The TJU-Didi-Huiyan system for Blizzard Challenge 2019

Ju Zhang[1], *Shaotong Guo*[1], *Cheng Gong*[1], *Shuaiting Chen*[1], *Yuguang Wang*[2], *Longbiao Wang*[1, 2], *Wei Zou*[3], *Xiangang Li*[3]

[1]Department of Intelligence and Computing, Tianjin University, Tianjin 300350, China
[2]Huiyan Technology (Tianjin) Co., Ltd
[3]Didi Chuxing

juzhang@tju.edu.cn, ygwang@huiyan-tech.com, longbiao_wang@tju.edu.cn

Abstract.

In this paper, we introduce an end-to-end text-to-speech system based on Tacotron 2 for Blizzard Challenge 2019. The main aim of our system is to synthesis voice as similar as possible to the voice provided by the real male speaker. In the front-end, we convert the Chinese character sequences to Pinyin sequences with tone and prosody annotation. In the back-end, the Tacotron 2 model is adapted for predicting spectrogram features. Then, the predicted spectrograms are used to generate 16-bit speech waveforms by Griffin-lim algorithm.

This is the first time for us to join the Blizzard Challenge, and the identifier for our system is X. Experimental results in subjective listening tests show that our system performed well on the naturalness test compared with merlin benchmark.

Index Terms: Blizzard Challenge 2019, Tacotron 2, end-to-end, speech synthesis, Griffin-lim algorithm

1. Introduction

The Blizzard Challenge is an open platform for the evaluation of speech synthesis technology. It has been held every year since 2005 and is dedicated to better understanding and comparing the research techniques of speech synthesis on the same corpus. There are four evaluation items for the task, namely: similarity, naturalness, error rate, and overall feeling of the paragraph; the overall feeling of the paragraph is divided into 6 sub-items: pleasure, pause rhythm, accent, tone, emotion, and hearing resistance.

Drawing on segmentation technology of traditional speech synthesis technology, in 2005, a waveform-cascading system based on unit selection was proposed to generate speech segments similar to natural speech [1, 2]. There is a large increase in the synthesis effect but a large corpus and expert corrections are required.

In 1999, the statistical parameter speech synthesis (SPSS) method based on HMM was firstly proposed and successfully applied [3]. In this method, the HMMs framework was modeled simultaneously by the spectrum, pitch, and duration. In summary, the Statistical Parameters Speech Synthesis (SPSS) method is intended to parameterize waveforms and establish acoustic models to predict their acoustic characteristics [3, 4, 5]. Flexibility is the biggest advantage of SPSS systems, but the quality of synthesized speech is restricted by vocoders.

In recent years, multilayered neural network models have been successfully applied to SPSS [6, 7, 8]. Recently, a neural network-based autoregressive model named WaveNet [9] can directly generate speech waveforms. It is best to choose the speech naturalness of the voice WaveNet compared to the HMM-based reference unit. After that, an end-to-end architecture called Tacotron [10, 11], attended by a modified trough model as a vocoder, gave a speech equivalent to the professional-level average score (MOS) record.

In this paper, we proposed an end-to-end mandarin speech synthesis system based on Tacotron-2. This system can use text to generate mel-spectrum directly, all system train only one model. However, since the Chinese text is special and cannot be the input for the system directly, the front-end processing is necessary. In the front-end processing stage, we transform Chinese words into pinyin with tone to make embedding more concise. We also add prosody and polyphone information into the input to get a better result. Finally, we use linear-spectrum to generate waveform with Griffin-lim algorithm.

In the following sections, data processing, front-end, back-end, and system architecture will be introduced with more details.

2. The task in Blizzard 2019

In Blizzard Challenge 2019, the task is to build a speech synthesis system with data about 8 hours mandarin from a show recorded by a male speaker. These recordings consist of 480 audio files of which type is ‘mp3’ and every audio is about 60 seconds. The only aim of our system is to generate a new voice as similar as possible to the voice provided by a real male speaker. More details about our system will be introduced in the following sections.

3. TJU-Didi-Huiyan System

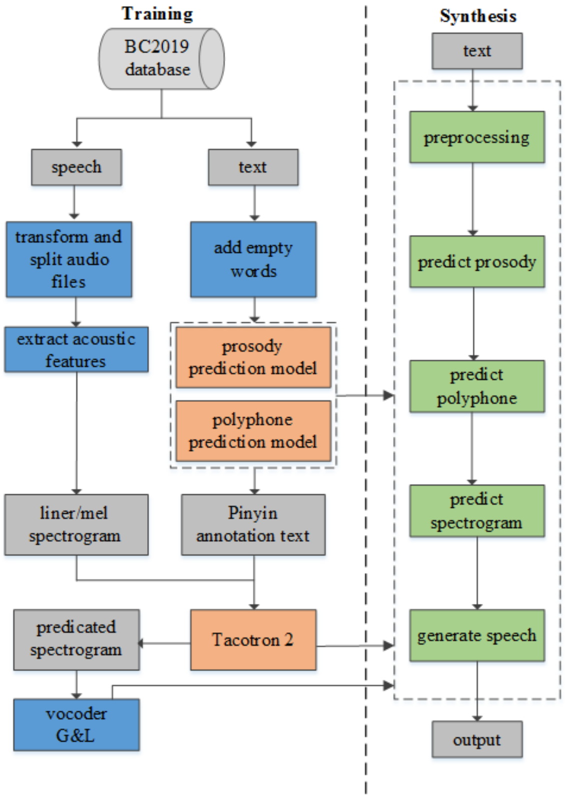

Figure 1: The flowchart of TJU-Didi-Huiyan text-to-speech system.

3.1. Data processing

The training data were provided by the committee and that consist of three parts, the first part is 480 audio files of which type is 'mp3'. The second part is a text file which contains all text according to every audio. The third part is the license.

The format of provided audio files is 'mp3'. Since the 'wav' format is more suitable for our system, we transform all audio files' format to 'wav' by sox. Shorter sentences are helpful for the model to learn more information. Thus, we divided every audio into several clauses according to the annotation provided by the force-alignment system.

For original provided texts, we add some extra words (such as empty words) to them. Using the expanded texts instead of original texts, we can get a better alignment result.

In addition, since other information such as prosody, pinyin with the tone, polyphone is very important to a mandarin text-to-speech system, we annotated this information for each sentence. Thus, the model can learn more details from the given data.

We use Griffin-lim algorithm to extract spectrum as acoustic features, which are used for the testing and training of the speech synthesis model.

3.2. Front-End

This challenge is to synthesize speech of Chinese. So our front-end processing part is mainly divided into the following parts: regularization processing, word Segmentation, Chinese characters to pinyin (G2P), and prosody prediction.

3.2.1. Regularization

From [12], we can realize that there are often a large number of non-standard words in the real text, these words are not found in the dictionary, their pronunciation cannot be obtained through the normal Pinyin rules [12], these non-standard words are meant to contain non-Chinese characters (such as Arabic numerals, English characters, Words of various symbols, etc.), in which non-Chinese characters need to be converted into corresponding Chinese characters, this conversion process is called text regularization. Difficulties in this work: First, the regularization object, the non-Chinese character string form is complex and diverse, difficult to generalize; second, the non-Chinese character string is ambiguous and needs disambiguation. Text regularization is a key part of speech synthesis and it directly affects the quality of speech services.

A three-layer model of text regularization is proposed in [12]: non-standard word recognition, disambiguation, and standard word generation. The maximum entropy model is introduced in the disambiguation of non-standard words. This method directly processes the actual text without the need for word segmentation and labeling.

So our final annotation effect example is as follows: for example, the number string "11" should be read as "yāo yāo" in the phone number, and be read as "yī yī" in "2.11cm".

In order to minimize some of the avoidable errors in the middle, and also to achieve the final best speech synthesis, we mainly use artificial regularization transformation.

3.2.2. Word Segmentation

The Chinese text does not have an explicit table such as an English space to mark the boundary mark of the word. Therefore, the task of Chinese automatic word segmentation is to automatically add space between words and words in the Chinese text by the machine. With the development of Chinese information processing, Chinese word segmentation has also been considerably developed, and numerous algorithms have appeared. According to its characteristics, the existing word segmentation algorithms can consist of four categories: segmentation methods based on string matching, word segmentation methods based on understanding, word segmentation methods based on statistics, and word segmentation methods based on semantics. Among them, the treatment of divergence includes two parts: (1) detection of divergent meanings; (2) digestion of divergent meanings. These two parts can be logically split into two relatively independent steps.

We use semantic rules to perform word segmentation on corpus data, and the final effect is as follows:

Input: 刘华清楚地重游。

Output: 刘华清 | 楚地 | 重游。

Even though we have our own automatic word segmentation system, we used manually labeled data to reduce the final error in this challenge.

3.2.3. G2P

In the Chinese speech synthesis system, the task of word-to-speech conversion is to convert the sequence of characters into corresponding pinyin sequences, which is an indispensable module of the speech synthesis system (TTS), and its correct rate directly affects the intelligibility of the speech synthesis system. In most cases, the word-to-speech conversion retrieves the current word in the dictionary, with the corresponding pinyin. However, some words in Chinese correspond to multiple Pinyin, so the choice of the correct pronunciation for a multi-word case is a difficult point.

If using Chinese characters as the input directly in Chinese speech synthesis, there is not the way to encode them. Because there are too many Chinese characters, one word per word is unrealistic, so the Chinese characters are transformed into pinyin. This only encodes twenty-six letters, and there is numerical punctuation. It's easier to implement coding.

Lexicon

The pronunciation dictionary (lexicon) contains a mapping from words to phones, which are used to connect acoustic models to language models. The pronunciation dictionary contains a collection of words that the system can handle and indicates its pronunciation. The mapping relationship between the modeling unit of the acoustic model and the modeling unit of the language model is obtained through the pronunciation dictionary so that the acoustic model and the language model are connected to form a search state space for the decoder to perform decoding.

Implementation steps: the first thing to ascertain is the conversion rule/mapping relationship from pinyin to phoneme. There may be very different mapping relationships, such as the pinyin of Chinese character one "yi1" may correspond to "ii i1" or "yi1". Then you require to list as many Chinese words and their corresponding pinyin. If there are multiple words, you can list different combinations. It should be noted that the pronunciation dictionary needs to cover as many words as possible.

The final effect is as follows:

Input: 我爱北京天安门。

Output: wo3 ai4 bei3 jing1 tian1 an1 men2

Polyphone

In Chinese speech synthesis, since there is a plurality of pronunciation in a Chinese character, that is, the existence of homographs of Chinese characters, the elimination of multi-syllable words in word-to-speech conversion is also a crucial step.

At present, the method of multi-word disambiguation has a TBL-based multi-phonetic phonetic algorithm [13], based on the error-driven rule machine learning algorithm, which compares the initial state of the sample with the correct mark, and uses greedy search for the sample with errors. The way to learn a series of correction rules [14], based on the maximum entropy model of multi-tone word disambiguation [15] and so on.

For inputs with polyphonic words, for example:

Input: 我在古都西安。(I am in the ancient capital of Xi'an.)

Output: wo3 zai4 gu3 du1 xi1 an1.

Among them, "都" is a multi-word, and finally, the correct pronunciation can be obtained.

Even though we have the model that can handle multi-word disambiguation, we used the way of manual annotation in order to reduce the error in the middle of the experiment.

Prosody prediction

As we all know, Chinese is a fluent language, which is the biggest difference between it and other Western languages. Every word in Chinese (except for children) is usually regarded as a tuned syllable. Each tune has some fixed type (baseband shape). But what we usually say is often a continuous statement consisting of multiple words. These pronunciations are affected by adjacent words. Transforming, even losing the creative type, this is the phenomenon of co-sounding that is often said in Chinese. This is why people get a sense of continuity when speaking, rather than being pronounced word by word. At the same time, there will be a short pause in the middle of the nonstop sentence pronunciation, which in turn reflects the rhythm of the person's speech. The main task of the Chinese TTS prosody model is to control the pronunciation of the TTS system by predicting the fundamental frequency, length, pause, etc. according to the information in the text so that the pronunciation is natural and nice.

There are two main methods for prosody level prediction: the first one, the prosody level prediction is usually predicted by the CRF (Conditional Random Field) model, that is, the CRF-based prosody level prediction method is used in the model. Introducing context information requires expanding the characteristics of the training, and introducing a manually written feature template to train the prosody level model. Secondly, the model used in prosody level prediction is built on the word granularity for training and prediction. The word segmentation system is utilized to obtain the training or predictive text segmentation results, and the features such as part of speech and word length are obtained, and the feature template is manually written. Generate corresponding text features for training and prediction.

In this system, we specify two types when prosody labeling: #1, a short pause, apply between two words; #3, long pause, apply between punctuation marks, the final effect is as follows.

Input: 我爱北京天安门。

Output: 我 #1 爱 #1 北京 #1 天安门 #3.

Due to the tight time and the co-sounding phenomenon in Chinese, we use rhythm labeling by hand. Now we have our own prosody prediction model.

3.3. Back-End

In order to obtain the final speech wave, there are still two tasks that predicting the spectrograms features and vocoding the resulting magnitude spectrograms. And we use the Tacotron 2 neural approach to complete the above tasks.

Tacotron 2 is composed of an encoder and a decoder with attention which predicts a sequence of mel-spectrogram frames from an input character sequence [11].

The encoder converts a character sequence into a hidden feature representation, which can be simplify described as:

$$f_e = \mathrm{ReLU}(F_3 * \mathrm{ReLU}(F_2 * \mathrm{ReLU}(F_1 * \bar{E}(X)))) \quad (1)$$

$$\mathrm{H} = \mathrm{EncoderRecurrency}(f_e) \quad (2)$$

where X is the input character sequence. In our system, the mandarin texts are preprocessed into a pinyin sequence including Latin alphabet, tone and rhythm annotation, and punctuation, etc. $\bar{E}(X)$ is the embedding represent of the above characters, and we use a fully connected layer to transfer the one-hot to embedding. What's more, F_3, F_2 and F_1 are three convolutional layers and each containing 512 filters with shape 5 × 1. And the output of the final convolutional layer (f_e) is passed into a single bi-directional LSTM layer containing 512 units (256 in each direction) to generate the encoded features (H).

The output of the encoder will be consumed by an attention network to form a fixed context vector for each decoder output step. As in Tacotron 2, we use location-sensitive attention which can be described as:

$$e_{ij} = \tanh(Ws_i + Vh_j + Uf_{i,j} + b) \tag{3}$$

where s_i is the i-th output of the decoder (current step), h_j is the hidden state of the encoder, b is a bias vector and all of the initial parameters are zero. Besides, the $f_{i,j}$ is location feature consumed by 32 1-D convolution filters. Obviously, the b is bias value, the W, V, and U are weight matrix used to keep the dimensions consistent, and the result e_{ij} is the attention alignment between s_i and h_j.

The decoder is an autoregressive recurrent neural network which predicts a mel-spectrogram from the encoded input sequence one frame at a time. The prediction from the previous time step is first passed through a small pre-net containing 2 fully connected layers of 256 hidden ReLU units. And in the training, we use the teacher forcing model (feeding in the correct output instead of the predicted output). We concatenated the pre-net output and attention context vector as the input of the decoder which consists of a stack of 2 uni-directional LSTM layers with 1024 units. Similarly, the concatenation of the LSTM output and the attention context vector is projected through a linear transform to predict the target spectrogram frame and stop token. The final output spectrogram is defined as:

$$y_{final} = y + y_r \tag{4}$$

$$y_r = \text{PostNet(y)} \tag{5}$$

y is the predicted spectrogram by the 2 uni-directional LSTM and linear transform. PostNet represents a 5-layer convolutional post-net (each layer consists of 512 filters with shape 5 × 1) which predicts a residual to add to the prediction to improve the overall reconstruction.

It is worth noting that we choose the liner spectrogram rather than mel-spectrogram, because we find that the liner spectrogram is much suitable for the G&L algorithm in this system. So the mel-spectrogram (80 dims) should be projected to liner spectrogram (1024 dims). Though that projection can be done just with a fully connected layer, but we use a "CBHG' module before the projection. The "CBHG" module which realized in the project [16] includes a convolution layer, 2 projection layers, a residual connection, and a highway net layer. In our test, we find that "CBHG" acting as an essential role for extracting the liner spectrogram features.

4. Results

We present the listening tests results of our system in Blizzard Challenge 2019. There are 26 systems in total, 24 from participating teams, one natural speech A and one merlin benchmark B. The participating systems are represented as C~Z, and our system is named as X.

4.1. Naturalness test

The boxplot of naturalness evaluation results is presented in figure 2. Though the results indicate that our system has a better performance on the naturalness test compared with merlin benchmark, the naturalness of our system still should be improved compared with natural speech and other state-of-the-art systems.

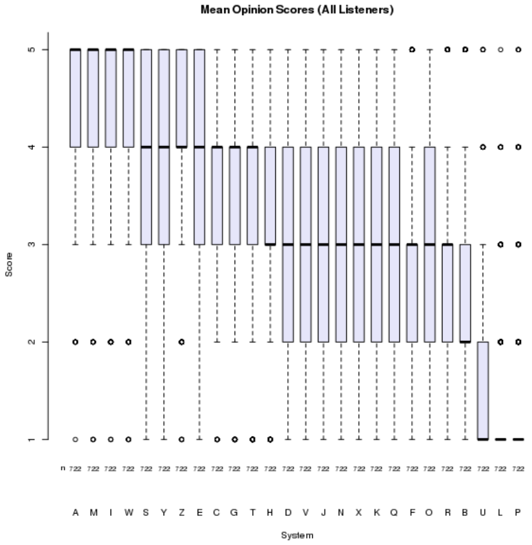

Figure 2: Boxplot of naturalness scores of each submitted system for all listeners.

4.2. Intelligibility test

The Pinyin error rates of all participant systems are presented in figure 3 and figure 4, representing without tone (PER) and with tone (PTER) respectively. When evaluated by all listeners, the PER of our system is 25.9%, and the PTER is 27.8%. Since we add some scripts in the pre-processing on the training data, these results show that the intelligibility of our system is reasonable.

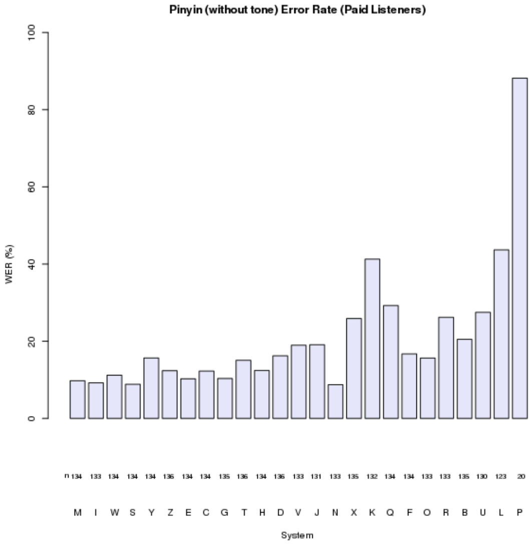

Figure 3: Pinyin error rate (without tone) of each submitted system.

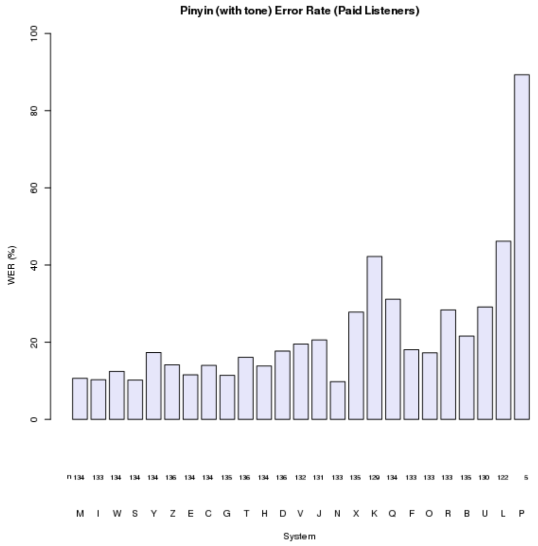

Figure 4: Pinyin error rate (with tone) of each submitted system.

4.3. Similarity test

The mean opinion of similarity evaluation compared to the original speaker is presented in Figure 5. From this result, we can find that the score of our system is just the same as the score of merlin benchmark. As shown in Figure 5, there is a significant difference between our systems and A, M, S, Y, Z, E and C. We will try to enhance our performance in the future to achieve the best results.

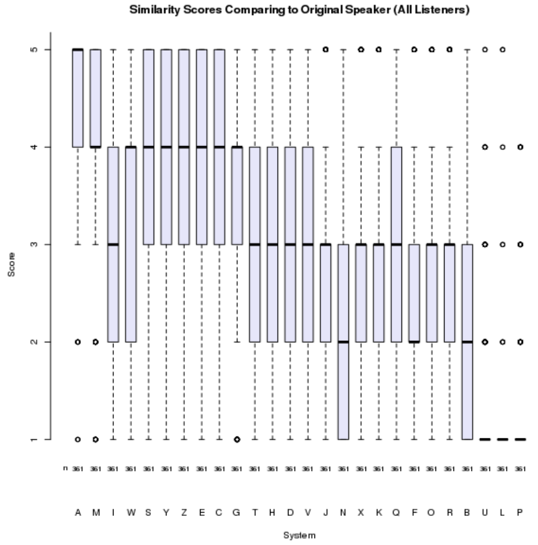

Figure 5: Boxplot of similarity scores of each submitted system for all listeners.

5. Conclusions and future work

This paper presents the details of our submitted system and the results in Blizzard Challenge 2019. We built an end-to-end speech synthesis system based on Tacotron 2. Our system can automatically complete the front-end processing of Chinese text (including prosody and multi-word prediction, etc.). Besides, the spectrograms features predicated by Tacotron 2 are used to generate the final sound through Griffin-lim algorithm.

Though the results show that our system has a better performance on the naturalness test compared with merlin benchmark, there are still many modules in the system that are not perfect enough. In future work, we will investigate the code-switched speech synthesis and quantization for deep neural network, and try to achieve good performance in all criterion for further speech synthesis challenge.

6. References

[1] Ling, Z. H., Qin, L., Lu, H., Gao, Y., Dai, L. R., Wang, R. H., & Hu, G. P, " USTC and iFlytek speech synthesis systems for Blizzard Challenge 2007, *In Blizzard Challenge Workshop*, 2007.

[2] Liu, L. J., Ding, C., Jiang, Y., Zhou, M., & Wei, S, "The IFLYTEK system for blizzard challenge 2017". I*n Blizzard Challenge Workshop*, Stockholm, 2017.

[3] Yoshimura, T., Tokuda, K., Masuko, T., Kobayashi, T., & Kitamura, T. (1999). Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis. *In Sixth European Conference on Speech Communication and Technology*.

[4] H. Ze, A. Senior, and M. Schuster, "Statistical parametric speech synthesis using deep neural networks," *in Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 7962–7966.

[5] H. Zen and H. Sak, "Unidirectional long short-term memory recurrent neural network with recurrent output layer for low-latency speech synthesis," *in Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 4470–4474.

[6] Ze, Heiga, Andrew Senior, and Mike Schuster, "Statistical parametric speech synthesis using deep neural networks," *in Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp.7962-7966.

[7] Ling Z H, Kang S Y, Zen H, et al., "Deep learning for acoustic modeling in parametric speech generation: A systematic review of existing techniques and future trends" *IEEE Signal Processing Magazine*, 2015,vol.3, pp.35-52.

[8] Wu Z, Valentini-Botinhao C,Watts O, et al., "Deep neural networks mploying multi-task learning and stacked bottleneck features for peech synthesis," i*n Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp.4460-4464.

[9] A. 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:1609.03499, 2016.

[10] Wang Y, Skerry-Ryan R J, Stanton D, et al., "Tacotron: A fully nd-to-end text-to-speech synthesis model" arXiv preprint, 2017.

[11] Shen J, Pang R, Weiss R J, et al., "Natural TTS synthesis by conditioning avenet on mel spectrogram predictions" arXiv preprint rXiv: 1712.05884, 2017.

[12] JIA Yuxiang, HUANG Dezhi, LIU Wu, YU Shiwen, "Text Normalization in Chinese Text2to2Speech System," *JOURNAL OF CHINESE INFORMATION PROCESSING*, vol.22, no.5, 2008, pp.44-50.

[13] LIU Fangzhou， ZHOU You.Polyphone disambiguation based on tree-guided TBL. *Computer Engineering and Applications*， 2011, vol.47, pp.137-140.

[14] Novak, Josef Robert, Nobuaki Minematsu, and Keikichi Hirose, "Phonetisaurus: Exploring grapheme-to-phoneme conversion with joint n-gram models in the WFST framework," *Natural Language Engineering*, vol.22, no.6, 2016, pp.907-938.

[15] Fangzhou Liu, Qin Shi, Jianhua Tao, "Maximum Entropy Based Homograph Disambiguation," *in National Conference on Man-Machine Speech Communication (NCMMSC),* 2007, pp.1-6.

[16] Rayhane Mama, Tensorflow implementation of DeepMind's Tacotron-2, https://github.com/Rayhane-mamah/Tacotron-2.