. 2, pp. 236–243, 1984.